

Tivoli® NetView® for z/OS™



Installation: Getting Started

Version 5 Release 1

Tivoli® NetView® for z/OS™



Installation: Getting Started

Version 5 Release 1

Tivoli NetView for z/OS Installation: Getting Started

Copyright Notice

© Copyright IBM Corporation 2001, 2002. All rights reserved. May only be used pursuant to a Tivoli Systems Software License Agreement, an IBM Software License Agreement, or Addendum for Tivoli Products to IBM Customer or License Agreement. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without prior written permission of IBM Corporation. IBM Corporation grants you limited permission to make hardcopy or other reproductions of any machine-readable documentation for your own use, provided that each such reproduction shall carry the IBM Corporation copyright notice. No other rights under copyright are granted without prior written permission of IBM Corporation. The document is not intended for production and is furnished "as is" without warranty of any kind. **All warranties on this document are hereby disclaimed, including the warranties of merchantability and fitness for a particular purpose.**

U.S. Government Users Restricted Rights—Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corporation.

Trademarks

IBM, the IBM logo, Tivoli, the Tivoli logo, AIX, APPN, BookManager, CICS, IBMLink, IMS, MVS, NetView, OS/390, RACF, S/390, SAA, Tivoli Enterprise, TME, TME 10, VTAM, and z/OS are trademarks or registered trademarks of International Business Machines Corporation 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.

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 no intended Programming Interfaces that allow the customer to write programs to obtain services of Tivoli NetView for z/OS.

Contents

Preface	vii
Who Should Read This Document	vii
What This Document Contains	vii
Publications	vii
Prerequisite and Related Documents	vii
Accessing Publications Online	viii
Ordering Publications	viii
Providing Feedback about Publications	viii
Contacting Customer Support	viii
Accessibility Information	ix
Keyboard Access	ix
Conventions Used in This Document	ix
Platform-specific Information	ix
Terminology	ix
Reading Syntax Diagrams	x
Required Syntax	xi
Optional Keywords and Variables	xi
Default Values	xi
Long Syntax Diagrams	xii
Syntax Fragments	xii
Commas and Parentheses	xiii
Highlighting, Brackets, and Braces	xiv
Abbreviations	xv
Chapter 1. Understanding the NetView Installation	1
NetView and Network Organization	1
Organization of the NetView Program	1
Naming Conventions	3
Installation Package	4
NetView Program Directory	4
Preventive Service Planning (PSP)	4
Applying a Required PTF	4
Estimating Required Resources	5
Installation Procedures	5
Installation Results	5
Chapter 2. Preparing the MVS System.	7
Using SMP/E to Load the V5R1 Libraries	7
Modifying the Maximum Number of Language Processor (REXX) Environments for NetView	7
Updating Member SCHEDxx	7
Updating Member LPALSTxx	8
Updating Member PROGxx	9
Updating Member LNKLSTxx	10
Updating Member IEAAPFxx	10
Updating Member IEFSSNxx	11
Updating Member IEASYSxx	12
Updating Member IEASYMxx	13
Updating Member LOADxx	13
Updating Member COUPLExx	13
Updating Member SMFPRMxx	14
Using MVS Consoles with NetView	14
Using Subsystem Allocatable Consoles	14
Using EMCS Console Support	14
Assigning Unique EMCS Console Names	15
Restarting the Target System with the CLPA Option	16

Chapter 3. Preparing NetView	17
Creating the Installation JCL Samples Library Using Job CNMSJBUP	17
Defining the User Catalog Using Job CNMSJ001	17
Allocating Partitioned Data Sets Using Job CNMSJ002	18
Changing the Domain and Subarea Numbers Using Job CNMSJ000	18
Loading Members of Partitioned Data Sets Using Job CNMSJ003	19
Allocating VSAM Clusters Using Job CNMSJ004	20
Loading Sample Databases for Verification Using Job CNMSJ005	23
Chapter 4. Preparing VTAM	25
Modifying the Application (APPL) Major Node	25
Defining the NetView Management Console to VTAM (LU 6.2 only)	25
Defining the SNA Topology Manager to VTAM	26
Defining the VTAM Primary Program Operator Interface Task	26
Changing the Password	26
Changing the Domain Name (DOMAINID).	26
Changing the Logmode Table (LOGMODE)	27
Defining Resources to the Status Monitor (STATOPT)	27
Defining Multiple Concurrent NetView Operators	27
Allowing for Additional TAF LU1 Operators	28
Allowing for Additional TAF LU2 Operators	28
Defining Alias Names	28
Reviewing VTAM and NCP Definitions for Mode Table and USS Table Changes	31
Assembling VTAM Tables Using Job CNMSJ006	31
Reviewing VTAM Start Options	32
Modifying the VTAM Start Procedure	33
Starting VTAM ACBs and Corresponding NetView Tasks	33
Chapter 5. Getting Ready to Start NetView	35
NetView and Subsystem Application Procedures	35
Modifying the NetView Subsystem Interface Procedure.	35
Modifying the NetView Startup Procedure	39
Specifying the Japanese Version	39
Setting System Variables	39
Setting the Dispatch Priority.	40
Modifying the Buffer Size and Slot Size	40
Defining Command Lists	40
Specifying Definition Members and Sense Codes	41
Running with Information/Management	42
Enabling Network and Trace Logs to Print Automatically	42
Omitting Some of the Logs from Automatic Printing	42
Defining TCP/IP to NetView	43
Running the NetView Application as a Job	43
Updating CNMSTYLE.	44
Enabling Towers.	44
%INCLUDE Members	46
Using Symbolics.	46
Setting up Security	47
Specifying That a Command Is to Run Automatically When the NetView Program Is Started	47
Customizing NetView Components in CNMSTYLE	48
Starting the NetView Program Before Starting JES	49
Chapter 6. Activating NetView	51
Starting VTAM Using Job CNMNET	51
Starting the NetView Subsystem Address Space Using Job CNMPSSI	52
Starting the NetView Program Using Job CNMPROC	53
Chapter 7. Verifying the Installation	55
Testing the Command Facility	55
Testing the Hardware Monitor	68

Testing the Session Monitor	70
Preparing for Production Test	73
Appendix A. NetView Functions and Tasks	75
Appendix B. NetView Samples Overview	81
NCP Samples.	82
Coding an NCP	82
NCP Definition	83
VTAM Major Node Names and Definitions.	83
VTAM Major Node Names	84
VTAM Definitions	85
VTAM Tables	100
Modifying the VTAM Start Procedure	102
Appendix C. NetView Samples Reference	103
Appendix D. Partitioned Data Sets and Members	123
Index	125

Preface

This document is designed to help system programmers install Tivoli® NetView® for z/OS™ (NetView) with the base functions.

Who Should Read This Document

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

What This Document Contains

Tivoli NetView for z/OS Installation: Getting Started contains the following sections:

- “Chapter 1. Understanding the NetView Installation” on page 1 gives an overview of the installation task.
- “Chapter 2. Preparing the MVS System” on page 7 provides instructions to prepare your MVS™ system.
- “Chapter 3. Preparing NetView” on page 17 provides instructions to install the V5R1 NetView program.
- “Chapter 4. Preparing VTAM” on page 25 provides instructions to update VTAM® definitions.
- “Chapter 5. Getting Ready to Start NetView” on page 35 provides instructions to update NetView definitions for your environment.
- “Chapter 6. Activating NetView” on page 51 provides instructions to start the NetView program.
- “Chapter 7. Verifying the Installation” on page 55 leads you through a series of commands to test the NetView program you have just started.
- The following appendixes are also provided:
 - Appendix A. NetView Functions and Tasks
 - Appendix B. NetView Samples Overview
 - Appendix C. NetView Samples Reference
 - Appendix D. Partitioned Data Sets and Members

Publications

This section lists prerequisite and related documents. It also describes how to access Tivoli publications online, how to order Tivoli publications, and how to make comments on Tivoli publications.

Prerequisite and Related Documents

To read about the new functions offered in this release, refer to the *Tivoli NetView for z/OS Installation: Migration Guide*.

You can find additional product information on these Internet sites:

Table 1. Resource Web sites

IBM®	http://www.ibm.com/
Tivoli Systems	http://www.tivoli.com/

Preface

Table 1. Resource Web sites (continued)

Tivoli NetView for z/OS	http://www.tivoli.com/nv390
-------------------------	---

The Tivoli NetView for z/OS Web site offers demonstrations of the NetView product, 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 the NetView program using the program-to-program interface (PPI)
- Sending and receiving MVS commands using the PPI
- Sending TSO commands and receiving responses

Accessing Publications Online

You can access many Tivoli publications online using the Tivoli Information Center, which is available on the Tivoli Customer Support Web site:

<http://www.tivoli.com/support/documents/>

These publications are available in PDF format. Translated documents are also available for some products.

Ordering Publications

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

<http://www.ibm.com/shop/publications/order>

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

- In the United States: 800-879-2755
- In Canada: 800-426-4968
- In other countries, for a list of telephone numbers, see the following Web site:
http://www.tivoli.com/inside/store/lit_order.html

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 an e-mail to pubs@tivoli.com.
- Complete our customer feedback survey at the following Web site:
<http://www.tivoli.com/support/survey/>

Contacting Customer Support

If you have a problem with any Tivoli product, you can contact Tivoli Customer Support. See the *Tivoli Customer Support Handbook* at the following Web site:

<http://www.tivoli.com/support/handbook/>

The handbook provides information about how to contact Tivoli Customer Support, depending on the severity of your problem, and the following information:

- Registration and eligibility
- Telephone numbers and e-mail addresses, depending on the country you are in
- What information you should gather before contacting support

Note: Additional support for Tivoli NetView for z/OS is available at the NetView for z/OS home page:

<http://www.tivoli.com/nv390> Under Related Documents, select **Other Online Sources**. The page displayed contains a list of newsgroups, forums, and bulletin boards.

Accessibility Information

Refer to *Tivoli NetView for z/OS User's Guide* for information about accessibility.

Keyboard Access

Standard shortcut and accelerator keys are used by the product and are documented by the operating system. Refer to the documentation provided by your operating system for more information.

Refer to *Tivoli NetView for z/OS User's Guide* for more information about keyboard access.

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 .
<i>Italics</i>	Variables and new terms appear like <i>this</i> , in <i>italics</i> . Words and phrases that are emphasized also appear like <i>this</i> , in <i>italics</i> .
Monospace	Code examples, output, and system messages appear like <code>this</code> , in a monospace font.
ALL CAPS	Tivoli NetView for z/OS 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 z/OS Licensed Program Specification*.

Terminology

For a list of Tivoli NetView for z/OS 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 z/OS Version 5 Release 1

Preface

- 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 OS/390 or z/OS operating systems.

RACF®

RACF is a component of the SecureWay® Security Server for z/OS and OS/390, providing the functions of authentication and access control for OS/390 and z/OS resources and data, including the ability to control access to DB2® objects using RACF profiles. Refer to:

<http://www-1.ibm.com/servers/eserver/zseries/zos/security/racfs.html>

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 are included in the IBM Communications Server element of the OS/390 and z/OS operating systems. Refer to <http://www.ibm.com/software/network/commserver/about/>.

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 1 specifies that the *resname* variable must be used for the CCPLOADF command.

CCPLOADF

▶▶—CCPLOADF *resname*—————▶▶

Figure 1. 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 2, MEMBER is an operand and *membername* is a variable that defines the name of the data set member for that operand.

TRANSMMSG

▶▶—TRANSMMSG MEMBER=*membername*—————▶▶

Figure 2. Syntax for Variables

Optional Keywords and Variables

Optional keywords, variables, and operands are below the main syntax line. Figure 3 specifies that the ID operand can be used for the DISPREG command, but is not required.

DISPREG

▶▶—DISPREG ———▶▶
 └ ID=*resname* ─┘

Figure 3. 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.

Preface

Figure 4 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

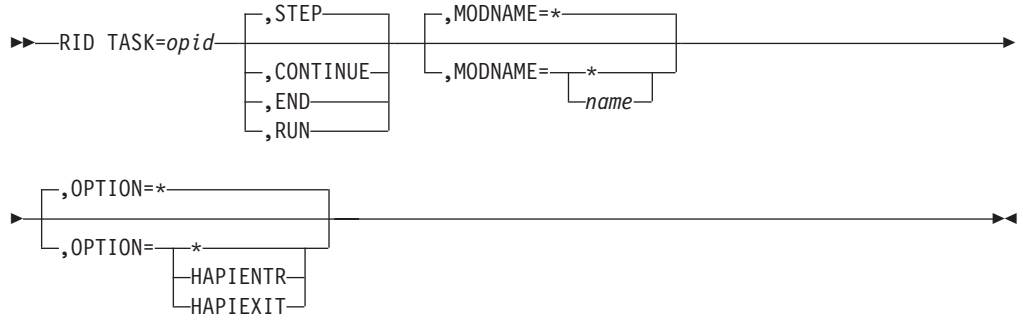


Figure 4. Sample of Defaults Syntax

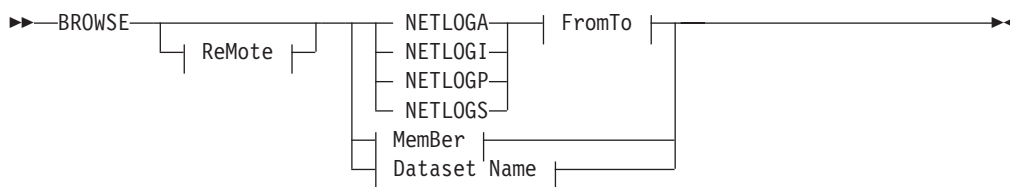
Long Syntax Diagrams

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 4.

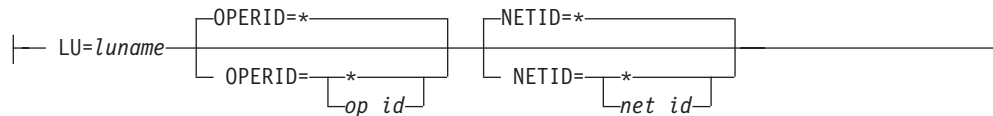
Syntax Fragments

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 5 on page xiii for a syntax with the fragments ReMote and FromTo.

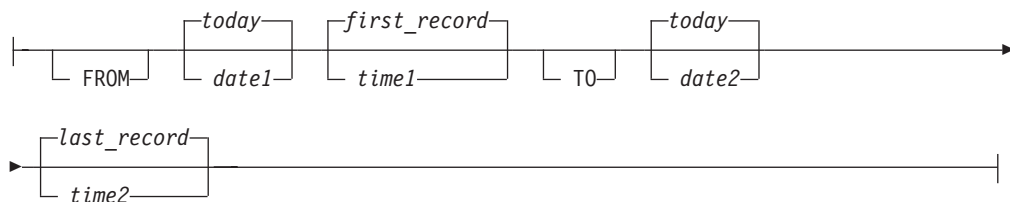
BROWSE



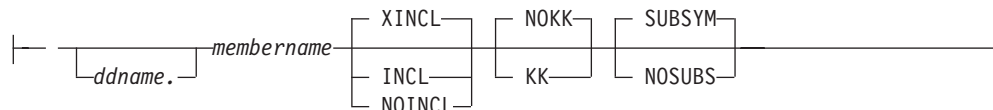
ReMote:



FromTo:



MemBer:



Dataset Name:



Figure 5. Sample Syntax Diagram with Fragments

Commas and Parentheses

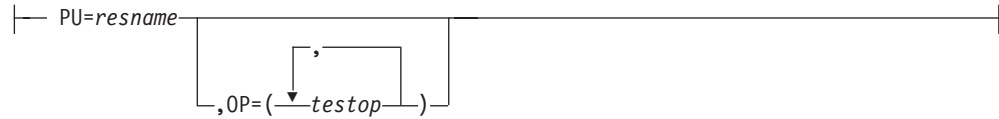
Required commas and parentheses are included in the syntax diagram. When an operand has more than one value, the values are typically enclosed in parentheses and separated by commas. In Figure 6 on page xiv, the OP operand, for example, contains commas to indicate that you can specify multiple values for the *testop* variable.

Preface

CSCF



Pu



PurgeAll



PurgeBefore

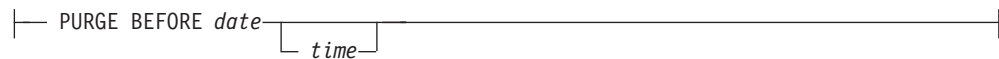


Figure 6. Sample Syntax Diagram with Commas

If a command requires positional commas to separate keywords and variables, the commas are shown before the keyword or variable, as in Figure 4 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	<i>resname</i>
Operand	MEMBER= <i>membername</i>
Default	<u>today</u> or INCL

Abbreviations

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

Chapter 1. Understanding the NetView Installation

Before beginning your installation, read the NetView program directory.

This document is designed to guide you through the installation and verification of the NetView program in a minimum amount of time. Initially, make only essential changes to the sample definitions provided with the NetView program. Your first objective is a fast and error-free installation.

Note: If you are a migrating user, refer to the *Tivoli NetView for z/OS Installation: Migration Guide*.

This chapter describes:

- The organization of the NetView program
- Installation of the NetView program

NetView and Network Organization

A sample network comes with NetView. It contains definitions for both SNA and non-SNA resources. You can use this sample network as a model for building a new network or for expanding your existing network. For more information about the sample network, see “Appendix B. NetView Samples Overview” on page 81.

Organization of the NetView Program

The NetView program enables you to manage your systems and networks from *one console* and can benefit users of small or large networks and small or large systems. Small installations with a limited staff can take advantage of NetView graphics and automation functions to simplify operators’ jobs and improve productivity. Graphics and automation can reduce, simplify, and in some cases even eliminate operator intervention, enabling faster and more reliable problem diagnosis and resolution.

These same advantages apply to large installations, contending with a higher volume of activity and greater complexity. These users can *automate* repetitive and syntax-sensitive tasks to improve reaction time and reduce human errors. They can also automate knowledge-intensive tasks, such as problem determination, and develop expert systems, using the NetView programming interfaces.

Figure 7 on page 2 shows the relationship between NetView components.

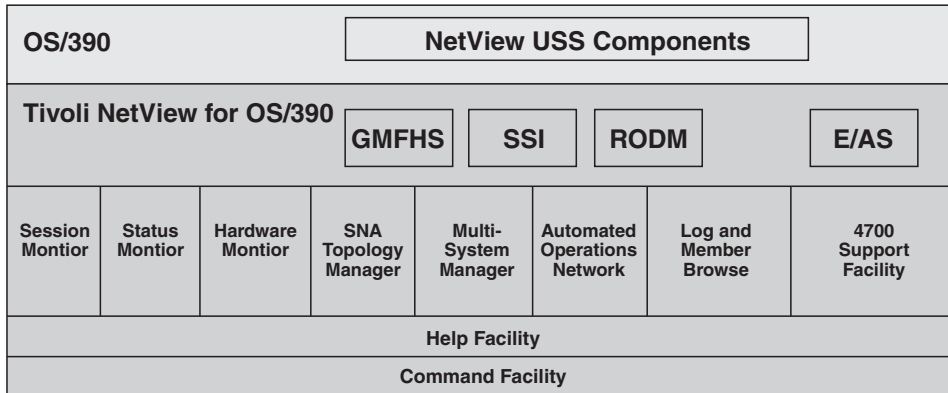


Figure 7. NetView Program Host Components

NetView includes:

- Command facility that enables you to send commands and receive messages. The command facility also provides base functions and services for components such as intercomponent communication, presentation services, database services, and automation facilities.
- Session monitor that provides information about SNA sessions (subarea and APPN[®]) including session partner identification, session status, connectivity of active sessions, and response time data.
- Status monitor that provides status information about SNA subarea network resources.
- Hardware monitor that collects and displays events and statistical data for both hardware and software applications to identify failing resources in a network.
- Integrated graphical interface that includes:
 - Resource Object Data Manager (RODM), an object-oriented data cache. Objects in RODM represent resources in your network.
 - Graphic Monitor Facility host subsystem which works with RODM and the NetView management console (NMC) to manage resources.
 - NetView management console, graphically displays network topology and status.
 - Graphics applications including MultiSystem Manager (used to manage non-SNA resources) and the SNA topology manager (used to manage SNA resources)
- Automation facilities that include:
 - Automation table
 - Automated tasks
 - Command lists
 - Timer commands
 - Installation exits

Additionally, there is drop-in automation for VTAM SNA and TCP/IP network resources using IBM-supplied automation policy in the Automated Operations Network component

- Programming facilities including REXX, PL/I, and PIPES support, and Application programming interfaces (APIs)
- Customization facilities including the VIEW command
- NetView bridge for accessing Tivoli Service Desk databases
- Remote operations support
- Subsystem interface (SSI) for communication with MVS

- Event/automation service that serves as a gateway for event data between the NetView management environment, the Tivoli Management Region environment, and Simple Network Management Protocol (SNMP) trap managers.
- NetView Resource Manager (NRM) agent
- Browse facility to view local or remote NetView data set members.
- Help and Help Desk facilities including:
 - General help and component information
 - Command help
 - Message help
 - Sense code information
 - Recommended actions
 - Helpdesk

For more information on NetView functions, see “Appendix A. NetView Functions and Tasks” on page 75.

Naming Conventions

The NetView program uses the following types of data sets:

- System data sets
- NetView partitioned data sets
- NetView VSAM data sets

NetView also uses hierarchical file system (HFS) data sets if you installed the optional NetView components relative to z/OS UNIX[®] System Services as described in the NetView program directory.

NetView allocates the following HFS directories:

```
./usr/lpp/netview/v5r1/bin/IBM/
./usr/lpp/netview/v5r1/mibs/IBM/
```

The default naming convention for system data sets is:

- The high-level qualifier is NETVIEW.V5R1M0, specified as &SQ1 in the NetView JCL.
- The low-level qualifier is the System Modification Program (SMP) target library name

For example, for data set NETVIEW.V5R1M0.CNMLINK, the high-level qualifier is NETVIEW.V5R1M0 and the low-level qualifier is CNMLINK.

The default naming convention for NetView partitioned data sets is:

- The high-level qualifier is NETVIEW.V5R1USER, specified as &Q1 in the NetView JCL.
- The middle-level qualifier is the host NetView domain
- The low-level qualifier is the library name

For example, with data set NETVIEW.V5R1USER.CNM01.DSIPARM, the high-level qualifier is NETVIEW.V5R1USER, the middle-level qualifier is CNM01, and the low-level qualifier is DSIPARM.

The default naming convention for NetView VSAM data sets is:

- The high-level qualifier is NETVIEW, specified as &VQ1 in the NetView JCL.
- The middle-level qualifier, when specified, is the host NetView domain

- In most cases, the low-level qualifier is defined by the DSTINIT primary VSAM member names (PDDNM) and secondary VSAM member names (SDDNM) parameters of the NetView initialization members

For example, with NETVIEW.CNM01.DSILOGP, the high-level qualifier is NETVIEW, the middle-level qualifier is CNM01, and the low-level qualifier is DSILOGP.

You can substitute your own values for &SQ1, &Q1, and &VQ1 in the Tivoli-supplied JCL samples and procedures.

Installation Package

The NetView program is shipped to you on a set of tapes in SMP/E format. These tapes are known collectively as the *distribution tape*. The NetView program directory accompanies the distribution tape.

Workstation-based NetView code is shipped in two formats:

- CD-ROM
- Tivoli Web site

For detailed information on the installation package contents, refer to the NetView program directory.

NetView Program Directory

The *Program Directory for Tivoli NetView for z/OS* is shipped with the product and is also available on the Tivoli web site. This directory is available in English and Japanese versions. The Program Directory's primary function is to provide instructions to install the NetView product to the System Modification Program (SMP) libraries. In addition, it contains helpful hints for a problem-free installation. It also lists the program temporary fixes (PTFs) that you should apply to enable other products to work properly with NetView. *Reading the NetView program directory from cover to cover is recommended.* Some sections might not apply to you, depending on which packaging option you are installing. Each section is clearly marked, so you know at a glance whether it applies to you.

Preventive Service Planning (PSP)

Testing with field-test and early-ship customers continues after the NetView program is manufactured and ready for shipment. If problems are found, they are collected by Tivoli Customer Support in what is called the *PSP bucket*. When you receive your copy of NetView, call Tivoli Customer Support to request a copy of the PSP bucket.

In the United States, the telephone number for Tivoli Customer Support is 1-800-237-5511. If you have an IBMLINK subscription, you can receive a copy of the PSP bucket through that service.

Applying a Required PTF

The NetView program directory lists all PTFs (code fixes) that have been created for z/OS, MVS, and other Tivoli Systems network and systems management products to work with this NetView release. It also lists all of the PTFs from previous NetView releases that have been incorporated into this release. You can

| get the PTFs either by calling IBM Customer Support, or by using SUF (Service
| Update Facility) or IBMLINK to download the files electronically. The SUF web
| address is:

| www-1.ibm.com/servers/eserver/zseries/zos/suf/

| In the United States, the telephone number for IBM Customer Support is
| 1-800-237-5511.

Estimating Required Resources

Refer to the *Tivoli NetView for z/OS Tuning Guide* and the NetView program directory for information about estimating the hardware and software resources required to install NetView. If you cannot determine how to estimate the resource types, call Tivoli Customer Support at 1-800-237-5511 or contact your Tivoli Systems account representative.

Installation Procedures

Installation consists of:

1. Preparing your MVS system for the NetView program
2. Loading the NetView program from the distribution tape using z/OS SMP/E with the help of the NetView program directory
3. Customizing and running sample jobs to install the basic functions of the NetView program and the sample network
4. Preparing VTAM system definitions
5. If wanted, starting and verifying the sample network using the sample databases supplied on the NetView distribution tape.

Use the sample network as a model for building a new network or for expanding your existing network. All VTAM, NetView, and NCP definition statements are supplied as samples. The CNMSAMP library provides the job control language (JCL) and data necessary to build the entire network in subarea 01. Also included, is network NETB which shows a multiple-network environment. Network NETB has one host subarea, 01.

This book assumes that you are installing NetView using the defaults provided in this manual and in the sample files (installation in subarea 01 and using domain CNM01 for network NETA). If you are installing the NetView program in a subarea other than 01 or a domain other than CNM01, see the sample files for directions on what to change.

Installation Results

| When you have completed the procedures described in this book, the following
| functions are operational:

- Base NetView including the command facility
- Automation table to respond automatically to messages and MSUs, including alerts
- NetView log that stores messages and commands
- Save / Restore VSAM facility to save critical information across NetView restarts
- NetView to NetView remote command execution (RMTCMD)

| Once you have installed the NetView program and it is functioning correctly, there
| are many NetView functions that are not yet active by default. See “Appendix A.
| NetView Functions and Tasks” on page 75 for a list of NetView functions that you
| can enable. Refer to *Tivoli NetView for z/OS Installation: Configuring Additional*
| *Components* for information on how to adapt the NetView system to your own
| production environment and to *Tivoli NetView for z/OS Installation: Configuring*
| *Graphical Components* to complete your installation for graphics support.

Chapter 2. Preparing the MVS System

This chapter describes the steps necessary to prepare your MVS system if you are installing the NetView V5R1 program.

NetView consists of the following address spaces:

- Base NetView (including the command facility, session monitor, status monitor, and hardware monitor)
- NetView subsystem address application
- Resource Object Data Manager (RODM)
- Graphic Monitor Facility host subsystem (GMFHS)
- Event/Automation Service

The installation of RODM and of GMFHS (including applications dependent on GMFHS such as the MultiSystem Manager and the SNA Topology Manager) are covered in *Tivoli NetView for z/OS Installation: Configuring Graphical Components*. The installation of the Event/Automation Service is covered in *Tivoli NetView for z/OS Installation: Configuring Additional Components*.

Using SMP/E to Load the V5R1 Libraries

Use the information in the NetView program directory to load the NetView libraries from the distribution tape.

Modifying the Maximum Number of Language Processor (REXX) Environments for NetView

| 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 (for
| example, MultiSystem Manager and AON) as well as base NetView functions
| exercise code that has been written in REXX. The NetView program also contains
| several parts that make use of the Data REXX function. The Data REXX function
| enables you to include REXX instructions and functions in data files. In order to
| initialize NetView, you may need to adjust the maximum number of language
| processor environments defined by TSO that the system can initialize for the
| NetView address space.

If you want information about...

Refer to...

Language Processor (REXX) Environments
for NetView

*Tivoli NetView for z/OS Installation: Configuring
Additional Components*

Updating Member SCHEDxx

Define the NetView program as nonswappable in MVS. The NetView program always runs in MVS storage key 8.

Note: Storage key 8 is the default.

The NetView program subsystem-interface address space does not need any specifications because it is able to make itself nonswappable. To make the NetView

program nonswappable, use the SCHEDxx member of the SYS1.PARMLIB in place of the MVS program properties table (PPT) module IEFSD060. Ensure the SCHEDxx statements for the NetView program are:

- The NetView program with the hardware monitor (NPDA), PGM=BNJLINTX in your NetView JCL PROC:
PPT PGMNAME(BNJLINTX) NOSWAP KEY(8)
- The NetView program without the hardware monitor (NPDA), PGM=DSIMNT in your NetView JCL PROC:
PPT PGMNAME(DSIMNT) NOSWAP KEY(8)
- The RODM program, PGM=EKGTC000 in your RODM JCL PROC:
PPT PGMNAME(EKGTC000) NOSWAP NOCANCEL
- The NetView GMFHS program, PGM=DUIFT000 in your GMFHS JCL PROC:
PPT PGMNAME(DUIFT000) NOSWAP KEY(8)

Updating Member LPALSTxx

Member LPALSTxx contains the names of program libraries you want the system to concatenate to SYS1.LPALIB. This concatenation is called the LPALSTxx concatenation, which the system uses to build the pageable link pack area (PLPA).

Use the following steps to update member LPALSTxx in the SYS1.PARMLIB library:

1. Add the NetView library SCNMLPA1 to the LPALSTxx member.

Notes:

- a. The NetView notify routine, DSISPNOT, and routine BNJSVC76 should not reside in both SYS1.LPALIB and the NetView library SCNMLPA1. Do not concatenate SCNMLPA1 to the STEPLIB in CNMPROC.
 - b. If you wish to run a previous release of the NetView program on the same system as V5R1, add the V5R1 SCNMLPA1 to LPALSTxx. Ensure that LPALSTxx does not include any previous NetView SCNMLPA1.
2. Load module CNMCSRVP can be optionally loaded in LPALSTxx. Doing so will decrease the chance that a program running in module CNMCSRVP will abend if it is running when the SSI is ended. However, if you add CNMCSRVP to LPALSTxx, re-IPL MVS following any changes to that module.
 3. If you are installing RODM and 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, and to 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.

Note: Data sets listed in LPALSTxx must be authorized in IEAAPFxx and cataloged in the USER CATALOG(VOLSER), for example NETVIEW.V5R1M0.SCNMLPA1(NVPTFM).

If you want information about...	Refer to...
Modifying the IEFUSI step initiation exit	z/OS library
LPALSTxx	z/OS library

Updating Member PROGxx

Member PROGxx contains the names of program libraries that you want the system to concatenate to SYS1.LINKLIB as well as libraries that you want to define as authorized with the Authorized Program Facility (APF). You can use this member instead of members LNKSTxx and IEAAPFxx.

Add the following program libraries to the PROGxx member as concatenated data sets to SYS1.LINKLIB:

- NETVIEW.V5R1M0.CNMLINK
- NETVIEW.V5R1M0.SCNMLNKN

Note: SCNMLNKN is used for RODM trace, MVS command exit, and DSIPHONE. It is not referenced in the NetView program samples.

The following libraries must be authorized with the Authorized Program Facility (APF). To be authorized, a library's name and volume ID must appear in the list of authorized libraries in the PROGxx member of SYS1.PARMLIB.

1. If these are not already authorized, authorize all the libraries from the //STEPLIB, //VTAMLIB, and //NCPLOAD concatenations in your VTAM start procedure. The VTAM start procedure, CNMSJ008 (CNMNET), in NETVIEW.V5R1M0.CNMSAMP is the default member shipped with the NetView program.
2. Authorize all the libraries from the //STEPLIB concatenation in the following procedures:
 - The NetView start procedure, CNMSJ009 (CNMPROC), in NETVIEW.V5R1M0.CNMSAMP
 - The NetView procedure to start the subsystem interface (SSI) address space, CNMSJ010 (CNMPSSI), in NETVIEW.V5R1M0.CNMSAMP
 - The RODM start procedure, EKGXRODM, in NETVIEW.V5R1M0.CNMSAMP
 - The GMFHS start procedure, CNMSJH10 (CNMGMFHS), in NETVIEW.V5R1M0.CNMSAMP
 - The Event/Automation service (E/AS) start procedure, IHSAEVNT, in NETVIEW.V5R1M0.SCNMUXMS
3. Authorize the following NetView libraries:
 - NETVIEW.V5R1M0.SCNMLNK1
 - NETVIEW.V5R1M0.SCNMLPA1
 - NETVIEW.V5R1M0.CNMLINK
 - NETVIEW.V5R1M0.SCNMLNKN
 - NETVIEW.V5R1M0.SCNMUXLK (for customers using z/OS UNIX related services)

Note: If you are installing the Japanese program V5R1, authorize the NetView data set, SCNMMJPN. This data set is in the STEPLIB of CNMPROC.

4. If you are using a component that requires the REXX library, add the SEAGLMD or SEAGALT data set to the list of APF-authorized libraries.

If you specify an unauthorized library on a STEPLIB or concatenate unauthorized libraries with authorized libraries, all libraries are treated as if they are unauthorized.

To add the NetView libraries to the list of APF-authorized libraries without requiring a system IPL, you can first enable dynamic APF changes by issuing the following MVS command:

```
SETPROG APF,FORMAT=DYNAMIC
```

Then, for each NetView library, issue the following MVS command:

```
SETPROG APF,ADD,DSNAME=dsname,VOLUME=volser
```

If you want information about...	Refer to...
APF authorization	z/OS library

Updating Member LNKLSTxx

Member LNKLSTxx contains the names of program libraries that you want the system to concatenate to SYS1.LINKLIB. If you have updated PROGxx, the changes that follow are not needed.

Add the following program libraries to the LNKLSTxx member that defines the linklist for the target system:

- NETVIEW.V5R1M0.CNMLINK
- NETVIEW.V5R1M0.SCNMLNKN

Note: SCNMLNKN is used for RODM trace, MVS command exit, and DSIPHONE. It is not referenced in the NetView program samples.

Data sets listed in LNKLSTxx must be cataloged in the system master catalog.

If you want information about...	Refer to...
LNKLSTxx	z/OS library

Updating Member IEAAPFxx

Member IEAAPFxx contains the names of program libraries that you want the system to authorize with the Authorized Program Facility (APF). If you have updated PROGxx, the changes that follow are not needed.

The following libraries must be authorized with APF. To be authorized, a library's name and volume ID must appear in the list of authorized libraries in the APF member (IEAAPFxx) of SYS1.PARMLIB.

1. If these are not already authorized, authorize all the libraries from the //STEPLIB, //VTAMLIB, and //NCPLD concatenations in your VTAM start procedure. The VTAM start procedure, CNMSJ008 (CNMNET), in NETVIEW.V5R1M0.CNMSAMP is the default member shipped with the NetView program.
2. Authorize all the libraries from the //STEPLIB concatenation in the following procedures:
 - The NetView start procedure, CNMSJ009 (CNMPROC), in NETVIEW.V5R1M0.CNMSAMP
 - The NetView procedure to start the subsystem interface (SSI) address space, CNMSJ010 (CNMPSSI), in NETVIEW.V5R1M0.CNMSAMP
 - The RODM start procedure, EKGXRODM, in NETVIEW.V5R1M0.CNMSAMP

- The GMFHS start procedure, CNMSJH10 (CNMGMFHS), in NETVIEW.V5R1M0.CNMSAMP
 - The Event/Automation service (E/AS) start procedure, IHSAEVNT, in NETVIEW.V5R1M0.SCNMUXMS
3. Authorize the following NetView libraries:
- NETVIEW.V5R1M0.SCNMLNK1
 - NETVIEW.V5R1M0.SCNMLPA1
 - NETVIEW.V5R1M0.CNMLINK
 - NETVIEW.V5R1M0.SCNMLNKN
 - NETVIEW.V5R1M0.SCNMUXLK (for customers using z/OS UNIX System Services)

Note: If you are installing the Japanese program V5R1, authorize the NetView data set, SCNMMJPN. This data set is in the STEPLIB of CNMPROC.

4. If you are using a component that requires the REXX library, add the SEAGLMD or SEAGALT data set to the list of APF-authorized libraries.

If you specify an unauthorized library on a STEPLIB or concatenate unauthorized libraries with authorized libraries, all libraries are treated as if they are unauthorized.

To add the NetView libraries to the list of APF-authorized libraries without requiring a system IPL, you can first enable dynamic APF changes by issuing the following MVS command:

```
SETPROG APF,FORMAT=DYNAMIC
```

Then, for each NetView library, issue the following MVS command:

```
SETPROG APF,ADD,DSNAME=dsname,VOLUME=volser
```

If you want information about...	Refer to...
APF authorization	z/OS library

Updating Member IEFSSNxx

Update member IEFSSNxx in SYS1.PARMLIB. This member contains parameters that define the secondary subsystems during MVS system initialization. Each 80-byte IEFSSNxx record contains parameters defining a single secondary subsystem.

1. Add the RODM subsystem name EKGX to member IEFSSNxx of the SYS1.PARMLIB data set.
2. Add the NetView subsystem name to member IEFSSNxx of the SYS1.PARMLIB data set. The format of the NetView subsystem entry is xxxx, where xxxx is the 4-character name of the NetView subsystem. The first four characters in the name of the NetView subsystem procedure (PROC) and the NetView application PROC must match the subsystem name you have defined for the NetView program. The samples use CNMP. Any subsystem name you define for the NetView system can be used only for the NetView system. If you choose a subsystem name that has been previously used, re-IPL MVS before using the subsystem name. If the NetView system is started as a job rather than as a task, the job name must have the same first four characters as the subsystem interface and startup procedure.

Consider the following conditions before deciding where to place the NetView subsystem name in IEFSSNxx:

- If you place the NetView subsystem name before other subsystem names in IEFSSNxx, the NetView subsystem receives all MVS system messages and commands without any modification by the other subsystems.
- If you place the NetView subsystem name after other subsystem names in IEFSSNxx, all MVS messages and commands received by the NetView subsystem are affected by the changes made by the other subsystems listed before the NetView subsystem.

If you want information about...	Refer to...
IEFSSNxx	<i>z/OS</i> library
How to re-IPL MVS	<i>z/OS</i> library

Updating Member IEASYSxx

The update described in this section is required only if you are using one of the following:

- Status monitor
- System automation
- Cloning support or automatic restart manager (ARM) support

Determine the following:

- For the status monitor, determine the number of times you expect to stop and restart the NetView system over the life of the MVS IPL.
- For system automation, determine the number of times you expect to stop and restart the NetView subsystem interface (SSI) address space over the life of the MVS IPL.
- If both the preceding sets of conditions apply, add the values computed in those two procedures.

Increase the number of address spaces available to MVS by this value by changing the MAXUSER or RSVNONR value, or both, in the IEASYSxx member of SYS1.PARMLIB. The MVS system does not reuse any address space identifiers (ASIDs) that have established a program call interface. The NetView subsystem interface address space does reuse ASIDs that have established a program call interface. The RSVNONR value provides a way to specify replacement ASIDs for the ones lost.

Update member IEASYSxx in the SYS1.PARMLIB library as follows:

- Set MAXUSER to the number of ASIDs you want available at any one time.
- Set RSVNONR to the value you would like for replacement values.

Note: The total of the values of MAXUSER, RSVNONR, and RSVSTRT, cannot exceed 32767. If you want a low MAXUSER value, be sure to provide a reasonably large value for RSVNONR.

Add a COUPLE system parameter in member IEASYSxx to identify the COUPLExx member containing the DATA statements for the automatic restart manager (ARM). Add PLEXCFG=MONOPLEX or PLEXCFG=MULTISYSTEM for ARM support.

If you want information about...	Refer to...
IEASYSxx	<i>z/OS</i> library

Updating Member IEASYMxx

Add a SYSDEF statement in member IEASYMxx to identify user-defined system symbols for NetView, including the TCP/IP application name, RODM name, and network ID.

Setting these symbolics can alleviate modification of many of the NetView initialization members unless some default parameter such as a TCP/IP port needs to be changed.

For example, you can define the following symbolics (these are the default symbolic names used in NetView):

```
SYSDEF SYMDEF(&CNMTCPN='tcpip_name')
SYSDEF SYMDEF(&CNMRODM='rodm_name')
SYSDEF SYMDEF(&CNMNETID='network_id')
```

The initialization members that use the symbolics are shown in Table 4.

Table 4. Symbolic Usage by Initialization Members

Member	Task	TCP/IP NAME	RODM NAME	NETID
		&CNMTCPN	&CNMRODM	&CNMNETID
CNMSTYLE	NetView initialization	X	X	X
DUIGINIT	GMFHS (Graphics)		X	

If you do not want to define symbolics in IEASYMxx, you can use sample CNMSJM12 to replace the symbolics used in CNMSTYLE and DUIGINIT.

If you want information about...

Refer to...

IEASYMxx

z/OS library

Updating Member LOADxx

Add an IEASYM statement in member LOADxx to identify the IEASYMxx member to use for user-defined system symbolics. SYS1.PARMLIB member IEASYMxx must be created with the user-defined system symbolics using the MVS-defined SYSDEF statements.

If you want information about...

Refer to...

LOADxx

z/OS library

Updating Member COUPLExx

This step is for automatic restart manager (ARM) support.

The ARM couple data set must be identified to XCF by placing the following DATA statement in the SYS1.PARMLIB COUPLE member defined in member IEASYSxx:

```
DATA TYPE(ARM) PCOUPLE(primary-dsname) ACOUPLE(alternate-dsname)
```


Initiate the primary and alternate ARM couple data sets after you have created them.

If you want information about...	Refer to...
Initializing the primary and alternate ARM couple data sets	z/OS library

Updating Member SMFPRMxx

To write hardware monitor and session monitor records to the SMF log, ensure that member SMFPRMxx in SYS1.PARMLIB is set up to collect type 37 and type 39 SMF records. Hardware monitor records are SMF record type 37 and session monitor records are type 39.

Using MVS Consoles with NetView

The NetView program uses MVS consoles for receiving system messages, command responses, or both from MVS. The MVS consoles used by the NetView program are one of the following types:

- Subsystem allocatable consoles
- Extended multiple console support (EMCS) consoles

Determine the type of MVS console used by the NetView program by specifying the system message delivery mechanism on the MVSPARM statement in CNMSTYLE.

Using Subsystem Allocatable Consoles

The NetView program requires a subsystem allocatable console for each active task that can issue MVS system operator commands. The subsystem interface (SSI) has a 99-console limit and these consoles must be defined in CONSOLxx. If you are using subsystem allocatable consoles, the NetView MVS command obtains an MVS subsystem console ID for each issuing task.

Defining Subsystem Allocatable Consoles in CONSOLxx

Verify that enough subsystem consoles are defined to MVS. For each additional subsystem console that needs to be defined, add an entry in SYS1.PARMLIB (CONSOLxx) similar to the following:

```
CONSOLE DEVNUM(SUBSYSTEM),AUTH(ALL)
```

Note that there is a limit of 99 consoles. Reinitialize your MVS system for the additional console definitions to become effective.

Subsystem Allocatable Consoles in a Sysplex Environment

The NetView command prefix designator is local to the single system in the sysplex. Two NetView programs on different systems in the sysplex can use the same command prefix designator. Use the MVS ROUTE command to specify which system's NetView program should process the command.

Using EMCS Console Support

NetView provides an option to use the EMCS consoles. EMCS consoles are dynamically defined and do not have a 99 console restriction. The EMCS consoles are not defined in CONSOLxx, but can be assigned attributes. Refer to the *Tivoli NetView for z/OS Security Reference* for these attributes.

EMCS consoles can receive messages from any system in a sysplex environment. You can switch message traffic from one EMCS console to an alternate console using the RELCONID SWITCH command.

Use the AUTOTASK command to associate a NetView autotask with an EMCS console. Once this association is made, you can issue a NetView command from MVS to the NetView program. The NetView program only requires the subsystem interface when using EMCS console mode if you use subsystem designator characters (like %) instead of the MVS MODIFY command to send commands to NetView, or you use the SSIEXT option to route unsolicited messages through the SSI.

You can avoid starting the SSI for NetView operations if you use the MVS MODIFY command instead of subsystem designator characters such as the percent sign (%), and you specify MVSPARM.MSGIFAC = SYSTEM in CNMSTYLE (default value).

If you want information about...	Refer to...
The RELCONID command	<i>Tivoli NetView for z/OS Command Reference</i>

Assigning Unique EMCS Console Names

EMCS consoles are used to send commands from the NetView program to the MVS operating system and to receive messages from MVS. NetView enables you to assign a permanent, unique console ID for each operator by defining the CONSNAME attribute in the operator profile or by using the SETCONID command. If CONSNAME or SETCONID is not used, the operator can issue a GETCONID command to get a temporary console ID before each MVS command is issued. If you do not use CONSNAME or GETCONID, NetView automatically assigns a default console ID that is the same as the operator task ID.

The console ID must be unique for each operator for each MVS or each sysplex. If you do not use CONSNAME, SETCONID, or GETCONID, NetView automatically assigns a default console ID that is the same as the operator task ID. If another operator has the same ID for any system on which an operator attempts to use an EMCS console, the operator will not be allowed to use an EMCS console on that system.

To avoid these console name conflicts, use the CONSNAME attribute in the operator profiles, or the SETCONID command, to assign unique task names for each NetView operator, program, and automated operator. Ensure that each operator's ID is unique on all systems in a sysplex for which the operator might use an EMCS console. One way to create unique console IDs is to use the NetView support for the MVS cloning function. The &SYSCLONE system symbolic can be used to construct a unique CONSNAME value for each operator on each system within a sysplex.

The console names you specify in CONSOLxx are not available for use as EMCS console names. Therefore, if you specify any console names in CONSOLxx, ensure that they are different from your NetView operator IDs so that they do not conflict with the default EMCS console names.

If you want information about...	Refer to...
GETCONID, SETCONID commands	<i>Tivoli NetView for z/OS Command Reference</i>
z/OS cloning function, system symbolics	<i>z/OS library</i>

Restarting the Target System with the CLPA Option

Restart MVS with the CLPA option to use the new LPALIB members from NETVIEW.V5R1M0.SCNMLPA1 when you start NetView.

Chapter 3. Preparing NetView

This chapter describes the steps that help you prepare the NetView V5R1 program. Each step gives you explicit directions for installing an operational NetView system. Unless otherwise stated, each step is required for a fully functional NetView system.

Creating the Installation JCL Samples Library Using Job CNMSJBUP

Job CNMSJBUP (in NETVIEW.V5R1M0.CNMSAMP) copies all of the installation JCL members in NETVIEW.V5R1M0.CNMSAMP, which you have just loaded from the tape, into a data set called NETVIEW.V5R1USER.INSTALL.

The data set NETVIEW.V5R1USER.INSTALL enables you to submit installation JCL from a data set other than an SMP-managed data set. Use NETVIEW.V5R1USER.INSTALL to edit your installation JCL members as you complete the following steps.

Note: The entire NetView samples library is not copied. Only the installation members in the data set NETVIEW.V5R1M0.CNMSAMP are copied.

To create the installation JCL samples library:

1. Edit CNMSJBUP to match your system naming conventions, if necessary.
2. Run CNMSJBUP.
3. Verify your return codes before continuing with the next step.

Defining the User Catalog Using Job CNMSJ001

Job CNMSJ001 creates an ICF catalog and defines the ALIAS name NETVIEW as the high-level qualifier for the NetView data sets.

Run job CNMSJ001 if you did not define this alias name during the NetView program directory installation and you will be using this high-level qualifier. To define the user catalog:

1. Review the comments in the JCL. If you are using one of your existing catalogs, comment out the first step in CNMSJ001 and change the RELATE value in the second step accordingly. You can also change the &UNIT and &SER JCL symbolics to reflect a different DASD device type.
2. Run CNMSJ001.
3. Verify your return codes before continuing with the next step.

A return code of 12 may show a duplicate data set name exists from a previous run or release. The duplicate name may not require correction.

Allocating Partitioned Data Sets Using Job CNMSJ002

Job CNMSJ002 allocates the data sets listed in Table 5. These data sets are used by the NetView system you are installing. The data set names shown reflect the Tivoli-supplied values for &DOMAIN and &Q1.

Table 5. Data Sets for the V5R1 Program

Data Set Name	Function
NETVIEW.V5R1USER.CNM01.CNMPNL1	Contains user modifiable panels.
NETVIEW.V5R1USER.CNM01.DSIPARM	Defines NetView system definitions.
NETVIEW.V5R1USER.CNM01.DSIPRF	Defines NetView operator profiles.
NETVIEW.V5R1USER.CNM01.DSILIST	Stores the NetView reports, listings, files, and output from the security migration tool.
NETVIEW.V5R1USER.CNM01.DSIASRC	Contains the members to be used when testing the automation table. This file contains source messages and MSUs for testing the automation table.
NETVIEW.V5R1USER.CNM01.DSIARPT	Stores the output report produced from running tests of the automation table.
NETVIEW.V5R1USER.CNM01.SDSIOPEN	Stores non-protected data set members.
NETVIEW.V5R1USER.CNM01.VTAMLST	Contains VTAM source definitions for the sample network.
NETVIEW.V5R1USER.VTAMLIB	Contains VTAM load modules for the sample network (this must be an authorized data set).

Note: Add the VTAMLIB data set to the list of authorized libraries in the Authorized Program Facility (APF) member (IEAAPFxx) of SYS1.PARMLIB.

If you will be using an SAF security product, such as RACF, to define NetView operators, allocating NETVIEW.V5R1USER.CNM01.DSIPRF is not required, but it is recommended.

Also, if you plan to define operators through member DSIOPF for backup purposes, allocate the DSIPRF data set now. You can choose to deallocate this data set later.

Allocate the data sets as follows:

1. Review the symbolic variables in the comments supplied in this job. Change the &UNIT and &SER JCL symbolics to match your installation, if needed.
2. Change the &DOMAIN JCL symbolic to match the NetView domain name you are installing.
3. Run CNMSJ002.
4. Verify your return codes before continuing with the next step.

Changing the Domain and Subarea Numbers Using Job CNMSJ000

If you are installing the NetView program in subarea 01 and using domain CNM01 for network NETA, go to “Loading Members of Partitioned Data Sets Using Job CNMSJ003” on page 19.

If you are installing the NetView program in a domain other than CNM01 for network NETA, or a subarea other than 01, this job converts the NetView samples in NETVIEW.V5R1M0.CNMSAMP and NETVIEW.V5R1M0.DSIPARM and places them into NETVIEW.V5R1USER.*domain*.DSIPARM, NETVIEW.V5R1USER.INSTALL, and NETVIEW.V5R1USER.*domain*.VTAMLST.

Job CNMSJ000 also creates a VTAM APPL major node to define the NetView program to VTAM. If one of the sample network domains and subareas are chosen (CNM01, CNM02, CNM99 in NETA, or B01NV in NETB), the VTAM APPL major node already exists as part of the VTAM samples shipped on the NetView distribution tape. If the domain and subarea chosen is not one from the sample network, member NETVAPPL is created in data set NETVIEW.V5R1USER.*domain*.VTAMLST.

NETVAPPL can be used as part of your existing VTAM definitions if the resource naming convention used by NETVAPPL matches your system definitions. If the resource naming convention does not match your system definitions, you can edit NETVAPPL and your NetView definitions so that your resource names will be consistent.

To change the domain and subarea numbers:

1. Edit job CNMSJ000 and specify the domain ID (&DOMAIN) and subarea number (&SANUM) you want to use during your installation.
2. Run CNMSJ000.
3. Verify your return codes before continuing with the next step.

Note: The remainder of the steps assume that you are installing the NetView program in subarea 01 and domain CNM01. If you are installing the NetView program in a subarea other than 01 and domain CNM01, make the appropriate substitutions throughout this installation.

Loading Members of Partitioned Data Sets Using Job CNMSJ003

Job CNMSJ003 copies selected members from NETVIEW.V5R1M0.CNMSAMP, NETVIEW.V5R1USER.INSTALL, and NETVIEW.V5R1M0.SCNMUXMS into the target data sets that were previously allocated in “Allocating Partitioned Data Sets Using Job CNMSJ002” on page 18. Steps PDS3 through PDS7 are initially commented out to prevent inadvertent copying of members without checking to see if members already exist in the target PROCLIB data set. Before running these steps, ensure that the members being copied do not exist in the target data set or they will be replaced. Run the steps in job CNMSJ003 in numerical order. If you need to rerun a step, also run all appropriate steps that follow.

Except for the NetView definitions that might have been copied when CNMSJ000 was submitted, NETVIEW.V5R1USER.CNM01.DSIPARM and NETVIEW.V5R1USER.CNM01.DSIPRF should be empty. When performing the administration steps documented in this manual, copy the members you need to change from the following data sets:

- NETVIEW.V5R1M0.DSIPARM
- NETVIEW.V5R1M0.DSIPRF
- NETVIEW.V5R1M0.SDSIOPEN

to these data sets:

- NETVIEW.V5R1USER.CNM01.DSIPARM
- NETVIEW.V5R1USER.CNM01.DSIPRF
- NETVIEW.V5R1USER.CNM01.SDSIOPEN

By doing this, you maintain a copy of the original member as it was installed from the distribution tape.

Table 6. NetView JCL procedures copied to SYS1.PROCLIB

CNMSJ003 Step	IEBCOPY sample used by CNMSJ003	Procedure	Description
//PDS1 1	CNMSJI03		VTAM samples for network NETA
//PDS2 1	CNMSJI21		VTAM samples for network NETB
//PDS3 2	CNMSJI07	CNMNET	Sample VTAM start procedure
		CNMPROC	NetView start procedure
		CNMPSSI	NetView subsystem interface start procedure
		CNMPRT	Print network or trace logs procedure
//PDS4 2	CNMSJI18	EKGXRODM	RODM start procedure
		EKGLOADP	Load RODM data cache procedure
//PDS5 2	CNMSJI24	CNMNDEF	Status monitor preprocessor procedure
//PDS6 2	CNMSJI12	CNMGMFHS	GMFHS start procedure
//PDS7 3	CNMSJI23	IHSAEVRT	E/AS start procedure

where:

- 1** Copies members from NETVIEW.V5R1M0.CNMSAMP to NETVIEW.V5R1USER.CNM01.VTAMLST.
- 2** Copies members from NETVIEW.V5R1USER.INSTALL to SYS1.PROCLIB.
- 3** Copies members from NETVIEW.V5R1M0.SCNMUXMS to SYS1.PROCLIB.

To copy members:

1. Edit CNMSJ003 to match your system naming conventions, if necessary. You may want to change the specification for the &Q2 symbolic to use a different high-level qualifier for PROCLIB.
2. Run CNMSJ003.
3. Verify your return codes before continuing with the next step.

Allocating VSAM Clusters Using Job CNMSJ004

Job CNMSJ004 defines the VSAM clusters for various NetView facilities. Table 7 on page 21 lists the data set names associated with these facilities, and the names of members containing VSAM cluster information for the data sets.

Note: The records needed for the system to view databases as active data sets are added during NetView component initialization.

Before allocating VSAM clusters, follow these steps:

1. Review CNMSID01 and each of the following VSAM allocate members in Table 7 on page 21. These sample input members provide the name, volume, catalog, and password specifications for the VSAM clusters that get allocated

and deleted.

Table 7. VSAM Clusters for NetView Facilities

NetView Facility	VSAM Allocate Member	Data Sets	VSAM Delete Member
Network log	CNMSI101	NETVIEW.CNM01.DSILOGP NETVIEW.CNM01.DSILOGS	CNMSID01
Trace log	CNMSI101	NETVIEW.CNM01.DSITRCP NETVIEW.CNM01.DSITRCS	CNMSID01
Session monitor	CNMSI201	NETVIEW.CNM01.AAUVSPL NETVIEW.CNM01.AAUVSSL	CNMSID01
Hardware monitor	CNMSI301	NETVIEW.CNM01.BNJLGPR NETVIEW.CNM01.BNJLGSE	CNMSID01
4700 Support Facility	CNMSI401	NETVIEW.CNM01.BNJ36PR NETVIEW.CNM01.BNJ36SE	CNMSID01
Central site control facility	CNMSI501	NETVIEW.CNM01.DSIKPNL	CNMSID01
Save/restore	CNMSI601	NETVIEW.CNM01.DSISVRT	CNMSID01

Note: All members referenced by CNMSJ004 for deleting or allocating VSAM data sets are found in NETVIEW.V5R1USER.INSTALL.

2. Ensure these values are correct for your system and that the VSAM cluster names match the names you use in your NetView startup procedure, CNMPROC.
3. If you want to define a security password for the VSAM databases, add the password to the CNMSTPWD member (CNMSTYLE %INCLUDE member) in DSIPARM. Also add this password to the VSAM allocate members (CNMSI101, CNMSI201, and so forth).

Note: Specify a password on the catalog that contains the entry for the data set before the system verifies the password for that data set.

4. If you want to allocate RODM VSAM clusters at this time, follow these steps:
 - a. Review EKGSI01 and each of the following VSAM allocate members in Table 8. These sample input members provide the name, volume, catalog, and password specifications for the VSAM clusters that get allocated and deleted for RODM.

Table 8. VSAM Clusters for RODM Facilities

RODM 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

- b. Ensure these values are correct for your system and that the VSAM cluster names match the names you use in your RODM startup procedure, EKGXRODM.

If you add additional data window checkpoint data sets, ensure that you have defined them to the RODM startup procedure, EKGXRODM (similar to EKGCK001 and EKGCK002).

- c. Uncomment //STEP9 EXEC and //STEP10 EXEC in CNMSJ004. This will cause the RODM VSAM clusters to be allocated when you run sample job CNMSJ004.

Note: If you are unfamiliar with allocating RODM VSAM clusters, and would prefer to allocate them at a later time, use sample job EKGSI004. For additional information, refer to *Tivoli NetView for z/OS Installation: Configuring Graphical Components*.

To allocate VSAM clusters, follow these steps:

1. The supplied default volume for the NetView VSAM clusters is VOL(CPDLB2). If your volume is not CPDLB2, change each of the VSAM allocate members shown in Table 7 on page 21 (and in Table 8 on page 21 if allocating RODM VSAM clusters) to a valid volume. Review the VSAM allocate members (CNMSI101, CNMSI201, and so forth) to ensure the correct volume is specified for each VSAM cluster to be allocated.

Notes:

- a. If you are deleting your existing VSAM clusters, uncomment //STEP1 and review CNMSID01 to make sure you delete the correct VSAM cluster names.
 - b. If you are deleting your existing RODM VSAM clusters, uncomment //STEP2 and review EKGSI001 to make sure you delete the correct VSAM cluster names.
2. Run CNMSJ004.
 3. Verify that all steps ran correctly before continuing with the next step.

To rerun CNMSJ004:

1. Delete the asterisk (*) following the slashes (//) in the //STEP1 statement of the job (//STEP2 for RODM VSAM clusters). Deleting the asterisk deletes any previously allocated data sets.
2. Modify CNMSID01 (EKGSI001 for RODM VSAM clusters) to reflect your system naming conventions.
3. Review the members referenced in the //STEP3–//STEP8 statements for any VSAM data set name changes for NetView. Review the members referenced in the //STEP9–//STEP10 statements for any VSAM data set name changes for RODM.

If you are operating in an environment such as an SMS environment and receive an open error such as message DSI556I with a return code of X'08' and an ACB error field value of X'A8', you can delete and redefine the database, moving it to a different volume. To accomplish these tasks:

1. Free the database.
2. Delete the database.
3. Reallocate the database.

You can use the NetView FREE and ALLOCATE commands to free and reallocate the database.

Loading Sample Databases for Verification Using Job CNMSJ005

Job CNMSJ005 loads sample data into the VSAM data sets that were allocated in “Allocating VSAM Clusters Using Job CNMSJ004” on page 20. The sample data helps you verify that the NetView program is installed properly. The verification is done in “Chapter 7. Verifying the Installation” on page 55.

Note: This step is optional. If you did not run “Allocating VSAM Clusters Using Job CNMSJ004” on page 20 or if you do not wish to see test data displayed during verification, go to “Chapter 4. Preparing VTAM” on page 25.

When you run this job, make sure the DOMAIN symbolic parameters properly refer to your system’s VSAM data set names.

To load the sample databases:

1. Run CNMSJ005.
2. Verify your return codes.

If you need to rerun CNMSJ005 for any reason, first rerun the job you ran in “Allocating VSAM Clusters Using Job CNMSJ004” on page 20 to delete and reallocate the NetView databases.

Chapter 4. Preparing VTAM

To define the NetView program to VTAM, you need to:

- Modify the application (APPL) major node.
- Review VTAM and NCP definitions for mode table and unformatted system service (USS) table changes.
- Review VTAM start options.

Modifying the Application (APPL) Major Node

The APPL statements for the NetView tasks are located in A01APPLS (CNMS0013).

Add this member to your VTAM configuration list so it is included by the status monitor preprocessor and started. The VTAM configuration list used in the samples is ATCCON01 (CNMS0003). You might need to change some of these statements or rename this member.

If you change the APPL statement names, add the ACBNAME operand to each statement you are changing. The ACBNAME must be the original IBM-supplied APPL name of the statement being changed. For example, you could rename:

```
DSICRTR APPL AUTH=CNM,PRTCT=CNM01
```

to

```
HN2CRTR APPL AUTH=CNM,PRTCT=CNM01,ACBNAME=DSICRTR
```

Exception Notes:

1. You cannot change the corresponding task ID on the TASK definition in member CNMSTASK in DSIPARM.
2. All APPL statements with the CNM01 name prefix must follow the existing DOMAINID constraints.
3. An APPL name prefixed with CNM01 cannot have its suffix changed. For example, CNM01LUC must retain the LUC suffix.
4. Verify the original APPL name with the TASK statement in CNMSTASK. The MEM operand specifies the startup initialization member in DSIPARM.
5. Command help panels reference the original APPL names.
6. You cannot change BNJHWMON because of its downward compatibility with the BNJDSESV task.
7. You cannot change ALIASAPL because it is necessary for the VTAM Alias Name Translation function to work.
8. The ACBNAME keyword for CNM01VPD in A01APPLS must match the name in DSIVPARM.

Defining the NetView Management Console to VTAM (LU 6.2 only)

You can use TCP/IP or LU 6.2 to communicate between NetView and the NetView management console. If you use LU 6.2, the following APPL statement defines both the NetView system and the NetView management console:

```

CNM01  APPL  AUTH=(VPACE,ACQ,PASS),PRTCT=CNM01,          X
          MODETAB=AMODETAB,DLOGMOD=DSIL6MOD,            X
          APPC=YES,PARSESS=YES,                          X
          DMINWNL=4,DMINWNR=4,DSESLIM=8,VPACING=10,     X
          AUTOSSES=2

```

Defining the SNA Topology Manager to VTAM

VTAM needs APPL definitions for the SNA topology manager application. Sample definitions are included in member A01APPLS (CNMS0013). The APPL name for the SNA topology manager must match the APPLNAME specified in the VTAM section of the FLBSYSD initialization file used by the manager application.

If you want information about...	Refer to...
VTAM requirements	the description of the VTAM MIBConnect parameters contained in the FLBSYSD initialization file or refer to <i>Tivoli NetView for z/OS SNA Topology Manager Implementation Guide</i>

Defining the VTAM Primary Program Operator Interface Task

The primary program operator interface (POI) task is defined with the statement:

```

CNM01PPT APPL  AUTH=(NVPACE,PPO),PRTCT=CNM01,EAS=1,          X
          MODETAB=AMODETAB,DLOGMOD=DSILGMOD
*          STATOPT='NETVIEW PPT'

```

VTAM does not open more than one access method control block with AUTH=PPO specified. If you are migrating and running a previous release of the NetView program concurrently with the NetView program for verification or testing, change this statement from AUTH=(NVPACE,SPO) back to AUTH=(NVPACE,PPO).

Changing the Password

The original password on the ACBpassword keyword in CNMSTPWD (%INCLUDEd by CNMSTYLE) is CNM01. If you change this password, change *every* occurrence of PRTCT in A01APPLS (CNMS0013) to the same value. For example, if you change the password to PW006, then change:

```

CNM01000 APPL AUTH=(NVPACE,SPO,ACQ,PASS),PRTCT=CNM01,EAS=4,  X

```

to:

```

CNM01000 APPL AUTH=(NVPACE,SPO,ACQ,PASS),PRTCT=PW006,EAS=4,  X

```

Changing the Domain Name (DOMAINID)

The original DOMAINID or value on the DOMAIN keyword in CNMSTYLE is CNM01. If you change this DOMAINID, change *every* occurrence of CNM01 in A01APPLS (CNMS0013) to the current DOMAINID, except for the password. For example, if you changed the DOMAINID to CNM12, then change:

```

CNM01000 APPL AUTH=(NVPACE,SPO,ACQ,PASS),PRTCT=PW006,EAS=4,  X

```

to:

```

CNM12000 APPL AUTH=(NVPACE,SPO,ACQ,PASS),PRTCT=PW006,EAS=4,  X

```

Note: If you code the optional ACBNAME operand on the APPL statement, it must match the APPL name in column 1. Therefore, if you change the DOMAIN name, you must also change ACBNAME.

Changing the Logmode Table (LOGMODE)

The sample network has a logmode table named AMODETAB (CNMS0001) that includes logmode entries for NetView sessions. In A01APPLS (CNMS0013), the DLOGMOD operand for an application points to an entry in this logmode table. For example, the operands for the application CNM01000 are:

```
MODETAB=AMODETAB,DLOGMOD=DSILGMOD
```

Check the DLOGMOD operands for your applications and ensure that they are pointing to the proper entries for your installation.

Defining Resources to the Status Monitor (STATOPT)

The sample A01APPLS (CNMS0013) has STATOPT statements defining many of the resources to the status monitor. For example, the statement for the application CNM01000 is:

```
* STATOPT='NETVIEW 000'
```

If you want to redefine the name of this resource, or any other resource, change the STATOPT statement for it. Read about the STATOPT statement before you change these statements.

Defining Multiple Concurrent NetView Operators

You can define tasks for multiple concurrent NetView operators using APPL statements, for example:

```
CNM01000 APPL AUTH=(NVPACE,SPO,ACQ,PASS),PRTCT=CNM01,EAS=4, X
          MODETAB=AMODETAB,DLOGMOD=DSILGMOD
*          STATOPT='NETVIEW 000'
```

You must specify an APPL statement for each concurrent NetView operator. Each APPL statement name contains a 3-character suffix which is a hexadecimal number in the range of 000–FFF. If you are starting autotasks before VTAM, you must have enough consecutively numbered APPL statements to allow each autotask to obtain one. The numbers after those reserved for the autotasks do not need consecutive suffixes.

To enable takeover or reconnect, include PASS value in your AUTH= definition. If PASS is not specified and a takeover or reconnect is attempted, the authorized receiver will receive message DSI133I, and the operator attempting to log on will receive the following message:

```
DSI213I ACCESS TO 'CLSDST PASS' IS NOT AUTHORIZED
```

Note: The maximum number of concurrent NetView users cannot exceed 4096.

This number represents the sum of the following:

- The total number of LU names specified on all HARDCOPY statements
- The total number of concurrent autotasks
- The total number of concurrent logons from VTAM, NetView 3270 management console, the Web browser, and NMC

Using the REFRESH command, refresh DSIOPF to add or delete NetView operators and operator profiles and to change operator profile definitions while the NetView program is running. Changes to operator profile information (either NetView profiles or SAF NETVIEW segment) require the operator to log off and log back on for the changes to take effect.

If there are not enough APPL statements available for dynamically added operators after NetView is started, create a new APPL major node similar to A01APPLS. In this new member you can define as many additional APPL statements as you need for new operators. You do not have to stop and restart the NetView program to use the new operators you defined. When you start this new major node, the application definition statement for the new operator becomes available for immediate use. If you are using alert forwarding, you need to define additional APPL statements.

Allowing for Additional TAF LU1 Operators

Tasks for five concurrent terminal access facility (TAF) users of operator-control sessions are defined with APPL statements. The first one is:

```
TAF01000 APPL MODETAB=AMODETAB,EAS=9, X
          DLOGMOD=M3767
*          STATOPT='TAFAPPL 000'
```

If you have more than five concurrent TAF users of operator control sessions, you need to add APPL statements.

Allowing for Additional TAF LU2 Operators

Tasks for 20 concurrent TAF users of full-screen sessions are defined with APPL statements. The first one is:

```
TAF01F00 APPL MODETAB=AMODETAB,EAS=9, X
          DLOGMOD=M2SDLCNQ
*          STATOPT='TAFUSER 000'
```

If you have more than 20 concurrent TAF users of full-screen sessions, you need to add APPL statements.

Defining Alias Names

If you are using cross-network communication, you might need to use alias names to resolve duplicate resource names. With alias names, the name of the resource (such as a logical unit, a class of service, an SRCLU, or a logon mode table) from the sending network is translated to a name that is unique to the receiving network.

If you want to use the alias name translation facility provided by the NetView program, include an APPL definition statement for that application program in the application program major node. The name on the APPL definition statement must be ALIASAPL.

The example of the APPL statement in A01APPLS is:

```
ALIASAPL APPL AUTH=(CNM),PRTCT=CNM01
*          STATOPT='ALIAS TASK'
```

The AUTH=CNM operand shows that this application program can use the communication network management (CNM) interface. The PRTCT=CNM01 operand specifies the NetView program password.

Member DSIALATD contains the statement:

```
LABEL    ALIASMEM DSIALTAB
```

This statement points to DSIALTAB, which contains examples of alias translation statements.

To add an alias data set member that defines one of your networks, add the member name to the statement in DSIALATD as follows:

```
LABEL ALIASMEM DSIALTAB, YOURMEM
```

Create a data set member of the same name following the format of DSIALTAB to define your alias translations. You can add more than one member name to the same ALIASMEM statement. Ensure that you either modify DSIALTAB with valid aliases or comment out the example statements provided.

Note: Do not use alias names to translate NetView APPL names, especially the domain ID and any names that are constructed from it; for example, A01M, A01M001, A02M, and A02M002.

If you want information about...	Refer to...
The Alias Name Translation Facility	The VTAM library

DSIALTAB provides examples of the following statements. Modify these statements for your own environment.

ORIGNET Statement

For each member name you code on an ALIASMEM statement, code at least one ORIGNET definition statement for each network that requires alias name translations. The format of the ORIGNET statement is:

```
ORIGNET netid
```

Where:

netid Is the network name as coded on the NETWORK statement for the CDRM major node, or on the VTAM NETID start option. This name always represents the network that knows the resource by its alias name.

An example of this statement is:

```
ORIGNET NETA
```

LU Statement

If you have LUs with the same name in more than one network, code the LU name translation statement. With this name translation, the LU can be known in its owning network by its real name and in other networks by the assigned alias name. The format of the LU statement is:

```
LABEL LU realname,ownernet,aliasname[,ownersscp]
```

Where:

realname

Is the LU name as it is defined in the domain where the LU resides.

ownernet

Is the 1–8 character network name of the network where the LU is known by its *realname*.

aliasname

Is the alias name for this LU that is used by the network specified on the ORIGNET definition statement.

ownersscp

Is the SSCP that owns the LU in the *ownernet* network. If you do not code *ownersscp*, the SSCP name is taken from the VTAM CDRSC definition statement.

An example of this statement is:

```
ORIGNET NETA
LABEL LU TERM1,NETB,TERM1B
```

Class of Service Statement

In interconnected networks, the class-of-service (COS) names used in one network can be different from the names used in another network. However, two COS definitions can provide the same type of service. For example, NETA calls its fastest class of routes FAST, while NETB calls its fastest class of routes QUICK. The COS names are different, but the result in each case is the same: the fastest route is chosen. With alias names, you can make the COS name used in one network equivalent to a name in another network. Each network can use the name it knows and still get the class of service it expects.

The format of the COS statement is:

```
LABEL COS adjacnme,adjacnet,localnme
```

Where:

adjacnme

Is the COS name defined in the adjacent or destination network that provides the class-of-service equivalent to the COS entry specified in the local-name class of service.

adjacnet

Is the 1–8 character NETID of the adjacent network where the equivalent class of service is defined.

localnme

Is the name of the class of service defined in the network specified on the ORIGNET definition statement.

An example of this statement is:

```
ORIGNET NETA
LABEL COS QUICK,NETB,FAST
```

You can use the same local name (or alias) within one network for different adjacent networks. Also, each adjacent name can be given any number of aliases or local names.

MODE Statement

In interconnected networks, the set of logon mode entry names used in one network can be different from the names used in another network. However, two logon mode entries with different names can result in the session parameters being the same. With the name translation facility, you can make a logon mode entry name used in one network equivalent to a name in another network. In this way, you can access the information in a logon mode table from another network without changing the logon procedures used in either network.

The format of the MODE statement is:

```
LABEL MODE destname,destnet,localnme
```

Where:

destname

Is the name of the logon mode entry defined in the destination network.

destnet

Is the 1- to 8-character NETID of the destination network where the logon mode entry is known by its equivalent name.

localnme

Is the name of the logon mode entry defined in the network specified on the ORIGNET statement.

An example of this statement is:

```

ORIGNET NETA
LABEL  MODE LOG1B,NETB,LOG1A
ORIGNET NETB
LABEL  MODE LOG1A,NETA,LOG1B

```

The logon mode entry name defined to NETA as LOG1A is known in NETB as LOG1B. The logon mode entry name defined to NETB as LOG1B is known in NETA as LOG1A.

For more information on rules for name translation, refer to the VTAM library.

Reviewing VTAM and NCP Definitions for Mode Table and USS Table Changes

The sample definitions include the A01SNA (CNMS0073) statement for SNA locally-attached terminals, the A01LOCAL (CNMS0016) statement for non-SNA locally-attached terminals, and your VTAM and NCP definitions for any mode table changes. Update your USS table for the NetView program and reflect that change in your definitions. The USS table is AUSSTAB (CNMS0011) in the samples.

Note: If you modify AUSSTAB, run CNMSJ006 to reassemble and link-edit the table and carry out the changes.

Assembling VTAM Tables Using Job CNMSJ006

Job CNMSJ006 assembles and link-edits the following tables and places them in NETVIEW.V5R1USER.VTAMLIB:

- AMODETAB, which is the logmode table supporting miscellaneous devices defined in the sample network
- AUSSTAB, which is the USS table supporting SNA synchronous data link control
- ISTSDCOS, which is the class-of-service table containing routes.

To assemble VTAM tables:

1. Edit CNMSJ006.

Notes:

- a. The ISTMGC00 table is included with VTAM as part of the ISTMGC01 table. The ISTMGC01 table is a default routing table included with VTAM. This table tells VTAM to route unsolicited request units (RUs) to the NetView program so they can be received and routed to components such as the hardware monitor and the session monitor.
- b. Verify that the mode entries for DSIL6MOD, PARALLEL, and RBRIDGE from CNMS0001 (AMODETAB) are in your current AMODETAB member in VTAMLST.

- c. The sample job CNMSJ006 provided in &PGMNAME uses the High Level Assembler ASMA90. To use Assembler H, change the program name to PGM=IEV90.
2. Run CNMSJ006.
3. NETVIEW.V5R1USER.CNM01.VTAMLST has two members that define local terminal support:
 - A01LOCAL (CNMS0016) for non-SNA channel-attached terminals
 - A01SNA (CNMS0073) for SNA channel-attached terminals

Verify that in the member you use, each CUADDR parameter is the correct address for your host local terminals. Also, be sure that the node name references these addresses. These members reference the AMODETAB and AUSSTAB tables.

4. Verify your return codes before continuing with the next step listed for your version and release.

The changes to the tables take effect the next time you start VTAM or issue the VTAM command:

```
MODIFY NET, TABLE, NEWTAB=AMODETAB, OPTION=LOAD
```

If you want information about...	Refer to...
The MODIFY command	The VTAM library

Reviewing VTAM Start Options

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

NetView supplies a sample VTAM startup member CNMS0010 which is copied to NETVIEW.V5R1USER.CNM01.VTAMLST as ATCSTR00. This sample specifies PPOLOG=NO because the PPOLOG=YES statement is not needed to keep the status monitor panels accurate. However, if you want all VTAM messages kept in the NetView log, you should do one of the following:

- Specify PPOLOG=YES in ATCSTR00. If you choose this approach, ensure that any automation table statements for VTAM messages do not copy the automated messages to the NetView log by specifying NETLOG(YES). Also note that if you have automation statements for VTAM messages in your automation table, unsolicited VTAM messages from the SSI that match those statements might also be sent to the NetView log. This results in duplicate entries in the NetView log for VTAM commands issued outside the NetView program.
- Specify PPOLOG=NO and force the unsolicited VTAM messages to the NetView log using the automation table or the ASSIGN command. If you only specify PPOLOG=NO, VTAM commands issued outside the NetView program are not recorded in the NetView log unless there is a match in the automation table or an ASSIGN command for the message.

If you want information about...	Refer to...
Start options	The VTAM library
VTAM requirements	the description of the VTAM MIBConnect parameters contained in the FLBSYSD initialization file or refer to <i>Tivoli NetView for z/OS SNA Topology Manager Implementation Guide</i>

If you want information about...	Refer to...
VTAM CMIP services	the VTAM library
ISTCMIP, ACYGDMO, and ISTASN1 statements	the VTAM library

Modifying the VTAM Start Procedure

If you are using the sample VTAM start procedure CNMNET (CNMSJ008), go to “Creating the Installation JCL Samples Library Using Job CNMSJBUP” on page 17.

Complete this section to ensure that the status monitor is properly initialized. The changes are not in effect until you start VTAM.

Concatenate NETVIEW.V5R1USER.CNM01.VTAMLST and NETVIEW.V5R1USER.VTAMLIB to the VTAMLST and VTAMLIB DD statements in your production VTAM start procedure before SYS1.VTAMLST and SYS1.VTAMLIB, respectively. Sample CNMSJ002 allocates these data sets, see “Allocating Partitioned Data Sets Using Job CNMSJ002” on page 18.

Note: The VTAMLST that is shipped with NetView contains data set members that you may have already customized. Be careful to protect any members you have already customized when performing this concatenation. Otherwise, VTAM will not perform as you expect it to.

In addition, you should have already used the Authorized Program Facility (APF) to authorize SCNMLNK1. If you have not done so, see “Updating Member IEAAPFxx” on page 10 and follow the instructions there before continuing.

Add the following NetView data set to the VTAMLIB statements in your VTAM start procedure:

```
//          DD DISP=SHR,DSN=NETVIEW.V5R1M0.SCNMLNK1
```

The new VTAMLIB statements should then look similar to:

```
//VTAMLIB  DD DISP=SHR,DSN=&SQ1..VTAMLIB
//          DD DISP=SHR,DSN=SYS1.VTAMLIB
//          DD DISP=SHR,DSN=NETVIEW.V5R1M0.SCNMLNK1
```

Starting VTAM ACBs and Corresponding NetView Tasks

If a VTAM ACB and a corresponding NetView task are both active, and the ACB is inactivated and then reactivated, the NetView task usually becomes active. Some NetView tasks remain active when you terminate and restart VTAM without bringing NetView down. Other tasks become inactive but reactivate automatically once you restart VTAM.

DSICRTR, DSIGDS, BNJDSESV, ALIASAPL, and AAUTCNMI are examples of NetView tasks that remain active when the corresponding ACB becomes inactive.

VPDTASK, DSIAMLUT, CNM01VMT, CNM01BRW, and CNM01LUC are NetView tasks that become inactive when the corresponding ACB becomes inactive, but reactivate automatically once you restart VTAM. If you use a VTAM VARY INACT command to inactivate the ACB, you need to restart these NetView tasks after the ACB is reactivated. If you use a Z NET,QUICK command to inactivate the ACB, these NetView tasks are automatically reactivated when VTAM is restarted.

When NetView recognizes that VTAM is active, it invokes CLIST CNMEVTAM. This runs under the PPT. At this time the PPT ACB and the main NetView ACB are open.

Chapter 5. Getting Ready to Start NetView

When you start the NetView program, you use two START procedures, one for the NetView application (CNMPROC (CNMSJ009)) and one for the NetView subsystem (CNMPSSI (CNMSJ010)). It does not matter which you start first.

NetView and Subsystem Application Procedures

Review the copies of CNMPROC (CNMSJ009) and CNMPSSI (CNMSJ010) supplied with the V5R1 samples for the following considerations:

- The name of the PROCLIB member and the PROC statement must begin with the 4-character subsystem name you have defined for running the NetView program. The associated CNMPSSI (CNMSJ010) start procedure must also begin with the same subsystem name. CNMP is used in the sample network.

Note: If the PROCLIB member name matches an entry in IEFSSN xx , use the SUB= parameter with the START command to specify a subsystem other than the MASTER subsystem. Specify a subsystem where SYSIN and SYSOUT are not supported.

- If you start a second copy of the NetView program in the same host, you must use a procedure name that begins with a 4-character subsystem name that is different from the one you have already started.

Note: Remember to add any 4-character subsystem name to the IEFSSN xx member in SYS1.PARMLIB.

- If you plan to start the NetView application PROC under the master subsystem before you start JES, see “Starting the NetView Program Before Starting JES” on page 49.

Modifying the NetView Subsystem Interface Procedure

You can adjust the symbolic parameters in the sample CNMPSSI (CNMSJ010) procedures to meet your own installation requirements:

ARM Allows you to enable the NetView subsystem for MVS automatic restart management (ARM) and to supply a name (up to 16 characters) by which this NetView subsystem will be known to ARM.

The values for the CNMPSSI ARM parameter are:

***ARM** Specifies that the NetView subsystem is to be enabled. NetView will generate a NetView ARM element name.

name Specifies that the NetView subsystem is to be enabled and identifies the NetView ARM element name. This name can be 1 to 16 alphanumeric characters in length but cannot begin with a numeric character. It can also contain the special characters #, @, and \$.

***NOARM**

Specifies that the NetView subsystem is not to be enabled.

*NOARM is the default.

CBUF=200

Specifies the number of command buffers to be allocated in the NetView

subsystem address space for the command buffer queue to hold the NetView commands and command lists (entered from the MVS console) for asynchronous processing by the NetView program. If these buffers are exhausted, the NetView program suspends command buffering until buffers become available.

The size of the buffer is 256 bytes, and the required minimum number of command buffers is 200. The maximum number of command buffers is 8000000.

DSIG="

Specifies up to 8 characters to be used as the NetView subsystem command designator. It must precede all NetView commands and command lists that are issued from an MVS console to distinguish them from other OS/390 commands. The default value of null causes the subsystem to use the 4-character subsystem name as the prefix.

If you are running two NetView programs in the same host, the subsystem start procedures for the second copy must specify different characters from the ones used for the first copy. The designator must also be different from the one used by JES or any other subsystem. If multiple subsystems in the same host use the same command designator, commands from MVS consoles starting with that character are passed to each of the subsystems to be processed.

You can register the prefix with the OS/390 system on which the job runs or with the sysplex. To do this, use the PFXREG option.

MBUF=4000

Specifies the number of message buffers to be allocated in the NetView subsystem's buffer address space, to hold WTO messages and DOM requests for asynchronous processing by the NetView program. If these buffers are exhausted, the NetView program suspends message buffering until buffers become available.

The size of a message buffer is 256 bytes, and the minimum number of buffers you can specify is 200. The maximum number of message buffers is 8000000.

MSGIFAC

Allows you to specify whether to use the MVS extended console support or the NetView subsystem interface for OS/390 messages.

Notes:

1. The MVS extended console support is the default. To use the MVS extended console support, you do not need to make any changes.
2. Ensure that the MVSPARM.MSGIFAC statement in CNMSTYLE is consistent with the MSGIFAC= parameter that you specify in CNMPSSI.

The values for the CNMPSSI MSGIFAC parameter are:

SYSTEM

Specifies that the OS/390 system level determines whether the NetView program uses the subsystem interface (SSI) or extended EMCS consoles for OS/390 messages. SYSTEM is the default.

The NetView program uses EMCS consoles for OS/390 messages and the SSI for the command interface.

Note: The SSI is still used for NetView commands flowing in from MVS consoles.

If you specify SYSTEM, be sure to also specify MVSPARM.MSGIFAC=SYSTEM in CNMSTYLE.

Similarly, if you specify MVSPARM.MSGIFAC=CMDONLY in CNMSTYLE, be sure to specify MSGIFAC='SYSTEM' in the NetView subsystem interface start procedure, CNMPSSI (CNMSJ010).

USESSI

Specifies that the NetView program uses the SSI for command and message interfaces, regardless of the OS/390 system level. The NetView program provides this option so that you can control migration of automation from the SSI to EMCS consoles. Specify USESSI until you can plan and coordinate the automation impacts of EMCS consoles.

If you specify USESSI, be sure to also specify MVSPARM.MSGIFAC=USESSI in CNMSTYLE.

NOSSI

Specifies that the NetView subsystem address space will not have command or message interfaces. You use this option to start a NetView subsystem address space for the primary program operator interface (PPI). When you specify NOSSI, you also need to specify PPIOPT=PPI in CNMPSSI (CNMSJ010) if the NetView system was started with the same subsystem identifier as the PPI.

Note: If you are not using EMCS consoles and you specify NOSSI, the task with the load module name CNMCSSIR has no function. Commenting out the CNMCSSIR task statement can improve performance.

If you specify NOSSI, be sure to also specify MVSPARM.MSGIFAC=SYSTEM in CNMSTYLE.

QUESSI

Specifies that the NetView program uses the SSI for command and message interfaces, regardless of the OS/390 system level. The CNMCSSIR task uses the subsystem interface to route solicited and unsolicited OS/390 messages, and commands entered from MVS consoles, to the NetView program. Also, it requests messages be queued to the SSI address space while NetView is inactive.

If you specify QUESSI, be sure to also specify MVSPARM.MSGIFAC=QUESSI in CNMSTYLE.

QSSIAT

Specifies that the NetView program uses the SSI for command and message interfaces, regardless of the OS/390 system level. The CNMCSSIR task uses the subsystem interface to route solicited and unsolicited OS/390 messages, and commands entered from MVS consoles, to the NetView program. Also, it requests messages be queued to the SSI address space while NetView is inactive. The MPF auto token position 8 is overlaid with an @ character when a message is queued, while NetView is inactive, to show potentially old messages.

If you specify QSSIAT, be sure to also specify MVSPARM.MSGIFAC=QSSIAT in CNMSTYLE.

SSIEXT

Specifies that the NetView program uses the SSI for command and message interfaces, regardless of the OS/390 system level. The CNMCSIR task uses the subsystem interface to route solicited and unsolicited OS/390 messages, and commands entered from MVS consoles, to the NetView program. Also, it requests messages be queued to the SSI address space while NetView is inactive. The MPF auto token position 8 is overlaid with an @ character when a message is queued, while NetView is inactive, to show potentially old messages.

Additionally, with this option, EMCS consoles are obtained when operators or autotasks enter OS/390 commands (similar to the SYSTEM value). With this option, unsolicited MVS message traffic is routed using the SSI, and it can be queued while NetView or the CNMCSIR task is not active. Solicited (command response) messages are processed using EMCS and can be used in NetView pipelines for all OS/390 commands supporting 4-byte console ids and command and response token (CART).

If you specify SSIEXT, be sure to also specify MVSPARM.MSGIFAC=SSIEXT in CNMSTYLE.

Note: There are two MSGIFAC parameters for the NetView program. You can find these parameters in the:

- NetView subsystem startup procedure, CNMPSSI
- MVSPARM statement in CNMSTYLE

Ensure that the two statements are compatible. Refer to *Tivoli NetView for z/OS Automation Guide* for details on the valid combinations.

PFXREG

Enables you to register the NetView command prefix to a single system, an entire sysplex, or specify the prefix is not to be registered. The values for the PFXREG parameter are:

- ONE** Registers the NetView command prefix with the OS/390 system on which the job runs. If a conflict is found with a previously registered prefix, the job ends. ONE is the default.
- ALL** Registers the NetView command prefix with the sysplex. This enables you to use the command prefix anywhere in the sysplex and the command will be routed to this subsystem.
- NO** Does not register the NetView command prefix. If the prefix is a duplicate and the job starts, multiple subsystems could receive the same commands. Use this option if you are running a level of OS/390 that does not support command prefix registration or if you want to send all commands to multiple NetView systems.

PPIOPT

Specifies whether you want to initialize the PPI facility. The values for the PPIOPT parameter are:

- PPI** Specifies that the PPI facility is to be started and initialized for the NetView subsystem address space. You cannot request this option

for more than one subsystem address space. If the PPI is already active on another subsystem address space, it does not initialize for additional requests. PPI is the default.

NOPPI

Specifies that the PPI facility is not initialized for the subsystem address space.

P256BUF

Specifies the number of 256-byte PPI buffers. The default is 300.

P4000BUF

Specifies the number of 4000-byte PPI buffers. The default is 0.

REG=1250

Specifies the region size for the NetView subsystem's buffer address space in kilobytes (the number specified times 1024).

The minimum value allowed is 200 plus 0.25 times the total number of messages and command buffers specified by MBUF and CBUF. The default values allow for 4200 total message and command buffers. To calculate the correct region size for your network, refer to *Tivoli NetView for z/OS Tuning Guide*.

REG=1250

Specifies the region size for the NetView subsystem's buffer address space in kilobytes (the number specified times 1024).

The minimum value allowed is 200 plus 0.25 times the total number of messages and command buffers specified by MBUF and CBUF. The default values allow for 4200 total message and command buffers. To calculate the correct region size for your network, refer to *Tivoli NetView for z/OS Tuning Guide*.

If you are running two NetView programs on the same system, refer to *Tivoli NetView for z/OS Installation: Configuring Additional Components*.

Modifying the NetView Startup Procedure

CNMPROC (CNMSJ009) was copied to the PROCLIB when you loaded partitioned data sets during installation. Make the following changes to the NetView startup procedure (CNMPROC). These changes do not involve CNMPSSI (CNMSJ010).

Specifying the Japanese Version

If you are installing the Japanese version of the NetView program, uncomment the appropriate statements in CNMPROC and comment out the corresponding English statements. Refer to the comments in CNMPROC.

Setting System Variables

You can set several system variables in CNMPROC.

The start procedure contains the specification of a two-character identifier NV2I. This value becomes a local system symbolic &NV2I, and is used to construct names that need to be unique to each NetView within an MVS image, a sysplex, or a network. If no value is specified for &NV2I, its value defaults to NM.

If you specify a value for &NV2I (xx), NetView will read CxxSTYLE in DSIPARM for initialization parameters. If this member is not found, NetView will read CNMSTYLE instead.

Notes:

1. If you will only have one NetView for each OS/390 system, you might want to consider using the value of the &SYSC clone symbolic for the value of &NV2I.
2. For more information, refer to the comments in CNMPROC (CNMSJ009).

Setting the Dispatch Priority

The procedure to start the NetView applications contains the following parameter:

```
//          DPRTY=(13,13)
```

Set the DPRTY parameter to the *highest dispatching priority possible* below the priority set for VTAM to get the best performance and to help prevent records from being overlaid.

Modifying the Buffer Size and Slot Size

The following sample statements specify the region size, buffer size, and slot size:

```
//          REG=32768,      ** REGION SIZE(IN K) FOR MAIN TASK
//          BFSZ=24,       ** BUFFER SIZE(IN K)
//          SLSZ=200       ** SLOT SIZE
```

If you are running AON or SA tower in CNMSTYLE, the region size should be increased to 65536. For more information on calculating the region size, refer to the *Tivoli NetView for z/OS Tuning Guide*.

The buffer size parameter specifies the amount of storage allocated to the global buffer, the local buffer, and the router buffer. Each buffer is allocated the amount of storage specified by this parameter. The acceptable values are 8, 12, 16, 20, and 24 kilobytes.

The slot size parameter refers to a fixed portion of a buffer that is used to contain incoming error records. As these error records increase in number, you can increase the size of the buffer. Acceptable values are 200, 250, or 300 bytes. If the slot size is too low, the SVC76 mapper truncates error records of greater length than the slot size. If the slot size is too great, the number of slots available is reduced and records can be lost unless you also define a larger buffer size.

Defining Command Lists

The NetView command lists are defined in the following statements:

```
//DSICLD DD DSN=&SQ1..CNMCLST,DISP=SHR
//          DD DSN=&SQ1..CNMSAMP,DISP=SHR
//*          DD DSN=SYS1.PROCLIB,DISP=SHR
```

where &SQ1. defaults to NETVIEW.V5R1M0 in the startup procedure. If you are not using the AON inform policy, TIMER command, CGED command, or the DM command, comment out the CNMCLST data set.

If you write additional command lists, define at least one data set to store the command lists. Each command list is a separate member of the command list data set. The name of the command list is the data set member name. You can have several data sets for different types of command lists. Add the names of any data sets that you have defined for your command lists before those of the NetView command lists. Your statements might appear as in the following example:

```
//DSICLD DD DSN=DATASETNAME1,DISP=SHR
//      DD DSN=DATASETNAME2,DISP=SHR
//      DD DSN=&SQ1..CNMCLST,DISP=SHR
```

Coding several data set names this way is called concatenation. Concatenate as many data sets as you need. When you concatenate, specify a block size that is adequate for the largest data set. This ensures that the buffer space is sufficient when you reorder the DD statements. Code the BLKSIZE parameter on the first data set in the list.

Note: Always code the BLKSIZE, even if the largest data set is first in the list, if you change the order of the data sets.

Choose a command list data set block size that will minimize command list I/O by reading in, as a block, most of the command list statements. 3920 bytes is recommended and accommodates 49 command list statements. This fits in a 4K page and leaves room for a buffer header.

An example of using the BLKSIZE parameter is:

```
//DSICLD DD DSN=DATASETNAME1,DISP=SHR,DCB=BLKSIZE=3920
//      DD DSN=DATASETNAME2,DISP=SHR
//      DD DSN=&SQ1..CNMCLST,DISP=SHR
```

In this example, adequate buffer space is allocated even if DATASETNAME1 requires a block size of 3120 and DATASETNAME2 requires a block size of 3920.

If you allocate your command list library with secondary space, you might have to recycle the NetView program to update the library.

Specifying Definition Members and Sense Codes

The NetView definition data set members and sense codes are defined in the following statements:

```
//DSIPARM DD DSN=&Q1..&DOMAIN..DSIPARM,DISP=SHR
//      DD DSN=&SQ1..DSIPARM,DISP=SHR
```

If you have other definition and sense code libraries, concatenate them after this statement.

The following statements define the VTAM library:

```
//DSIVTAM DD DSN=&Q1..&DOMAIN..VTAMLST,DISP=SHR
//*      DD DSN=SYS1.VTAMLST,DISP=SHR
```

The following statements define the NetView operator profiles:

```
//DSIPRF DD DSN=&Q1..&DOMAIN..DSIPRF,DISP=SHR
//      DD DSN=&SQ1..DSIPRF,DISP=SHR
```

The following statements define the online help panels for English.

```
//CNMPNL1 DD DSN=&SQ1..CNMPNL1,DISP=SHR
//      DD DSN=&Q1..&DOMAIN..CNMPNL1,DISP=SHR
```

where

- &SQ1. defaults to NETVIEW.V5R1M0
- &Q1. defaults to NETVIEW.V5R1USER
- &DOMAIN. defaults to CNM01

The following statement defines the online help panels for Japanese.

```
//*CNMPNL1 DD DSN=&SQ1..SCNMPNL2,DISP=SHR
```

The following statement defines the data set used for user-defined message members and translation members:

```
//DSIMSG DD DSN=&SQ1..SDSIMSG1,DISP=SHR
```

The following statements define the NetView non-secured data sets:

```
//DSIOPEN DD DSN=&Q1..&DOMAIN..SDSIOPEN,DISP=SHR  
// DD DSN=&SQ1..SDSIOPEN,DISP=SHR
```

Running with Information/Management

If you use Information/Management to log problems from the hardware monitor, the NetView program must have access to the Information/Management load library and session members. The NetView program can accomplish this if you install Information/Management in a library accessible to the NetView program, such as NETVIEW.V5R1M0.CNMLINK. Information/Management libraries must be APF-authorized.

If you want information about...

Refer to...

Installing Information/Management with the NetView program	<i>Information/Management</i> library
--	---------------------------------------

Enabling Network and Trace Logs to Print Automatically

CNMPROC (CNMSJ009) includes the statements needed to automatically print the network and trace logs whenever the start procedure ends. Statements are provided for both the primary and secondary logs. The initial character string, `//***`, identifies each statement needed for printing the logs. Uncomment the statement to make it an active part of your JCL. CNMPROC (CNMSJ009) includes approximately 35 statements that can be changed to cause the automatic printing of both network and trace primary and secondary logs. The following example shows the steps for the primary NetView log:

```
//***PRNTLOGP EXEC PGM=DSIPRT,COND=EVEN  
//***DSILT DD DSN=&VQ1..&DOMAIN..DSILOGP,  
//*** DISP=SHR,AMP=AMORG  
//***SYSPRINT DD SYSOUT=&SOUTA  
//***DSILST DD SYSOUT=&SOUTA,DCB=(LRECL=121,RECFM=F,BLKSIZE=121)  
//***DSIINP DD DUMMY,DCB=(LRECL=80,RECFM=F,BLKSIZE=80)
```

Uncomment these steps to make it an active part of the JCL:

```
//PRNTLOGP EXEC PGM=DSIPRT,COND=EVEN  
//DSILT DD DSN=&VQ1..&DOMAIN..DSILOGP,  
// DISP=SHR,AMP=AMORG  
//SYSPRINT DD SYSOUT=&SOUTA  
//DSILST DD SYSOUT=&SOUTA,DCB=(LRECL=121,RECFM=F,BLKSIZE=121)  
//DSIINP DD DUMMY,DCB=(LRECL=80,RECFM=F,BLKSIZE=80)
```

Omitting Some of the Logs from Automatic Printing

The sample start procedure, CNMPROC (CNMSJ009), includes steps that pertain to the automatic printing of logs when the start procedure ends. Each step is identified by a comment explaining its function. If you want automatic printing of some of the logs but not all of them, alter only those statements about the logs you want to print. Read the comments to determine which steps pertain to logs you want.

The first step ensures that VSAM databases are properly closed before printing. Activate the statements in the first step, then activate the statements in any other steps that pertain to logs you want to print automatically.

Defining TCP/IP to NetView

The NetView program provides many services that rely on TCP/IP to communicate with remote applications. In order to communicate with TCP/IP, each of these services use a program function library, referred to as the TCP/IP MVS sockets library. This makes the NetView application an MVS sockets application.

Any MVS sockets application needs to reference TCP/IP configuration data. The method of accessing this data is defined by the z/OS version of TCP/IP that you are running. Refer to the *z/OS Communications Server IP Configuration Guide* for general information on how MVS sockets applications interact with TCP/IP. Among other things, this book discusses how an MVS sockets application:

- Gains affinity to a TCP/IP stack
- Resolves names to IP addresses
- Finds required TCP/IP configuration data sets

A model SYSTCPD DD statement is provided in the NetView startup procedure to identify the location of TCP/IP configuration data. A SYSTCPD statement is not required for NetView, but any MVS sockets application must be able to locate TCP/IP configuration data. Refer to the *z/OS Communications Server IP Configuration Guide* for information on how an MVS sockets application uses the SYSTCPD DD statement, and determine whether you need to use a SYSTCPD statement in the NetView startup procedure.

You will also need access to z/OS TCP/IP data sets from your NetView start procedure. If the z/OS TCP/IP data sets are not accessible from LNKLST, add the following z/OS TCP/IP data sets to the STEPLIB DD concatenation:

SEZALINK Executable load modules for concatenation to LINKLIB
SEZALNKG GDDMXD LOAD modules and sample programs
SEZALNK2 LB@ADMIN for the NCS administrator

To optimize performance, make these data sets available from LNKLST.

Notes:

1. For each of the TCP/IP services provided by NetView, the stack affinity is specified in the CNMSTYLE file using the TCPname statement. Using this name, NetView sets the stack affinity by specifying it on an INITAPI socket call.
2. There are NetView applications that are not a part of the NetView address space that rely on TCP/IP to communicate with remote applications. Some of these applications use the z/OS UNIX sockets library. These applications are therefore z/OS UNIX sockets applications. Information on how these applications reference TCP/IP configuration data is discussed in the books describing those applications. Examples of z/OS UNIX sockets applications are the Event/Automation Service and the UNIX Command Server.

Running the NetView Application as a Job

To run the NetView application as a job, remove the asterisk (*) from the first two lines and the final two lines. Also update the JOB statement to reflect your installation. The first two lines are:

```

/*CNMSJ009 JOB 'ACCOUNTING INFORMATION','NETVIEW STARTUP PROC',
/* CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)

```

The final two lines are:

```

/*          PEND
/*CNMPROC EXEC PROC=CNMAPROC

```

Make sure the PROC name you specify in this statement agrees with the one you specify in CNMPSSI (CNMSJ010).

Updating CNMSTYLE

CNMSTYLE is a member of DSIPARM that is used during NetView initialization. Changes to the NetView initialization process are made in CNMSTYLE.

The member name for CNMSTYLE is controlled by the value of &NV2I in the NetView start procedure. The NetView default for &NV2I is NM. If you specify a value for &NV2I (*xx*), NetView will read *CxxSTYLE* in DSIPARM for initialization parameters. If this member is not found, NetView will read CNMSTYLE instead. For more information on NV2I, see “Setting System Variables” on page 39.

The sample CNMSTYLE member in DSIPARM contains descriptive comments about the types of statements that can be included in the member. Read the comments and review the defaults. The sections that follow provide additional details for some of the NetView functions.

The recommended way to customize CNMSTYLE is to make global (enterprise) changes to the sample CNMSTYLE, then copy the modified CNMSTYLE to each NetView system. Make system-specific changes to %INCLUDE member *CxxSTGEN* (where *xx* is the value of &NV2I). You can code all override statements for CNMSTYLE in this member. Duplicate statements found in *CxxSTGEN* override earlier statements.

Note: If you make changes to CNMSTYLE or its %INCLUDE members while the NetView program is running, recycle NetView to enable the changes. For certain types of changes (including hardware monitor, session monitor, Web interface, NetView Resource Manager, visual BLDVIEWS, and various global variable updates), you can also use the RESTYLE command to activate these changes without recycling the NetView program. For more information on the types of changes that can be activated in this manner, refer to the *Tivoli NetView for z/OS Command Reference* for the RESTYLE command.

If you want information about...	Refer to...
CNMSTYLE statements	Comments in the CNMSTYLE file and <i>Tivoli NetView for z/OS Administration Reference</i>
RESTYLE command	<i>Tivoli NetView for z/OS Command Reference</i>

Enabling Towers

The NetView components are activated with TOWER statements in CNMSTYLE. Uncomment the components that you are using:

Tower	Description
SA	Enables System Automation for OS/390.
AON	Enables network automation (AON component).

	Subtower	Description
	SNA	SNA automation (AON/SNA).
		Note: The X25 subtower is a subtower of the SNA subtower. It provides X.25 support for AON/SNA.
	TCP	TCP/IP automation (AON/TCP).
MSM		Enables the MultiSystem Manager.
	Subtower	Description
	LNМ	Lan Network Manager feature.
	IP	IP feature.
	OPN	Open feature.
	NTF	NetFinity feature.
	TMR	Tivoli Managed Resource feature.
GRAPHICS		Enables the NetView management console.
	Subtower	Description
	SNATM	SNA Topology Manager.
MVScmdMgt		Enables MVS command management.
	NPDA	Enables the hardware monitor.
	TARA	Enables the 4700 support facility.
	NLDM	Enables the session monitor.
	AMI	Enables the Application Management Instrumentation.
	AutoBridge	Enables the NetView AutoBridge, which provides an MVS-based interface between the NetView program and Information/Management.

An example tower statement follows:

```
TOWER = *SA *AON *MSM *Graphics MVScmdMgt NPDA *TARA *NLDM
*AMI *AutoBridge
```

To enable a tower or subtower, remove the * before the tower name, and then recycle NetView.

Example subtower statements follow:

```
TOWER.AON = SNA TCP
TOWER.MSM = LNM IP OPN NTF TMR
TOWER.Graphics = SNATM
```

When the tower is enabled, various associated functions are also enabled during initialization. NetView must be recycled for the tower statements to take effect.

Note: You can add data REXX logic to CNMSTGEN to conditionally process definition statements based on whether a particular tower is enabled. An example follows:


```

%> IF tower('towername') THEN
%> DO;
definition statements
%> END;

```

If you use data REXX logic, make sure that all the statements begin in column one. Lines beginning with a blank are considered continuation statements from the previous line. Also, data REXX files must begin with either a `/* %DATA */` or `/* %LOGIC */` statement.

For AON and MultiSystem Manager, you must also enable the subtowers.

If you want information about...	Refer to...
AON, hardware monitor, session monitor, 4700 support facility	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
MultiSystem Manager subtowers	<i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i>

%INCLUDE Members

The following members are included when CNMSTYLE initializes:

%INCLUDE Member	Usage
CNMSTPWD	If needed, you can use this member to include VPD, VSAM, and ACB passwords. You can use READSEC to protect CNMSTPWD from being displayed by the BROWSE command.
CNMSTASK	NetView-provided task statements. Do not modify this member. Instead, include any task statements you want to include directly in CNMSTYLE or CNMSTGEN. The task statements in CNMSTYLE that come after the %INCLUDE for CNMSTASK override those provided in CNMSTASK.
CNMSTTWR	Includes style statements from non-NetView towers. Do not edit this member unless specifically instructed by documentation for a tower you are installing.
C&NV2I.STGEN	You can include additional or modified definition statements in this member, including DATA REXX logic.

Using Symbolics

Many NetView processes require the RODM name, NetView domain and ACB password. You can use CNMSTYLE to set global variables for these names. The values that you assign to these global variables can be system symbolics from member IEASYMxx in SYS1.PARMLIB or user-specified values.

Table 9. Symbolics in CNMSTYLE

Symbolic	CNMSTYLE Statement
RODM Name	RODMname = &CNMRODM. Note: This statement is ignored if you are not using RODM.

Table 9. Symbolics in CNMSTYLE (continued)

Symbolic	CNMSTYLE Statement
NetView domain	DOMAIN=C&NV2I.01 (NetView-supplied default is CNM01) Note: This identifier is the access method control block (ACB) name that appears on the VTAM APPL statement.
TCP name	TCPname=&CNMTCPN.
Network ID	NetID=&CNMNETID.

Notes:

1. If you specified the NetView domain ID or password in CNMPROC (CNMSJ009), the DOMAIN keyword in CNMSTYLE or the ACBpassword keyword in CNMSTPWD do not have to be coded. If they are coded, they are ignored unless the parameters passed by CNMPROC are null. If the domain password is not specified in CNMPROC (CNMSJ009) or in CNMSTYLE, the DOMAIN name becomes the password.
2. The system symbolics set in IEASYMxx are enabled for all address spaces. Global variables that you set using CNMSTYLE only apply to this NetView address space. For example, if you use IEASYMxx to specify the RODM name in DUIGINIT, your setting in CNMSTYLE must match because GMFHS cannot use the CNMSTYLE value.

Setting up Security

You can use the SECOPTS statement to specify:

- Operator security
- Command authority
- Span of control authority
- Web browser access

If you want information about...	Refer to...
Security options	<i>Tivoli NetView for z/OS Security Reference</i>
SECOPTS keywords	<i>Tivoli NetView for z/OS Administration Reference</i>

Specifying That a Command Is to Run Automatically When the NetView Program Is Started

To define a command or a command list to run automatically when the NetView program is started, use the auxInitCmd keyword in CNMSTYLE. You can specify any number of commands or command lists to be run. The EBCDIC value following the auxInitCmd keyword determines the order the commands are run.

An example follows:

```
auxInitCmd.A = MSG SYSOP,Auxiliary commands beginning.
auxInitCmd.AC = RESTORE TIMER
```

In this case, the MSG SYSOP command (A) will run before the RESTORE TIMER command (AC).

Note: These AuxInitCmd commands run before any commands at any autotask. All commands for autotasks, including both task initial command lists and commands sent by EXCMD, will be queued and held up. They will run only after all AuxInitCmds have completed. Messages are also queued; they are not submitted to automation nor logged until all AuxInitCmds have completed.

If you want information about...	Refer to...
Creating a command list to run at NetView initialization	<i>Tivoli NetView for z/OS Customization: Using REXX and the NetView Command List Language</i>

Customizing NetView Components in CNMSTYLE

The initialization values for some NetView components are specified in CNMSTYLE. Table 10 shows the NetView component, its primary task name, its initialization member in DSIPARM, and the CNMSTYLE statement prefix for its initialization values.

Table 10. NetView Component Initialization

NetView Component	Primary Task Name	Initialization Member	CNMSTYLE Statement Prefix
CNM data transfer	<i>domid</i> LUC	DSILUCTD	LUC.
Get-host-by task	DUIDGHB	DUIIGHB	GHB.
Hardware monitor	BNJDSERV	BNJMBDST	NPDA.
IP log	DSIIPLOG	DSIILGCF	IPLOG.
LU 6.2 communication	DSIU DST	DSIUINIT	RMTINIT.
NetView Resource Manager	AUTONRM	n/a	NRM.
Resource status monitor	CNMTAMEL	DUIISFP DUIIFPMEM	TAMEL.
REXEC server	DSIRXEXC	DSIREXCF	REXEC.
RSH server	DSIRSH	DSIRSHCF	RSH.
Session monitor	DSIAMLUT AAUTSKLP	DSIAMLTD AAUPRMLP	NLDM.
TCP/IP alert receiver	DSIRTR	DSIRTTD	RTT.
TCP/IP communication for the NetView 3270 management console.	DSITCPIP	DSITPCPF	MCON.
Visual BLDVIEWS	AUTOVBV	n/a	VBV.
Web server interface task	DSIWBT SK	DSIWB MEM	WEB.

If you want information about...	Refer to...
CNMSTYLE statements	<i>Tivoli NetView for z/OS Administration Reference</i>
RESTYLE command	<i>Tivoli NetView for z/OS Command Reference</i>

Starting the NetView Program Before Starting JES

If you plan to start the NetView program and the SSI under the master subsystem before you start JES, the following rules apply:

- Start the PROC with the START command using the parameter SUB=MSTR.
- When you start the NetView program with the SUB=MSTR parameter, ensure that the TASK.DSIRQJOB.INIT statement in CNMSTYLE has a value of YES to start the DSIRQJOB task. This is needed for the SUBMIT or ALLOCATE commands to complete successfully.
- Store the procedure in the data set SYS1.PROCLIB, not in a user PROCLIB supported by JES.
- The procedures must contain only a single job step.
- You cannot reference SYSIN, SYSOUT, or VIO data sets. If you are using the sample start procedures, comment out all references to the symbolic SOUTA=A in CNMPROC (CNMSJ009).
- JES should remain coded as the primary subsystem. But in the IEFSSN member for JES, code the NOSTART parameter so that MVS does not automatically start JES at initialization.
- You cannot specify AMP=AMORG on a log data set.

Chapter 6. Activating NetView

You are ready to start the NetView program.

If you have installed the NetView system in a subarea other than 01 and domain CNM01, make the appropriate substitution as you go through the steps in this chapter. Modify the PROCLIB members to match your system naming conventions before running the steps in this chapter.

Note: Be sure to read each step in its entirety before executing it.

Starting VTAM Using Job CNMNET

IPLing MVS with the CLPA option picks up any APF-authorization and subsystem definitions that are in effect, as well as LPALIB modules.

1. If you ran “Restarting the Target System with the CLPA Option” on page 16, you need to re-IPL MVS at this point if the following applies:
 - Running a previous release of the NetView program in production.
 - Defining a user-defined RACF resource class and a user-defined RACF router table for RODM.

Also, you need to IPL if you are installing the NetView program for the first time and did not IPL in “Restarting the Target System with the CLPA Option” on page 16.

2. Read the comments in the VTAM startup procedure. Some data sets may need to be uncommented.
3. Start VTAM by entering the following at the system console:

```
S CNMNET,,, (LIST=xx)
```

Where *xx* is 01, 02, or 99 if you installed in network NETA and B1 if you installed in network NETB. For example, to start VTAM in subarea 02 and domain CNM02 in network NETA, enter the following:

```
S CNMNET,,, (LIST=02)
```

To start VTAM in subarea 01 and domain B01NV in network NETB, enter the following:

```
S CNMNET,,, (LIST=B1)
```

You see messages similar to those in Figure 8 on page 52.

If you want information about...

Refer to...

Defining a user-defined RACF resource class and a user-defined RACF router table *Tivoli NetView for z/OS Security Reference.*

```

$HASP100 CNMNET   ON STCINRDR
$HASP373 CNMNET   STARTED
IEF403I CNMNET - STARTED - TIME=23.50.51
IST093I ISTCDRDY ACTIVE
IST315I VTAM INTERNAL TRACE ACTIVE - MODE = INT, SIZE = 002 696
IST199I OPTIONS = API SMS PSS LOCK PIU MSG CIO SSCP NRM APPC ESC VCNS
IST199I OPTIONS = LCS
IST093I A01ADJ ACTIVE
IST093I A01PATH ACTIVE
IST093I A01CDRM ACTIVE
IST093I A01M ACTIVE
IST093I A01CDRM1 ACTIVE
IST093I A01CDRSC ACTIVE
IST093I A01CDRS1 ACTIVE
IST093I A01LOCAL ACTIVE
IST093I A01SNA ACTIVE
IST093I CTCA0102 ACTIVE
IST380I ERROR FOR ID = A01P7A0 - REQUEST: ACTLINK, SENSE: 081C003C
IST105I A01P7A0 NODE NOW INACTIVE
IST093I CTNA0104 ACTIVE
IST093I A01APPLS ACTIVE
IST093I A01USER ACTIVE
IST093I A01MVS ACTIVE
IST322I CONFIGURATION A01NVAS ERROR IGNORED - TABLE NOT FOUND 718
IST330I TABLE TYPE = MODETAB NAME = EMSMODE
IST093I A01NVAS ACTIVE
IST020I VTAM INITIALIZATION COMPLETE
IST984I USER EXIT ISTEUCV IS ACTIVE

```

Figure 8. VTAM Messages

Starting the NetView Subsystem Address Space Using Job CNMPSSI

To start the NetView subsystem address application address space, enter the following at the system console:

```
S CNMPSSI
```

You see messages similar to those in Figure 9.

```

$HASP373 CNMPSSI   STARTED
IEF403I CNMPSSI - STARTED - TIME=12.13.35
CNM226I NETVIEW PROGRAM TO PROGRAM INTERFACE INITIALIZATION IS COMPLETED
CNM541I NETVIEW SUBSYSTEM INITIALIZED SUCCESSFULLY

```

Figure 9. Messages for Starting the Application Address Space

If you accidentally start the wrong level of the NetView program and immediately cancel it and start another level, you can receive message CNM555I stating that an invalid subsystem table was found. Ignore this message.

Note: Once the NetView subsystem address space is started, it is not intended to be stopped unless all applications using it have been stopped first. If you

stop the NetView subsystem address space before stopping its applications, unpredictable results can occur, including system abends and lost data.

Starting the NetView Program Using Job CNMPROC

If you are running an additional copy of the NetView program on this LPAR and have the hardware monitor, enter the following at the system console:

```
S CNMPROC,PROG=DSIMNT
```

If you are not running an additional copy of the NetView program in this LPAR, enter the following at the system console:

```
S CNMPROC
```

You see messages similar to those in Figure 10.

```
$HASP373 CNMPROC  STARTED
IEF403I CNMPROC -  STARTED - TIME=12.14.27
BNJ080I BNJLINTB -  BUFFER SIZE=24K,SLOT SIZE=200
DSI530I 'DSIMONIT' : 'DSIMONIT' IS READY AND WAITING FOR WORK
DSI530I 'DSIDCBMT' : 'DSIDCBMT' IS READY AND WAITING FOR WORK
DSI530I 'DSITIMMT' : 'DSITIMMT' IS READY AND WAITING FOR WORK
BNH350I NETVIEW HIGH-LEVEL LANGUAGE SUPPORT HAS BEEN SUCCESSFULLY
INITIALIZED WITH LE/370 LIBRARIES
DSI530I 'DSIHLLMT' : 'DSIHLLMT' IS READY AND WAITING FOR WORK
DSI530I 'DSISTMMT' : 'DSISTMMT' IS READY AND WAITING FOR WORK
*0003 DSI802A CNM01  REPLY WITH VALID NCCF SYSTEM OPERATOR COMMAND
DSI530I 'DSILOGMT' : 'DSILOGMT' IS READY AND WAITING FOR WORK
```

Figure 10. Messages for Starting the NetView Program

You will see messages for the following tasks:

- DSIDCBMT
- DSIHLLMT
- DSILOGMT
- DSIMONIT
- DSISTMMT
- DSITIMMT

These tasks are part of the main task of the NetView program. They are automatically started when NetView is started and are not defined by TASK statements externally. They cannot be started or stopped by the STARTCNM and STOPCNM commands. They have no user exits, and cannot be stopped with the STOP TASK command either. Although these tasks can be forced to ABEND with the STOP FORCE command, after the ABEND, NetView automatically restarts the task. Therefore, do not try to stop these tasks during NetView initialization.

VSAM might return IEC161I messages. Do not be concerned with these messages if all the tasks start properly.

You do not need to respond to message DSI802A.

You might see several installation exit load failure messages, such as DSIEX01 through DSIEX21, or BNJPALLEX. These messages might not be a cause for concern. You can use the LOADEXIT statement in CNMSTYLE to load some, all, or none of the NetView-provided installation exits.

Note: It is not recommended that you code a dummy exit for EPWSVC76.
However, if you choose to do so, place it in LPALIB.

Chapter 7. Verifying the Installation

This chapter leads you through a series of commands to test the NetView program you have just started. Execute the commands in exactly the order presented. If you make errors, or issue the commands out of sequence, the contents of the network logs and the results of the operator tests will be different from the examples you see in this chapter.

To verify the NetView installation, issue instructions that will test the installation. If you cannot start the tests, ensure that you have correctly performed all the installation procedures in the NetView program directory and all the installation procedures in this book.

Notes:

1. This chapter leads you through several NetView panels. Your data can differ from the panels in this book because display format depends on the type of terminal you are using.
2. If you customized your system to suppress messages, remember to check the system and NetView log for critical information that might have been suppressed during the verification of your NetView installation.
3. If you are not running a previous release of the NetView program in production, enter CNMxx on the NetView operator's screen. The xx specifies the VTAM subarea. The subarea can be 01, 02, or 99 if you are using network NETA. CNMxx specifies the main task defined in A01APPLS (CNMS0013).
The subarea is B1 if you are using network NETB. CNMB1 specifies the main task defined in B01APPLS (CNMS0113).
4. For the tasks that will be initialized, see Appendix A. NetView Functions and Tasks.

Testing the Command Facility

To start the test of NetView, begin with one of the following instructions:

Note: If you enter either of the first two instructions, ensure that the LOGMODE entry matches your hardware device. Also, specify a minimum RU size of 16 bytes or greater on the bind when you are logging a 327x terminal or emulator onto the NetView system as an OST.

- For an SNA terminal, issue this instruction:
LOGON APPLID(CNM01) LOGMODE(M2SDLCQ)
- For a non-SNA terminal, issue this instruction:
LOGON APPLID(CNM01) LOGMODE(M23270I)
- For a non-SDLC NetView system, it is not necessary to specify the RU size. If the LOGMODE entry does not match, enter:
LOGON APPLID(CNM01) LOGMODE(axxxxxa)

Where *axxxxxa* is the ID of your hardware device. To take advantage of wide-screen capability, such as an IBM 3290 terminal, use either the MSDLCQ or MBSCQ LOGMODE. Regardless of the screen size, use a query LOGMODE with the IBM 3290.

If you want information about...	Refer to...
Available LOGMODEs for AMODETAB (CNMS0001)	"Appendix B. NetView Samples Overview" on page 81

- If you are running a previous release of the NetView program in production, enter the following at the NetView operator terminal:

```
LOGON APPLID(CNM01) LOGMODE(bxxxxxb)
```

Where *bxxxxxb* is your LOGMODE entry.

A panel similar to Figure 11 is displayed.

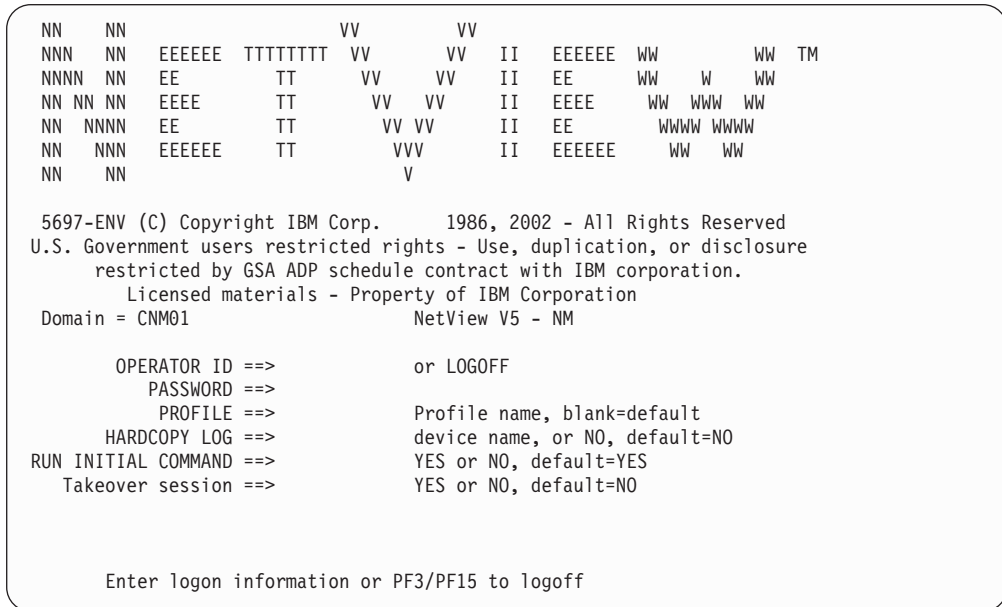


Figure 11. Tivoli NetView for z/OS Logon Panel

The NetView logon panel is different if SECOPTS.OPERSEC=SAFDEF is specified in CNMSTYLE or OPERSEC=SAFDEF is specified on the REFRESH command: There is no PROFILE ==> field and the HARDCOPY LOG ==> field has no default.

In the PROFILE ==> field, system symbolic substitution will be performed on records read from the DSIOPF member in the DSIPARM data set and the specified profile member in the DSIPRF data set. The NetView-supplied &DOMAIN symbolic is also included in the substitution process. The substitution is performed after comment removal but before record processing. After substitution, comments are also removed. Substitution is always performed on the &DOMAIN symbolic, unless substitution was disabled when NetView was started.

Enter the following in both the OPERATOR ID field and in the PASSWORD field:

```
NETOP1
```

Note: Blanks entered in the NetView logon fields will be treated as null characters. For example, OPER 1 entered in the OPERATOR ID field of the NetView logon panel will be treated as OPER1 because the blank between R and 1 is treated as a null character.

A panel similar to Figure 12 is displayed.

```
NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 07:32:27
- CNM01  DSI020I OPERATOR NETOP1 LOGGED ON FROM TERMINAL N701 USING PROFILE
(DSIPROFB ), HCL ( )
- CNM01  DSI083I AUTOWRAP STOPPED
C CNM01  CNM357I PFKDEF : PF KEY SETTINGS NOW ESTABLISHED. 'DISPFK' TO SEE
YOUR PF KEY SETTINGS
| CNM01

Enter LOG or LOGOFF to terminate session.
Enter HELP to obtain help.
Lead operator has been notified of your logon.
To obtain help from the NETWORK CONTROL CENTER, enter

MSG PPT, your question here

| CNM01
News for April 1, 2002

Tivoli NetView for z/OS V5 contains these enhancements and more:

* Web interface with new look and feel
  - Frame-based with a portfolio of available functions
  - Integrated online help
* Native SNMP and PING commands
* Dynamic VIPA management
* Router fault isolation support in MSM IP agent
* Enhanced MIB Browser with integrated MIB groups
* Additional launches from NMC such as TBSM and NPM for TCP/IP
* NMC Server on Linux for zSeries and NMC Console on Linux (Intel)
* Security enhancements for NMC
??? ***
```

Figure 12. NetView News Panel

The information that appears in this panel is contained in member CNMNEWS in the NETVIEW.V5R1M0.SDSIOPEN data set. The date and time at the top of your panel will differ from the examples.

Note: When verifying your installation, be sure to clear the console between each step. In Figure 12, the 3 asterisks at the bottom of a panel indicate a held screen.

To clear the screen and go to the NetView main menu, press the Clear or ENTER key. After the NetView program executes the operator profile, a panel similar to Figure 13 on page 58 is displayed.

```

CNM1NETV          Tivoli NetView for z/OS Version 5          Main Menu

                    Operator ID = NETOP1   Application = CNM01007

Enter a command (shown highlighted or in white) and press Enter.

Browse Facility          BROWSE command
Command Facility        NCCF command
News                    NEWS command
PF Key Settings         DISPFK command
Help Facility           HELP command
Index of help topics    INDEX command
Help Desk               HELPDESK command

To log off or disconnect LOGOFF command or DISC command

TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
Action====>

```

Figure 13. NetView Main Menu Panel

To go to the NetView help facility, enter the following:

HELP NETVIEW

You see a panel similar to Figure 14.

```

CNMKNEEW          NETVIEW HELP FACILITY MAIN MENU

Select To get information about

 1 Operator's overview of the NetView Program
 2 Using the NetView Help Desk for operators
 3 Using NetView online message help
 4 Using command and command list help
 5 Finding help on VTAM in NetView
 6 Finding help on IBM LAN Network Manager
 7 Finding help on RODM (Resource Object Data Manager)
 8 Finding help on GMFHS (Graphic Monitor Facility Host Subsystem)
 9 Help for the NETVIEW stage (NetView Pipelines)
 A All NetView commands
 I Finding help in the Index
 P Help for PIPE syntax

Type a value (1 to 9, A, I, or P) and press ENTER.

TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
Action====>

```

Figure 14. NetView Help Facility Main Menu

Press PF3 twice or End twice to go to the command facility. You see a panel similar to Figure 15 on page 59.

```

NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 07:35:02
| CNM01
News for April 1, 2002
* Enhanced e-mail support
* Security improvements for AON, TSOSERV, and auditing commands
* Improved TIMER function and security through the use of RMTCMD
* Streamlined installation and customization with additional CNMSTYLE
  support and removal of obsolete functions
-----
???
```

Figure 15. NetView Command Facility Panel

Then enter:

HELP

at the action line. You see a panel similar to Figure 16.

```

CNMKNCCF          COMMAND FACILITY HELP MENU

Select To get information about

  1  Operator's overview of the command facility
  2  Using the terminal access facility (TAF)

  3  The command facility screen
  4  Command facility commands and command lists

  5  Field level help

Type a number (1 through 5) and press ENTER.

                HELP NETVIEW ---> NetView Help Menu

TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
Action===>
```

Figure 16. Command Facility Help Menu Panel

Select option 4 to display the command facility commands. You see a panel similar to Figure 17 on page 60.

```

CNMPNL1.EUYCLIST      HELP NCCF COMMANDS      LINE 0 OF 252
*----- Top of Data -----*
ACQ      Acquire a resource . . . . . A2
ACT      Activate VTAM resources . . . . . A3
ADAPTER  Monitor and control a LAN adapter . . . . . A6
ADDCMD   Add or replace user-written commands . . . . . A7
AFTER    Schedule a command to run later . . . . . A8
AINQ     Display translation tables . . . . . A12
ALLOCATE Allocate an MVS data set . . . . . A18
APPLS    Display minor nodes . . . . . A41
APPLSPEN Display sessions state . . . . . A42
ASSIGN   Define operators and groups . . . . . A44
AT       Schedule a command to run . . . . . A47
ATTACH   Create a virtual OST (VOST) . . . . . A48
AUPD     Alter translation table . . . . . A49
AUTBNABL Determine AUTOBYPAS function authorization . . . A50
AUTOCNT  Using the automation table . . . . . A51
AUTOMAN  Add or change automation table statements . . A53
AUTODROP Check commands by LOADCL . . . . . A54
AUTOTASK Start automated operator . . . . . A56
AUTOTBL  Activate message automation . . . . . A57
AUTOTEST Start/stop automation table testing . . . . . A58
AUTOTR   Set buffer trace on or off . . . . . A59
AUTOWRAP Automatically overlay screen . . . . . A61
BFRUSE   Display VTAM buffer usage . . . . . B3
BFSESS   Start full-screen session . . . . . B4
BGNSSESS Start subsystem session . . . . . B5
BLOG     Browse network log . . . . . B7
BOSESS   Start operator session . . . . . B8
BRIDGE   Monitor and control LAN bridge . . . . . B10
TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
CMD==>

```

Figure 17. Using NetView Command and Command List Help Panel

Press PF3 twice or End twice to return to the command facility.

Use the CLEAR command to clear the messages from the command facility panel. Then, use the WHO command list to display information about your session.

Enter:

WHO

at the cursor. You see a panel similar to Figure 18 on page 61.

```

NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 07:37:19
* CNM01  WHO
C CNM01  LIST STATUS=OPS
- CNM01  OPERATOR: NETOP1  TERM: N701  STATUS: ACTIVE
- CNM01  OPERATOR: AUTOAON  TERM: AUTOAON  STATUS: ACTIVE
- CNM01  OPERATOR: AUTO1  TERM: AUTO1  STATUS: ACTIVE
- CNM01  OPERATOR: AUTO2  TERM: AUTO2  STATUS: ACTIVE
- CNM01  OPERATOR: DBAUTO1  TERM: DBAUTO1  STATUS: ACTIVE
- CNM01  OPERATOR: DBAUTO2  TERM: DBAUTO2  STATUS: ACTIVE
- CNM01  OPERATOR: DSIIPCHK  TERM: DSIIPCHK  STATUS: ACTIVE
- CNM01  OPERATOR: DSIMCAOP  TERM: DSIMCAOP  STATUS: ACTIVE
- CNM01  END OF STATUS DISPLAY
C CNM01  LIST STATUS=NNT
- CNM01  MAX SESS: 00032767
- CNM01  NO ACTIVE NCCF TO NCCF SESSIONS FOUND
C CNM01  LIST NETOP1
- CNM01  STATION: NETOP1  TERM: N701
- CNM01  HCOPY: NOT ACTIVE  PROFILE: DSIPROFB
- CNM01  STATUS: ACTIVE  IDLE MINUTES: 0
- CNM01  ATTENDED: YES  CURRENT COMMAND: WHO
- CNM01  AUTHRCVR: YES  CONTROL: GLOBAL
- CNM01  NGMFADMN: YES  DEFAULT MVS CONSOLE NAME: NONE
- CNM01  NGMVFSPN: NNNN (NO SPAN CHECKING ON NMC VIEWS)
- CNM01  NGMFCMDS: YES  AUTOTASK: NO
- CNM01  IP ADDRESS: N/A
- CNM01  OP CLASS LIST: NONE
- CNM01  DOMAIN LIST: CNM01 (I) CNM02 (I) CNM99 (I)
- CNM01  ACTIVE SPAN LIST: NONE
- CNM01  Task Serial: 16072
- CNM01  Messages Pending: 0 Held: 0
??? ***

```

Figure 18. Information about Your NetView Session Panel

Press the ENTER key to release any messages being held by the command facility. The three asterisks (***) at the bottom of the screen indicate that there are more messages to be displayed.

The DISPFK command displays the PF key settings for any NetView component from which you entered the command list. Enter:

DISPFK

at the cursor. You see a panel similar to Figure 19 on page 62.

```

CNMKWIND OUTPUT FROM  DISPFK                                LINE 0 OF 33
*----- Top of Data -----*
DISPLAY OF PF/PA KEY SETTINGS FOR NCCF
KEY  ----TYPE----  -----COMMAND-----  SET-APPL
PA1  IMMED,IGNORE  RESET                      NETVIEW
PA2  IMMED,IGNORE  AUTOWRAP TOGGLE           NETVIEW
PA3  IMMED,IGNORE  RETRIEVE AND EXECUTE      NETVIEW
PF1  IMMED,APPEND  HELP                      NETVIEW
PF2  IMMED,APPEND  GO                        NCCF
PF3  IMMED,IGNORE  RETURN                   NETVIEW
PF4  IMMED,APPEND  DISPFK                   NETVIEW
PF5  IMMED,IGNORE  BROWSE NETLOGA          NETVIEW
PF6  IMMED,IGNORE  ROLL                     NETVIEW
PF7  IMMED,APPEND  TASKUTIL                 NCCF
PF8  IMMED,IGNORE  PIPE NETVIEW LIST STATUS= NCCF
                                OPS|COL|CONS ONLY
PF9  DELAY,IGNORE  PIPE HELDMSG | CONSOLE DELETE NCCF
PF10 IMMED,APPEND  WINDOW                   NETVIEW
PF11 IMMED,IGNORE  HOLD                     NCCF
PF12 IMMED,IGNORE  RETRIEVE                 NETVIEW
PF13 IMMED,APPEND  CMD HELP                 NETVIEW
PF14 IMMED,APPEND  STATIONS                 NETVIEW
PF15 IMMED,IGNORE  LINES                    NETVIEW
PF16 IMMED,IGNORE  PFKDEF CNMKEYS2         NETVIEW
PF17 IMMED,IGNORE  BROWSE NETLOGI          NETVIEW
PF18 IMMED,APPEND  NCCF                     NETVIEW
PF19 IMMED,APPEND  TASKUTIL                 NCCF
PF20 IMMED,APPEND  TS                      NCCF
PF21 DELAY,IGNORE  PIPE HELDMSG | CONSOLE DELETE NCCF
PF22 IMMED,APPEND  PIPE NETVIEW LIST STATUS= NCCF
                                TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
CMD==>

```

Figure 19. PF Key Settings for the Command Facility

Press PF3 or End to return to the command facility.

Use the BROWSE DSIOPF command list to see NetView operator identifiers and passwords. Enter:

BROWSE DSIOPF

at the cursor. You see a panel similar to Figure 20 on page 63.


```

NETVIEW.BRWS ----- BROWSE DSIOPF (DSIPARM ) --- LINE 00000 TO 00025 OF 00218
                                                                SCROLL ==> CSR
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+-----
***** TOP OF DATA ***** DATASET: 6
*****
* LICENSED MATERIALS - PROPERTY OF IBM *
* 5697-ENV (C) COPYRIGHT IBM CORPORATION 1986, 2002 *
* ALL RIGHTS RESERVED. *
* US GOVERNMENT USERS RESTRICTED RIGHTS *
* - USE, DUPLICATION OR DISCLOSURE RESTRICTED BY *
* GSA ADP SCHEDULE CONTRACT WITH IBM CORPORATION. *
*****
* NAME(DSIOPF) SAMPLE(DSIOPF) RELATED-TO( ) *
* DESCRIPTION: SAMPLE DSIPARM - OPERATOR DEFINITIONS/PASSWORDS *
* *
*****
* INCLUDE ANY CUSTOMER OPERATOR DEFINITIONS *
*****
***** START OF MEMBER DSIOPFU FROM DSIOPF DATASET: 6 *****
*****
* LICENSED MATERIALS - PROPERTY OF Tivoli Systems *
* 5697-B82 (C) COPYRIGHT Tivoli Systems 1997, 1999 *
* 5655-007 (C) COPYRIGHT IBM CORP. 1993, 1995. *
* ALL RIGHTS RESERVED. *
* *
* US GOVERNMENT USERS RESTRICTED RIGHTS *
* - USE, DUPLICATION OR DISCLOSURE RESTRICTED BY *
* GSA ADP SCHEDULE CONTRACT WITH IBM CORPORATION. *
*****

CMD==>
TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'

```

Figure 20. Display of Operator Identifiers and Passwords

Use PF8 to page forward through the contents.

Press PF3 to return to the command facility panel.

You also need to verify your installation of the NetView system by entering the NetView program through the terminal access facility (TAF). Use the BFSESS command list to start a full-screen session with another subsystem. If you are using subarea 01 for your installation, enter:

```
BFSESS CNM01
```

or, if you are running a previous release of the NetView program in production, enter:

```
BFSESS CNM02
```

If you are installing network NETB, you need to designate your TAF LU name as follows:

```
BFSESS B01NV,TAF01Fxx
```

Where xx is the last two digits of the program application name. You can verify the program application name on the NetView main menu. For example, in Figure 13 on page 58, the application name is CNM01007.

Press ENTER again to display the NetView logon panel, if needed.

To log on to the NetView system, enter:

```
OPER1
```

in the OPERATOR ID field, and
OPER1

in the PASSWORD field. You see a panel similar to Figure 21.

```
NetView V5 - NM          Tivoli NetView  CNM01 OPER1  11/08/01 07:48:58
- CNM01  DSI020I OPERATOR OPER1 LOGGED ON FROM TERMINAL TF01#000 USING
        PROFILE (DSIPROFA ), HCL ( )
- CNM01  DSI083I AUTOWRAP STOPPED
C CNM01  CNM357I PFKDEF : PF KEY SETTINGS NOW ESTABLISHED. 'DISPFK' TO SEE
        YOUR PF KEY SETTINGS
| CNM01

        Enter LOG or LOGOFF to terminate session.
        Enter HELP to obtain help.
        Lead operator has been notified of your logon.
        To obtain help from the NETWORK CONTROL CENTER, enter

                MSG PPT, your question here

| CNM01
News for April 1, 2002

        Tivoli NetView for z/OS V5 contains these enhancements and more:

        * Web interface with new look and feel
          - Frame-based with a portfolio of available functions
        ??? ***
```

Figure 21. Entering the Command Facility after Logging On

Use the Clear or ENTER key to have the NetView program run the operator profile. After the NetView system executes the operator profile, you see the NetView main menu.

Enter:
BR DSIOPF

to see a panel similar to Figure 22 on page 65. BR is a synonym for BROWSE. Scroll down to view the operator identifiers and passwords.

```

NETVIEW.BRWS ----- BROWSE DSIOPF (DSIPARM ) --- LINE 00054 TO 00071 OF 00218
                                           SCROLL ==> CSR
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+-----
*
*          NOTE: THE FIRST "NETOP " SIGNED ON WILL BE          *
*          THE AUTHORIZED MESSAGE RECEIVER.                    *
*
*          NOTE: THE PASSWORD PARAMETER IS PROVIDED            *
*          BUT IT WILL NOT BE CHECKED IF                      *
*          OPERSEC=MINIMAL HAS BEEN SPECIFIED                 *
*          ON THE OPTIONS STATEMENT (DSIDMN)                   *
*
*****
OPER1      OPERATOR      PASSWORD=OPER1
           PROFILE      DSIPROFA
OPER2      OPERATOR      PASSWORD=OPER2
           PROFILE      DSIPROFA
OPER3      OPERATOR      PASSWORD=OPER3
           PROFILE      DSIPROFA
OPER4      OPERATOR      PASSWORD=OPER4
           PROFILE      DSIPROFA
OPER5      OPERATOR      PASSWORD=OPER5

CMD==>
TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'

```

Figure 22. Command Facility Message Panel

To end your operator OPER1 session, enter:
LOGOFF

at the cursor.

You return to a panel similar to Figure 23 (the command facility panel for operator NETOP1).

```

NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 07:53:23
* CNM01  BFSESS CNM01
- CNM01  DSI498I SRCLU HAS BEEN DEFAULTED TO SRCLU = TF01#000 FOR FLSCN
        SESSION TO APPLID = CNM01

- CNM01  DSI496I FLSCN SESSION BETWEEN APPLID = CNM01  AND SRCLU =
        TF01#000 ENDED

-----

???
```

Figure 23. Command Facility Panel for Operator NETOP1

Use the CLEAR command to clear the messages from the command facility panel.

The common global CNMSTYLE.STYLE is set to the name of the CNMSTYLE member read during NetView initialization. To list the active CNMSTYLE member, enter:

```
QRYGLOBL COMMON VARS=CNMSTYLE.STYLE
```

to see a panel similar to Figure 24.

```
NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 07:58:19
* CNM01  QRYGLOBL COMMON VARS=CNMSTYLE.STYLE
' CNM01
BNH031I NETVIEW GLOBAL VARIABLE INFORMATION
BNH103I COMMAND ISSUED AT: 11/08/01 07:58:19
BNH061I
BNH032I COMMON GLOBAL VARIABLES
BNH036I GLOBAL VARIABLE NAME:          GLOBAL VARIABLE VALUE:
BNH061I -----
BNH039I CNMSTYLE.STYLE                 CNMSTYLE
BNH035I NUMBER OF VARIABLES FOUND: 1
BNH061I
BNH037I NETVIEW GLOBAL VARIABLE INFORMATION COMPLETE
-----

???
```

Figure 24. Command Facility Panel to Display the Active CNMSTYLE Member

Use the BROWSE NETLOGA command list to browse the active network log.

Enter:

BR NETLOGA

to see a panel similar to Figure 25.

```
STATMON.BROWSE      ACTP  NETWORK LOG FOR 11/08/01 (01312) COLS 017 094 07:59 A
                                                                SCROLL ==> CSR
--2---+---3---+---4---+---5---+---6---+---7---+---8---+---9---
CNM01  07:49:51 C BR DSIOPF
  CNM01  07:52:39 * LOGOFF
CNM01  07:52:39 - DSI081I OPERATOR OPER1, LOGOFF PROCEEDING: TERMINAL = TF01
CNM01  07:52:43 - DSI496I FLSCN SESSION BETWEEN APPLID = CNM01 AND SRCLU
  CNM01  07:58:19 * QRYGLOBL COMMON VARS=CNMSTYLE.STYLE
CNM01  07:58:19 ' BNH031I NETVIEW GLOBAL VARIABLE INFORMATION
CNM01  07:58:19 ' BNH103I COMMAND ISSUED AT: 11/08/01 07:58:19
CNM01  07:58:19 ' BNH061I
CNM01  07:58:19 ' BNH032I COMMON GLOBAL VARIABLES
CNM01  07:58:19 ' BNH036I GLOBAL VARIABLE NAME:          GLOBAL VARIABL
CNM01  07:58:19 ' BNH061I -----
CNM01  07:58:19 ' BNH039I CNMSTYLE.STYLE                 CNMSTYLE
CNM01  07:58:19 ' BNH035I NUMBER OF VARIABLES FOUND: 1
CNM01  07:58:19 ' BNH061I
CNM01  07:58:19 ' BNH037I NETVIEW GLOBAL VARIABLE INFORMATION COMPLETE
CNM01  % 07:58:53 - DUI432I AN UNSUCCESSFUL CALL TO INITAPI HAS BEEN MADE BY T
CNM01  07:59:25 * BROWSE NETLOGA

CMD==>
TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
```

Figure 25. Active Network Log for the NetView Program Panel

On the top line of Figure 25, the letters ACTP tell you that you are browsing the active primary network log.

Low system activity can cause the data presented in the log panel to lag a few moments behind real events in the network. The time lag is greater with low system activity because message buffers must fill before messages are written to the log.

Press PF3 to return to the command facility. Use the CLEAR command to clear the messages from the command facility panel.

Use the SWITCH command to switch network logs. Enter:

```
SWITCH DSILOG,S
```

to activate the secondary network log.

Use the CLEAR command to clear the messages from the command facility panel.

If you are not running another copy of the NetView program on this LPAR, enter the VTAM commands:

```
F NET,PPOLOG=YES
```

and

```
D NET,APPLS
```

from the NetView operator terminal.

You see a panel similar to Figure 26.

```
NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 08:03:48
* CNM01  F NET,PPOLOG=YES
  CNM01  IST097I  MODIFY  ACCEPTED
  CNM01  IST223I  MODIFY          COMMAND COMPLETED
* CNM01  D NET,APPLS
  CNM01  IST097I  DISPLAY  ACCEPTED
' CNM01
IST350I  DISPLAY TYPE = APPL MAJ NODES/NAMES
IST089I  VTAMSEG TYPE = APPL SEGMENT      , ACTIV
IST360I  APPLICATIONS:
IST080I  ISTATA00 CONCT      ISTNOP  ACTIV      ISTPDCLU ACTIV
IST080I  NMP120  ACTIV
IST089I  A01APPC TYPE = APPL SEGMENT      , ACTIV
IST360I  APPLICATIONS:
IST080I  BASELU  CONCT
IST089I  A01APPLS TYPE = APPL SEGMENT      , ACTIV
IST360I  APPLICATIONS:
IST080I  CNM01  ACTIV      CNM01PPT ACTIV      CNM01000 ACTIV
IST080I  CNM01001 ACTIV      CNM01002 ACTIV      CNM01003 ACTIV
IST080I  CNM01004 ACTIV      CNM01005 ACTIV      CNM01006 ACTIV
IST080I  CNM01007 ACT/S      CNM01008 CONCT      CNM01009 CONCT
IST080I  CNM0100A CONCT      CNM0100B CONCT      CNM0100C CONCT
IST080I  CNM0100D CONCT      CNM0100E CONCT      CNM0100F CONCT
IST080I  TAF010PT CONCT      TAF01000 CONCT      TAF01001 CONCT
IST080I  TAF01002 CONCT      TAF01003 CONCT      TAF01004 CONCT
IST080I  TAF01F00 CONCT      TAF01F01 CONCT      TAF01F02 CONCT
IST080I  TAF01F03 CONCT      TAF01F04 CONCT      TAF01F05 CONCT
?? * **
```

Figure 26. VTAM Message Display Panel

To continue, press the Clear or ENTER key.

The VTAM command and resulting VTAM messages are passed across the PPOLOG interface to the NetView system and added to the network log. The messages do not pass across the PPOLOG interface if you are running another copy of the NetView program in this LPAR because the NetView program is the secondary program operator (SPO), not the primary program operator (PPO). When you browse the active network log again, you see the VTAM messages resulting from the command just entered.

Testing the Hardware Monitor

Use the CLEAR command to clear the messages from the command facility panel. Start the hardware monitor by entering the following command:

```
STARTCNM NPDA
```

You should see results similar to Figure 27.

```
NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 08:06:13 * CNM01
STARTCNM NPDA
| CNM01  START TASK=BNJDSERV
| CNM01  START TASK=BNJMNPDA
| CNM01  START TASK=CNM01LUC
| CNM01  START TASK=DSICRTR
- CNM01  DSI166I BNJDSERV IS ACTIVATED BY NETOP1
- CNM01  DSI166I BNJMNPDA IS ACTIVATED BY NETOP1
+ CNM01  BNJ014I NPDA LOGGER IS ACTIVE
- CNM01  DSI166I CNM01LUC IS ACTIVATED BY NETOP1
- CNM01  DSI166I DSICRTR IS ACTIVATED BY NETOP1
Z CNM01  DSI530I 'DSICRTR ' : 'DST' IS READY AND WAITING FOR WORK
Z CNM01  DSI530I 'CNM01LUC' : 'DST' IS READY AND WAITING FOR WORK
Z CNM01  DSI530I 'BNJDSERV' : 'DST' IS READY AND WAITING FOR WORK
C CNM01  CNM354I STARTCNM : COMMAND COMPLETE
C CNM01  **** SETTING HARDWARE MONITOR FILTERS
C CNM01  CNM354I PDFILTER : COMMAND COMPLETE
- CNM01  DSI530I 'CNMCALRT' : 'ALERT RECEIVER TASK' IS READY AND WAITING FOR
      WORK
-----
???
```

Figure 27. Command Facility Panel for the STARTCNM NPDA Command

To go to the hardware monitor, enter:

```
NPDA
```

You see a panel similar to Figure 28 on page 69.

```
N E T V I E W          SESSION DOMAIN: CNM01  NETOP1  11/08/01 08:07:39
NPDA-01A              * MENU *              HOST DOMAIN: CNM01

SEL#  PRODUCES:
( 1)  ALERTS-DYNAMIC DISPLAY
( 2)  TOTAL EVENTS DISPLAY
( 3)  TOTAL STATISTICAL DATA DISPLAY
( 4)  HELP MENU DISPLAY

      REQUEST DATA FROM NETWORK RESOURCES:
( 5)  SNA CONTROLLERS (CTRL)
( 6)  MODEMS AND ASSOCIATED LINKS (TEST)

                        DATA TYPES INITIALIZED/PURGED
AL.. (05/08/01)  EV.. (05/08/01)  ST.. (05/08/01)  GMFALERT.. (05/08/01)

ENTER SEL#

???
CMD==>
```

Figure 28. Hardware Monitor Main Menu

Examine the dates in the AL.., EV.., ST.. and GMFALERT.. fields. The dates should not be asterisks. Do not be concerned if your dates do not match the dates in Figure 28.

Use the ALERTSH (ALH) command to display the alerts recorded on the hardware monitor database. To see a panel similar to Figure 29 on page 70, enter:
ALH

This panel displays alerts recorded on the hardware monitor sample database.

If no alerts have been recorded, you receive message BNJ925I.

```

N E T V I E W          SESSION DOMAIN: CNM01  NETOP1  11/08/01 08:13:45
NPDA-31A              * ALERTS-HISTORY *          PAGE  1 OF  1

SEL# DOMAIN RESNAME TYPE TIME  ALERT DESCRIPTION:PROBABLE CAUSE
( 1) CNM01@ATCHM019 ISDN 08:03 THRESHOLD HAS BEEN REACHED:ISDN NTKW COMPONENT %
( 2) CNM01 PU3725  *LINE 08:03 COMM SUBSYSTEM FAILURE:COMM SUBSYST CTRL      %
( 3) CNM01 ATCHM001 COMC 08:02 INITIALIZATION FAILURE:RING SUBSYSTEM ATTACHMENT %
( 4) CNM01 ATCHM002 COMC 08:02 EQUIPMENT MALFUNCTION:COMMUNICATIONS          %
( 5) CNM01 ATCHM003 COMC 08:02 INITIALIZATION FAILURE:RING INTERFACE COUPLER  %
( 6) CNM01 ATCHM004 LCTL 08:02 LINK TIMEOUT:ADJACENT PRODUCT                 %
( 7) CNM01 ATCHM005 LCTL 08:02 ERROR TO TRAFFIC RATIO EXCEEDED:                %
( 8) CNM01 ATCHM006 LCTL 08:02 NO DATA RECEIVED:DEVICE OFF/MODEM OFF/COMM    &
( 9) CNM01 ATCHM007 LINE 08:02 NO LPDA RESPONSE RECEIVED:LOCAL MODEM          %
(10) CNM01 ATCHM008 CTRL 08:02 LINK ERROR:REMOTE DCE INTERFACE CABLE          %
(11) CNM01 DCHANNP PORT 08:01 PROBLEM RESOLVED:DS1 NETWORK COMPONENT          %
(12) CNM01 ATCHM013 CTRL 08:01 PROBLEM RESOLVED:SOFTWARE PROGRAM              %
(13) CNM01 ATCHM016 ISDN 08:00 THRESHOLD HAS BEEN REACHED:ISDN NTKW COMPONENT %
(14) CNM01@ATCHM018 CP   08:00 THRESHOLD HAS BEEN REACHED:ISDN NTKW COMPONENT %
(15) CNM01@ATCHM032 PORT 08:00 ERROR TO TRAFFIC RATIO EXCEEDED:COMMUNICATIONS %

ENTER SEL# (ACTION),OR SEL# PLUS M (MOST RECENT), P (PROBLEM), DEL (DELETE)

???
CMD==>

```

Figure 29. Alerts History Panel

Press PF3 twice or End twice to return to the command facility panel.

Testing the Session Monitor

Use the CLEAR command to clear the messages from the command facility panel. Start the session monitor by entering the following command:

```
STARTCNM NLDM
```

You should see results similar to Figure 30.

```

NetView V5 - NM          Tivoli NetView  CNM01 NETOP1  11/08/01 08:11:02 * CNM01
STARTCNM NLDM
| CNM01  START TASK=AAUTCNMI
| CNM01  START TASK=AAUTSKLP
| CNM01  START TASK=DSIAMLUT
- CNM01  DSI166I AAUTCNMI IS ACTIVATED BY NETOP1
- CNM01  DSI166I AAUTSKLP IS ACTIVATED BY NETOP1
- CNM01  DSI166I DSIAMLUT IS ACTIVATED BY NETOP1
Z CNM01  DSI530I 'AAUTCNMI' : 'DST' IS READY AND WAITING FOR WORK
Z CNM01  DSI530I 'DSIAMLUT' : 'DST' IS READY AND WAITING FOR WORK
Z CNM01  DSI530I 'AAUTSKLP' : 'DST' IS READY AND WAITING FOR WORK
C CNM01  CNM354I STARTCNM : COMMAND COMPLETE
- CNM01  AAU136I SESSION STAT FUNCTION IS NOW ACTIVE
- CNM01  % BNH163I THE CNM01 ABOVE 16M STORAGE IS 81% USED, 6096K IS LEFT
-----

???

```

Figure 30. Command Facility Panel for the STARTCNM NLDM Command

To go to the session monitor, enter:

NLDM

at the command line. You see a panel similar to Figure 31 if you have initialized NLDM.

```
NLDM.MENU                                PAGE  1
                                         NETVIEW
                                         DOMAIN  CNM01

SEL#                                     DESCRIPTION
( 1)  LUNAME LIST  LIST OF ALL ACTIVE LOGICAL UNIT NAMES
( 2)  SLUNAME LIST LIST OF ACTIVE SECONDARY LOGICAL UNIT NAMES
( 3)  PLUNAME LIST LIST OF ACTIVE PRIMARY LOGICAL UNIT NAMES
( 4)  PUNAME LIST  LIST OF ACTIVE PHYSICAL UNIT NAMES
( 5)  CPNAME LIST  LIST OF ACTIVE CP AND SSCP NAMES
( 6)  DOMAIN LIST  LIST OF NLDM DOMAINS
( 7)  ER LIST      LIST OF ACTIVE EXPLICIT ROUTES
( 8)  VR LIST      LIST OF ACTIVE VIRTUAL ROUTES

ENTER: H OR HELP FOR INFORMATION ON THE USE OF NLDM
      HELP NLDM COMMANDS FOR NLDM COMMAND LIST

      NLDM FILE LAST INITIALIZED 05/08/01

ENTER SEL# OR COMMAND
CMD==>
```

Figure 31. Session Monitor Main Menu

The date on the field “NLDM File Last Initialized” reflects the sample database.

To display a historical listing of logical units, enter:

LIST HISTORY LU

to see a panel similar to Figure 32 on page 72.

```

NLDM.LIST                                     PAGE 1
                                RESOURCE NAME LIST
LIST TYPE: HISTORY LU                                DOMAIN: CNM01
-----
SEL#   NAME   STATUS   SEL#   NAME   STATUS   SEL#   NAME   STATUS
( 1)  A01A4A08 HISTORY
( 2)  A01A442 HISTORY
( 3)  A01A443 HISTORY
( 4)  A01A444 HISTORY
( 5)  A01A445 HISTORY
( 6)  CNM01   HISTORY
( 7)  CNM01LUC HISTORY
( 8)  CNM01000 HISTORY
( 9)  CNM01001 HISTORY
(10)  CNM02   HISTORY
(11)  CNM02LUC HISTORY
(12)  CNM20LUC HISTORY
(13)  ECH001  HISTORY

END OF DATA - TYPE FIND NAME TO LOCATE SPECIFIC NAME
ENTER SEL# (SESS LIST), SEL# RTS (RESP TIME SUM) OR SEL# RTT (RESP TIME TREND)
CMD==>

```

Figure 32. LIST HISTORY LU Panel

Press PF3 twice or End twice to return to the command facility panel.

To return to NetView's main menu, enter:

MAINMENU

at the command line.

```

CNM1NETV          Tivoli NetView for z/OS Version 5          Main Menu

                    Operator ID = NETOP1   Application = CNM01007

Enter a command (shown highlighted or in white) and press Enter.

Browse Facility          BROWSE command
Command Facility        NCCF command
News                    NEWS command
PF Key Settings         DISPFK command
Help Facility           HELP command
Index of help topics    INDEX command
Help Desk               HELPDESK command
Hardware Monitor      NPDA command
Session Monitor      NLDM command

To log off or disconnect  LOGOFF command or DISC command

TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
Action===>

```

Figure 33. NetView Main Menu Panel

Note: Notice that the hardware monitor and session monitor have now been added to the list of possible components to select. Compare this figure to Figure 13 on page 58.

Preparing for Production Test

This completes installation of NetView with minimum function. To run NetView in production, consider the following:

- Ensure that the V5R1 modules are active in the system, and that the V5R1 VTAMLIB members are in use by VTAM. This could require an IPL with CLPA before running the NetView program in production.
- Allocate the VSAM for the production LPAR.
- After starting VTAM, rerun the status monitor preprocessor CNMNDEF.
- If you have been running multiple NetView programs in the same LPAR, then make sure that one NetView program is set up as the primary program operator (PPO) and the second NetView is set up as the secondary program operator (SPO).
- Complete any tuning and customization tasks your system requires. See Table 11 for more information.
- If you are using NetView for system automation, review your system automation planning and verify that any new operating procedures are ready for implementation.

For each administration task that you prepared, test to ensure that it was done correctly. When you are satisfied, the NetView program is ready for full production.

Note: If you resume production under a previous release of the NetView program, cancel the NetView subsystem job and close the V5R1 application.

Table 11. Additional Installation, Configuration, Customization, and Tuning Information

If you want information about...	Refer to...
Updating NetView for your environment	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
Updating NetView for graphics	<i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i>
Writing installation exits	<i>Tivoli NetView for z/OS Customization: Using Assembler or Tivoli NetView for z/OS Customization: Using PL/I and C</i>
Writing command processors	<i>Tivoli NetView for z/OS Customization: Using Assembler or Tivoli NetView for z/OS Customization: Using PL/I and C</i>

Appendix A. NetView Functions and Tasks

The following table contains some of the NetView functions with information on how they benefit you and where more information can be found.

Table 12. Benefits of Individual NetView Functions

NetView Function	Benefit to You	Where Documented in NetView Library
Application programming	Tailor or supplement NetView to satisfy unique requirements or operating procedures. Command procedures can be written in PL/I, C, assembler, REXX, or NetView CLIST languages. Installation exits can be written in PL/I, C, or assembler.	<ul style="list-style-type: none"> • <i>Tivoli NetView for z/OS Application Programmer's Guide</i> • <i>Tivoli NetView for z/OS Automated Operations Network Customization Guide</i> • <i>Tivoli NetView for z/OS Customization: Using Assembler</i> • <i>Tivoli NetView for z/OS Customization: Using PL/I and C</i> • <i>Tivoli NetView for z/OS Customization: Using REXX and the NetView Command List Language</i> • <i>Tivoli NetView for z/OS Customization: Using Pipes</i>
Automated operation for networks (AON)	Provides drop-in automation for TCP/IP, VTAM, and SNA using NetView AON automation policy	<ul style="list-style-type: none"> • <i>Tivoli NetView for z/OS Automated Operations Network User's Guide</i> • <i>Tivoli NetView for z/OS Automated Operations Network Customization Guide</i>
NetView automation	Perform repetitive or routine tasks without operator action	<i>Tivoli NetView for z/OS Automation Guide</i>
Browse facility ¹	View logs, data sets, and files on either a local or remote NetView	<i>Tivoli NetView for z/OS User's Guide</i>
Command facility ¹	Provides basic command support for the NetView product; lets you issue VTAM, MVS, and NetView commands from the NetView command line.	<i>Tivoli NetView for z/OS User's Guide</i>
Event/Automation service (EAS)	Manages distributed and S/390 [®] events from a single interface on a single console. Exploits event correlation and automation capability of the Tivoli Enterprise Console.	<ul style="list-style-type: none"> • <i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i> • <i>Tivoli NetView for z/OS User's Guide</i>
Extended multiple console support (EMCS) consoles	Provides a consistent MVS operation interface, an alternative to the subsystem interface (SSI)	<i>Tivoli NetView for z/OS Automation Guide</i>
Hardware monitor	Collects and stores data about failed resources in networks	<ul style="list-style-type: none"> • <i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i> • <i>Tivoli NetView for z/OS User's Guide</i>
Help facility ¹	Provides online help for messages and commands	<i>Tivoli NetView for z/OS User's Guide</i>
Help Desk facility ¹	Provides online help for diagnosing problems	<i>Tivoli NetView for z/OS User's Guide</i>

Table 12. Benefits of Individual NetView Functions (continued)

NetView Function	Benefit to You	Where Documented in NetView Library
MultiSystem Manager	Provides an integrated, centralized network management facility that enables you to manage your non-SNA network resources from a single workstation	<ul style="list-style-type: none"> • <i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i> • <i>Tivoli NetView for z/OS MultiSystem Manager User's Guide</i>
NetView 3270 management console	Provides access to the NetView console from UNIX, HP-UX, Windows NT®, Windows® 2000, and Sun Solaris, freeing you from the need for a 3270 emulator session	<i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i>
NetView management console	Displays topology and status information about network and system resources in a graphical format, from any Java-enabled platform	<i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i>
NetView Resource Manager	Enables you to graphically monitor and manage NetView task resource utilization and status using NMC. You can monitor all NetView programs in your enterprise via one NMC	<i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i>
NetView Web application	Enables you to connect to the NetView program from a web browser	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
Program-to-Program Interface (PPI)	Enables user programs to send or receive data buffers from other user programs; also allows system and application programs to send alerts to the hardware monitor	<i>Tivoli NetView for z/OS Application Programmer's Guide</i>
Remote operations	Interact with remote SNA systems without the overhead of cross-domain logons. (The receiver can be an unattended system.)	RMTCMD in the <i>Tivoli NetView for z/OS Command Reference</i>
Resource Object Data Manager (RODM)	Provides a central location for storing, retrieving, and managing operational resource information	<ul style="list-style-type: none"> • <i>Tivoli NetView for z/OS Installation: Configuring Graphical Components</i> • <i>Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer's Guide</i>
Session monitor	Collects and stores data about SNA resources in subarea, APPN, and mixed networks.	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
SNA topology manager	Obtains the status and topology information for SNA subarea and SNA APPN resources dynamically, for graphical display using NMC	<i>Tivoli NetView for z/OS SNA Topology Manager Implementation Guide</i>
Status monitor	Collects status information about SNA resources in the network	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
Subsystem interface (SSI)	Enables MVS operators to send commands to NetView and NetView operators to receive output from commands sent to MVS through the SSI; also enables you to monitor MVS operations, because unsolicited messages are sent to NetView through the SSI	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
TSO command server	Supports operations and procedures that issue commands through TSO.	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>
UNIX command server	Enables UNIX commands to be entered from the NetView command line and returns output to the NetView console.	<i>Tivoli NetView for z/OS Installation: Configuring Additional Components</i>

Table 12. Benefits of Individual NetView Functions (continued)

NetView Function	Benefit to You	Where Documented in NetView Library
Notes:		
1. No additional steps are required before using this function.		

Table 13 lists the optional NetView tasks and how they are started.

Table 13. Optional NetView Tasks

Task	Description	NetView Samples Set Up to Start by Default?
AAUTCNMI	Passes data to and from the communication network management interface (CNMI). This task is used by the session monitor.	N
AAUTSKLP	Writes data to the session monitor database.	N
ALIASAPL	Creates aliases for VTAM resources	N
BNJDSERV	Processes events and alerts. This task is used by the hardware monitor and the 4700 support facility.	N
BNJDSE36	Writes 4700 support facility data to the hardware monitor database.	N
BNJMNPD	Receives certain types of locally generated event and statistical records. This task is used by the hardware monitor.	N
CNMCALRT	Receives alerts through the PPI	Y
CNMCSSIR	Routes commands and messages to appropriate NetView tasks	Y
CNMTAMEL	Receives status changes for resources and forwards them to NMC.	N
<i>domainid</i> BRW	Provides the log-browse facility. The <i>domainid</i> is the NetView domain identifier.	Y
<i>domainid</i> LUC	Provides communications between the functions of NetView and their counterparts in other NetView domains. The <i>domainid</i> is the NetView domain identifier. This task is used by the hardware monitor, session monitor, and NMC.	N
<i>domainid</i> VMT	Provides communication between NetView and VTAM. The <i>domainid</i> is the NetView domain identifier. This task is used by the status monitor and NMC.	N
DSIAL2WS	Provides alert dynamic function for the SNA/6000 workstation	N
DSIAMLUT	Receives session awareness and other data from VTAM. This task is used by the session monitor.	N
DSIATOPT	Performs disk write services for the AUTOTEST command	N
DSICRTR	Routes data from the CNMI to the appropriate NetView component.	N
DSIDB2MT	Provides a DB/2 environment	N
DSIELTSK	Provides external SMF logging support	N
DSIGDS	Provides a CNM interface for communication with resources in the network such as 3710, 586x modems, 786x modems, and service points using the RUNCMD command.	N

Table 13. Optional NetView Tasks (continued)

Task	Description	NetView Samples Set Up to Start by Default?
DSIHPDST	Provides high performance LU 6.2 communications for sending and receiving large amounts of data.	Y
DSIIPLOG	Provides system logging for remote users	N
DSIKREM	Communicates with 3172 and 3174 network controllers. This task is used by the central site control facility (CSCF).	N
DSILOG	Writes data to the network log.	Y
DSIMCAT	Provides MVS command management	Y
DSIQTSK	Provides facilities for communicating with the Resource Object Data Manager (RODM).	N
DSIROVS	Provides support for programmable network access (PNA) to send commands to downstream devices and receive records from these devices.	N
DSIRQJOB	Ensures that NetView is known to JES	N
DSIRSH	Enables the RSH server	N
DSIRTR	Allows NetView to receive alerts over a TCP/IP connection	N
DSIRXEXC	Enables the REXEC server	N
DSISVRT	Defines the save/restore area for timers, global variables, PNA registrations and focal point information	Y
DSITCPIP	Provides a TCP/IP environment for NetView	N
DSITRACE	Writes data to the NetView trace data set.	N
DSIUDDST	Sends RMTCMD data to another NetView program and receives responses.	Y
DSIWBTSK	Starts the NetView Web server interface task.	N
DSI6DST	Provides management services (MS) transport function for sending and receiving management data.	Y
DUIDGHB	Gets the host name from the IP address	Y
DUIFSSCO	Provides command and span authorization checking for NMC commands and provides other NMC-to-NetView or NetView-to-NMC communication services.	N
EZLTCFG	Stores the NetView policy file definitions	Y
EZLTDDF	Provides DDF functions for AON	N
EZLTLOG	Provides automation logging functions for AON	N
EZLTSTS	Provides automation status functions for AON	N
VPDTASK	Solicits and logs vital product data from network inventory	N

Table 14 lists the automatic NetView tasks and how they are started.

Table 14. Automatic NetView Tasks

Task	Description	NetView Samples Set Up to Start by Default?
AUTOAMI	Autotask for Application Management Instrumentation	N
AUTOAON	Autotask to read policy definitions and initialize AON	Y
AUTONRM	Autotask for NetView Resource Manager	N
AUTOVTAC	Autotask for VTAM ACB monitor	N

Table 14. Automatic NetView Tasks (continued)

Task	Description	NetView Samples Set Up to Start by Default?
AUTOVTDB	Autotask for VTAM ACB monitor	N
AUTO1	Autotask for NetView initialization	Y
AUTO2	Autotask for NetView initialization	Y
BRIGOPER	Autotask for NetView Bridge	N
DBAUTO1	Autotask for NetView VSAM DB automation	N
DBAUTO2	Autotask for NetView VSAM DB automation	N
DSILCOPR	Autotask for focal point autodial	N
DSIMCAOP	Autotask for MVS command processing function	Y
DSINVGR	Autotask for NetView generic receiver	N
DSIWEB	Autotask for Web server interface functions	Y
DUIFCSGW	Autotask for common operations services (COS) gateway functions for NMC.	N
DUIFEAUT	Autotask for GMFHS	N
DUIFPOLI	Autotask for GMFHS to process NMCSTATUS policy definitions	Y
FLBTOPO	Autotask to provide SNA topology manager support for subarea and APPN resources.	N
REMOPER	Autotask for NetView Bridge	N

Notes:

1. For AON autotasks associated with the AON tower, see the DSIOPF %INCLUDE member EZLOPF.
2. For AON autotasks associated with the AON.SNA subtower, see the DSIOPF %INCLUDE member FKVOPF.
3. For AON autotasks associated with the AON.TCP subtower, see the DSIOPF %INCLUDE member FKXOPF.
4. For MSM autotasks associated with the MSM tower, see the DSIOPF %INCLUDE member FLCSOPF.

Table 15 lists internal tasks which perform vital NetView services. These tasks automatically start when NetView starts. You should not routinely start or stop them.

Table 15. Internal NetView Tasks

Task	Description
DSIDCBMT	NetView data set management
DSIHLLMT	NetView high-level language, preinitialized environments
DSILOGMT	NetView log browse synchronization
DSIMONIT	NetView task resource monitoring
DSISTMMT	NetView STOP and MODIFY processing
DSITIMMT	NetView main task auxiliary processing

Appendix B. NetView Samples Overview

All of the sample definitions listed in this book (and some not listed) are contained on the NetView distribution tape. These definitions are distributed as members of the following data sets:

- NETVIEW.V5R1M0.DSIPARM
- NETVIEW.V5R1M0.DSIPRF
- NETVIEW.V5R1M0.CNMSAMP
- NETVIEW.V5R1M0.SDSIOPEN

Each sample's name begins with *AAU*, *BNJ*, *CNMB*, *CNMS*, *DSI*, *DUIF*, *EKG*, *EZL*, *FKV*, *FKW*, *FKX*, or *FLC*, and can have a maximum length of 8 characters. The last four or five characters of all samples vary. Table 16 summarizes the samples distributed with the NetView program.

Table 16. Summary of Sample Members

Name	Description
AAUxxxxx	Session monitor samples
BNJxxxxx	Hardware monitor samples
BNJ36xxx	4700 support facility samples
CNMBnnn	Sense code descriptions
CNMSAAnn	SAA [®] DM members
CNMSInnn	MVS IDCAMS statement members
CNMSJHnn	GMFHS installation samples
CNMSJInn	MVS IEBCOPY procedures and control statements
CNMSJMnn	Miscellaneous MVS samples
CNMSJVnn	IDCAMS verify statements
CNMSJnnn	MVS installation samples
CNMSnVSM	Sample VSAM load data
CNMS0nnn	VTAM and NCP definitions for sample network
CNMS1Ann	External logging support samples
CNMS42nn / CNMS43nn	High-level language (HLL) PL/I samples
CNMS42nn / CNMS44nn	High-level language (HLL) C samples
CNMS42nn / CNMS45nn	Assembler samples
CNMS6nnn	Automated operations samples
CNMS71nn	IBM Network Configuration Application/MVS samples
CNMS80nn	REXX samples
DSIxxxxx	Command facility samples
DUIFxxxx	GMFHS samples
EKGxxxxx	RODM samples
EZLxxxxx	AON samples
FKVxxxxx	AON/SNA samples
FKXxxxxx	AON/TCP samples

Table 16. Summary of Sample Members (continued)

Name	Description
FLCxxxx	MSM samples

“Appendix C. NetView Samples Reference” on page 103 lists all the samples on the distribution tape and a brief description of each sample.

CNMSJ003 allows you to copy these files into the correct libraries where they can be used by the NetView and VTAM programs. Sometimes, you must rename the files to meet NetView or VTAM program requirements. For example, VTAM start options are distributed on the samples tape in CNMS0010. However, the VTAM program requires that you name this file ATCSTR00. CNMSJ003 copies CNMS0010 into the VTAM library as ATCSTR00.

If you want information about...	Refer to...
HLL samples	<i>Tivoli NetView for z/OS Customization: Using PL/I and C</i>
Assembler samples	<i>Tivoli NetView for z/OS Customization: Using Assembler</i>
REXX samples	<i>Tivoli NetView for z/OS Customization: Using REXX and the NetView Command List Language</i>
Automation samples	<i>Tivoli NetView for z/OS Automation Guide</i>

The sample network uses a standard naming convention for the NCP and VTAM definitions.

NCP Samples

A Network Control Program (NCP) is a program load module that resides in a communication controller. You create the load module by coding an NCP generation definition (also called a generation deck) and by generating the NCP. You generate an NCP by assembling and link-editing the generation definition.

The System Support Program (SSP), which is a package that resides in a host, provides for generation of the NCP. NCP/EP definition facility (NDF) is an SSP utility that validates the generation definition, assembles the NCP tables, and creates link-edit statements for the NCP. The linkage editor then creates the NCP load module.

If you want information about...	Refer to...
NCP/EP definition facility (NDF)	<i>NCP, SSP, and EP Generation and Loading Guide</i>

Coding an NCP

Coding an NCP involves defining such things as binary synchronous communication (BSC) and synchronous data link control (SDLC) lines, the terminals controlled by the NCP, the paths from the NCP to other subareas in the network, and the hosts attached to the controller.

In the NCP, you should also code several statements that are needed by the VTAM program. Although these statements are included in the NCP generation definition,

they have meaning only to the VTAM program; NCP ignores them. For example, PCCU is a VTAM-only definition statement that identifies the communication controller where the NCP is to be loaded.

When you install the NetView program you should also define a time-out frame for LPDA-2 data circuit-terminating equipment (DCEs) operating in LPDA-2 mode. This time-out frame should be in the range of 40–120 seconds.

Once the NCP has been generated, you can load and start the NCP from the VTAM program. Before the VTAM program can load or start an NCP, however, it must have access to the file containing the NCP generation definition. The file must be a member of the data set pointed to by the VTAMLST DD statement (usually SYS1.VTAMLST). If you follow this convention, you can use the same NCP generation definition file for both the source input to generate the load module and the major node that the VTAM program needs to start the NCP.

NCP Definition

The NCP samples included with the NetView program as part of the sample network are NCP generation definitions. To use the NCP samples:

1. Rename the NCP samples copied to VTAMLST to a name other than the name given to the NCP samples during installation. For example, rename A04A54C to A04A54B.
2. Run NDF against the NCP samples to create a member in VTAMLST by the name that was originally in VTAMLST. For example, run NDF against A04A54B and create a member named A04A54C in VTAMLST.

The NCP definition samples included with the NetView program are A04A54C (CNMS0065), A31A54C (CNMS0066), and B30A54C (CNMS0109). All of these samples are used with a 3745 Communication Controller.

NCPs A04A54C and B30A54C are connected through a back-to-back configuration or null network. While null networks can exist as part of a link, the status of B30A54C and A04A54C cannot be monitored, therefore, the status is always unknown.

VTAM Major Node Names and Definitions

This section lists the names of the VTAM major nodes in the sample network. The sample MVS IEBCOPY member CNMSJ003 copies and renames the samples to the names of the VTAM major nodes for all subareas in the sample network. During the NetView installation process, the sample MVS IEBCOPY job CNMSJ003 uses input file CNMSJI03 to copy the NETA sample VTAM definition files from the library where they are distributed (NETVIEW.V5R1M0.CNMSAMP) to a user library which can be used as input to your VTAM program.

The sample MVS IEBCOPY job CNMSJ003 also uses input file CNMSJI21 to copy and rename the NETB sample VTAM definition files from the library where they are distributed (NETVIEW.V5R1M0.CNMSAMP) to a user library which can be used as input to your VTAM program.

Table 17 on page 84 lists the VTAM major nodes at A01MPU in network NETA. Table 18 on page 85 lists the VTAM major nodes at B01MPU in network NETB.

Note: The VTAM major nodes at A02MPU and A99MPU are very similar to the VTAM major nodes at A01MPU, and therefore, are not discussed in this appendix.

This section also describes the network definitions for A01MPU, one of the hosts in the sample network. A01MPU resides in network NETA for this book. This book does not provide detailed information on the definition process or syntax. For this type of information, refer to the appropriate product's resource definition or administration reference manual.

VTAM Major Node Names

Table 17 lists the names of the VTAM major nodes in the sample network (NETA).

Table 17. Summary of Sample VTAM Major Nodes at A01MPU in Network NETA

Name	Distributed As	Description
ATCCON01	CNMS0003	Defines configuration start list
ATCSTR00 ATCSTR01	CNMS0010 CNMS0007	Defines VTAM start options
A01ADJ	CNMS0012	Defines adjacent SSCPs for VTAM to query
A01APPLS	CNMS0013	Defines the NetView program application major node
A01CDRM	CNMS0014	Defines cross-domain resource managers in NETA
A01CDRM1	CNMS0088	Defines cross-domain resource managers in NETB
A01CDRSC	CNMS0015	Defines cross-domain resources
A01CDRS1	CNMS0089	Defines independent LU cross-domain resources
A01LOCAL	CNMS0016	Defines local non-SNA terminals
A01MVS	CNMS0047	Defines MVS applications
A01NVAS	CNMS0087	Defines the NetView access services applications
A01PATH	CNMS0018	Defines path table for host A01MPU
A01SNA	CNMS0073	Defines local SNA terminals
A01USER	CNMS0041	Defines various application programs
A04NTRI	CNMS0061	Defines the dial deck coded for NTRI lines
CNMCON01	CNMS0084	Defines major nodes not activated during VTAM initialization
CTCA0102	CNMS0038	Defines a channel-to-channel attachment
CTNA0104	CNMS0081	Defines a channel-to-NCP channel attachment

Table 18 on page 85 lists the names of the VTAM major nodes in the sample network (NETB). The actual samples are not shown in this book, however, their NETA equivalents are listed in the table for easy reference.

Table 18. Summary of Sample VTAM Major Nodes at B01MPU in Network NETB

Name	Distributed As	Description	NETA Equivalent
ATCCONB1	CNMS0102	Defines configuration start list	ATCCON01
ATCSTRB1	CNMS0101	Defines VTAM start options	ATCSTR01
B01ADJ	CNMS0104	Defines adjacent SSCPs for VTAM to query	A01ADJ
B01APPLS	CNMS0113	Defines the NetView program application major node	A01APPLS
B01CDRM	CNMS0106	Defines cross-domain resource managers in NETA	A01CDRM
B01CDRSC	CNMS0107	Defines cross-domain resource managers in NETB	A01CDRSC
B01LOCAL	CNMS0108	Defines local non-SNA terminals	A01LOCAL
B01MVS	CNMS0111	Defines MVS applications	A01MVS
B01PATH	CNMS0105	Defines path table for host B01MPU	A01PATH
B30NTRI	CNMS0110	Defines dial deck coded for NTRI lines	A04NTRI
CNMCONB1	CNMS0103	Defines major nodes not activated during VTAM initialization	CNMCON01
CTNB0130	CNMS0115	Defines a channel-to-NCP attachment	CTNA0104

VTAM Definitions

This section describes the process of modifying the sample network definitions to fit your network. It focuses on VTAM definition statements and VTAM tables.

VTAM Definition Statements

After installing the NetView program, you can begin to define the network, starting with VTAM definitions. Among the items you should define are programs, devices, and paths to other nodes in the network. These are defined by coding major nodes.

A *major node* is a representation of a set of resources. Specifically, a major node is a set of VTAM definition statements. For MVS, a VTAM major node is a member of the partitioned data set identified on the VTAMLST DD statement in the VTAM start procedure.

Once you have coded the necessary major nodes, the VTAM operator can activate or deactivate resources in the network by activating and deactivating major or minor nodes. (A *minor node* is one resource in a major node.) For example, at A01MPU, local terminals are defined in the major node named A01LOCAL. Once the VTAM program is active, the VTAM operator can use the VARY ACT command. The VARY ACT command is:

```
v net,act,id=a01local
```

Conversely, the operator can deactivate these terminals by entering the VARY INACT command. This command is:

```
v net,inact,id=a01local
```

Configuration Lists

The major nodes you have defined are not processed by the VTAM program until you activate them. There are two ways to activate these nodes:

1. Have the VTAM operator enter a VARY ACT command for each major node.
2. Include the major node names in a configuration list. When you start the VTAM program, it automatically activates every node that you have listed if CONFIG=*xx* is coded in your start list, where *xx* is the last two characters of your configuration list name (ATCCON*xx*).

VTAM Start Options

VTAM start options define such things as buffer pools, major nodes to be activated when the VTAM program is started, the network name, and the identifier given to the VTAM program for the host subarea. The only VTAM start options that you need to code are SSCPID=*n*, NETID=*n*, and SSCPNAME=*n*. However, most customers need to specify some options, especially the host subarea (HOSTSA=*n*). All subareas within a network must be unique.

Code VTAM start options in a major node named ATCSTR*yy*, where *yy* is specified by the user. The default is 00. If you wish, you can code start options in more than one file. The VTAM program always processes the file named ATCSTR00. You can add additional start options in another file as was done in the sample network.

Start options that apply to all hosts in the network are coded in major node ATCSTR00. The VTAM program processes defaults first. Defaults are overridden by ATCSTR*yy*. Start options specific to host A01MPU are coded in major node ATCSTR01. Both nodes are processed when the VTAM program is started. ATCSTR00, the default, is always processed. ATCSTR01 is processed when the operator starts the VTAM program and identifies 01 as the start option list by entering:

```
s cnmnet.net,,, (list=01)
```

If the same option is coded in both ATCSTR00 and ATCSTR01, the option coded in ATCSTR01 takes precedence.

The VTAM start options for B01MPU in NETB are coded in ATCSTRB1. To start the VTAM program in host B01MPU, enter:

```
s cnmnet.net,,, (list=B1)
```

ATCSTR00 (CNMS0010) (VTAM Startup Options):

ATCSTR00, the default start option file, is on the product tape. If you want to use these definitions in your network:

1. Check the PPOLOG start option. This option is used to specify whether messages are to be sent to the network log. The sample network uses the default value of PPOLOG=NO. If you are running a release of the VTAM program before V3R3, change this value to PPOLOG=YES to keep the status monitor panels in the NetView program accurate.
2. Review the defaults for each start option. Defaults were coded for all start options except TRACE and IOBUF.
3. Check the TRACE start option. This option indicates which functions are traced when the VTAM program is started. OPT=ALL was coded here, meaning all functions are traced when the VTAM program is started. If you wish to conserve storage, delete the OPT=ALL operand or specify a specific function that you wish to trace.

4. Check the IOBUF start option. This option is the message pool in fixed storage and is used for input/output data. For the sample network, this option is set to 256 bytes. This represents the value of the UNITSZ operand on the NETA definition statements in A04A54C (CNMS0065) and A31A54C (CNMS0066). The value of the UNITSZ operand on the NETB definition statement in B30A54C (CNMS0109) is set to 384 bytes.

ATCSTR01 (CNMS0007) (Start Options for Host Subarea): ATCSTR01 is the second start option file coded for A01MPU in NETA. The start option file coded for B01MPU in NETB is ATCSTRB1 (CNMS0101). If you want to use these definitions in your network:

1. Change the HOSTPU operand to the physical unit name you want assigned to the VTAM program. In this example, the HOSTPU name is the SSCP name followed by the string PU (for example: A01MPU). The HOSTPU name appears in some NetView panels.
2. Change the HOSTSA option to match your host subarea.
3. Change the NETID to match your network name. NETID is a required parameter. If you code NETID using the NLDM.NETID statement in CNMSTYLE, the NETID must be the same as the start parameter you used for the VTAM program.
4. Change the SSCPID to a number that is unique within your network and across all interconnected networks. In this example, the host subarea is also used for the SSCPID.
5. Change the SSCPNAME to the name you wish to use for your SSCP. We have used the name A01M.
6. You may want to add the CSA24 start option. This start option is used to specify the maximum amount of 24-bit common storage area that the VTAM program can access.

If you want information about...

Refer to...

Changing the NETID

Tivoli NetView for z/OS Administration Reference for additional information.

Configuration Start List

The configuration list must be named ATCCON xx , where xx is defined by the user. You identify the file name to the VTAM program with a start option. For example, if you have coded **config=01**, the VTAM program expects to find a configuration list named ATCCON01. ATCCON01 is coded for host A01MPU. It contains all the major nodes that are activated when the VTAM program is initialized, except the NCPs (which can also be initialized but are not initialized in this example). If you intend to use this list in your own network, change the major node names to match the major nodes your host activates.

ATCCON01 (CNMS0003) (Configuration Start List): The configuration list coded for host A01MPU in NETA is included on the distribution tape. Sample configuration lists for A02MPU and A99MPU are also included on the distribution tape. They are ATCCON02 (CNMS0004) and ATCCON99 (CNMS0005), respectively. The configuration list coded for B01MPU in NETB is ATCCONB1 (CNMS0102).

CNMCON01 (CNMS0084) (Major Nodes Not Activated During VTAM

Initialization): The STATMON preprocessor CNMNDEF (CNMSJ007) uses this major node to inform the status monitor which resources at this VTAM host (A01MPU) were not activated during VTAM initialization. The STATMON preprocessor also informs the status monitor that these resources need to be

monitored. If you want to use this sample in your network, change the node names to match the major nodes (in your host) that should not be activated during VTAM initialization.

Also included on the tape are equivalent samples for A02MPU and A99MPU. They are CNMCON02 (CNMS0085) and CNMCON99 (CNMS0086) respectively. The major node coded for B01MPU in NETB is CNMCONB1 (CNMS0103).

Application Programs

Each VTAM application program must be defined with an APPL definition statement. The statement is coded in an application major node defined with a VBUILD statement (TYPE=APPL).

In the sample network, the following major nodes are available for host A01MPU:

A01APPLS

Defines application programs common to all operating systems. APPL statements for the NetView program are in this major node.

A01MVS

Defines MVS-based application programs, such as TSO and NetView Performance Monitor (NPM).

A01NVAS

Defines NetView access services application major node.

A01USER

Defines additional application programs, such as IMS™ and CICS®

A detailed description of each application major node follows.

A01APPLS (CNMS0013) (NetView APPL Definitions for Subarea): A01APPLS is the first APPL major node coded for A01MPU. The application major node coded for B01MPU in NETB is B01APPLS (CNMS0113). If you want to use these definitions in your network, perform these steps:

1. For each APPL statement, check the names of the logon mode table and default logon mode. The sample network uses a logon mode table called AMODETAB and a variety of default logon modes.
2. Check the description in each STATOPT operand. The lines beginning with STATOPT are operands used by the status monitor component of the NetView program. The VTAM program interprets these lines as comments, so they have no effect if you do not use the NetView program. The STATOPT operands control the functions of the status monitor and are coded in the major and minor node members in VTAMLST. The STATOPT operands are displayed on the status monitor panel.
3. If you plan to have more than 10 concurrent NetView users, add additional CNM01*nnn* and TAF01*nnn* APPL statements.

If you want information about...	Refer to...
How to code the STATOPT operand	<i>Tivoli NetView for z/OS Administration Reference</i>

A01MVS (CNMS0047) (MVS Unique APPL Definitions):

A01MVS is an APPL major node coded for host A01MPU. The APPL major node coded for NETB is B01MVS (CNMS0111). If you want to use these definitions in your network, complete the following steps:

1. If you plan to have more than 10 concurrent TSO users, add additional TSOA01*nn* APPL statements.
2. This major node contains APPL definitions for NPM. The sample network does not contain NPM. This definition is included only as an example. If you do not plan to run NPM in your network, you can remove the NPM APPL definition.
3. Add APPL statements for any other VTAM applications you plan to run.

A01NVA (CNMS0087) (Defines NetView APPL Statements): This major node contains APPL statements for the NetView access services (NVA) application. If you want to use these definitions in your network, complete the following steps:

1. If you are not using NVA, remove this sample. EMSMODE is a logon mode table shipped with the NVA product and is not a part of the samples shipped with the NetView program.
2. If you plan to have more than 10 concurrent users for NVA, add additional PSA01*nnn* APPL statements.

A01USER (CNMS0041) (Defines APPL Samples): This major node contains definitions for other VTAM application programs, such as IMS and CICS.

Local Non-SNA Terminals

For this book, host A01MPU in the sample network has six local non-SNA 3277 terminals that you must define to the VTAM program. The definition is done with a local non-SNA major node. This major node consists of an LBUILD statement followed by a LOCAL statement for each terminal. A01MPU's local non-SNA major node is named A01LOCAL (CNMS0016). The local non-SNA terminal major node coded for NETB is B01LOCAL (CNMS0108). The terminals defined by A01LOCAL are illustrated by Figure 34 on page 91.

If you want to use these definitions in your network, complete the following steps:

1. Verify that the terminal types match the non-SNA terminals in your network. Check the model numbers and the use of the extended data stream feature. (The sample network uses 3277 Model 2 terminals, which use the extended data stream feature.)
2. Check the control unit address for each terminal. The sample network uses addresses 701 through 706. You may choose these addresses or change them to those available on your system.
3. Check the names of the default logon mode and logon mode table. The sample network uses a default logon mode called M2BSCQ in a logon mode table called AMODETAB.
4. Check the name of the USS table. The sample network uses a USS table called AUSSTAB.
5. Check the description in the NetView STATOPT operand to make sure it is appropriate for the device. This description is displayed in status monitor panels.

Local SNA Devices

Besides local non-SNA terminals, host A01MPU has a channel-attached SNA control unit with six terminals and two printers attached to it. These resources must be defined to the VTAM program with a local SNA major node. This major node consists of a VBUILD TYPE=LOCAL statement, followed by a PU statement for the control unit and an LU statement for each device.

The major node name is A01SNA (CNMS0073). The SNA and non-SNA devices defined by A01LOCAL and A01SNA are illustrated by Figure 34 on page 91.

If you want to use these definitions in your network, complete the following steps:

- For the PU statement:
 1. Check the control unit address (CUADDR). For the sample network, the address is 7A0.
 2. Check the names of the default logon mode (DLOGMOD) and logon mode tables (MODETAB). In this example, the default logon mode is M23278I and the logon mode table is AMODETAB.
 3. Check the name of the USS table. A01MPU uses a USS table called AUSSTAB.
 4. Check the description in the NetView STATOPT operand. What you code on this operand is displayed on status monitor panels.
- For each LU statement:
 1. Make sure the local address (LOCADDR) is appropriate.
 2. If necessary, change the default logon mode. In this example, the default logon mode for the LUs at addresses 5 and 9, which are printers, has been changed to the correct mode. These devices use a logon mode named M3287SCS, which is for 3287 sequential character stream (SCS) printers. The other LUs use the default defined in the preceding PU statement.
 3. Check the description in the NetView STATOPT operand.
 4. Check the SSCPFM operand. If the device supports character-coded messages in its communication with the VTAM program, code SSCPFM=USSSCS. Otherwise it defaults to SSCPFM=FSS, which is for formatted commands only.

The following major node defines the local SNA terminals at A01MPU.

A01SNA (CNMS0073) (SNA Channel-Attached Terminals): Figure 34 on page 91 illustrates the local SNA and non-SNA devices defined by A01LOCAL and A01SNA.

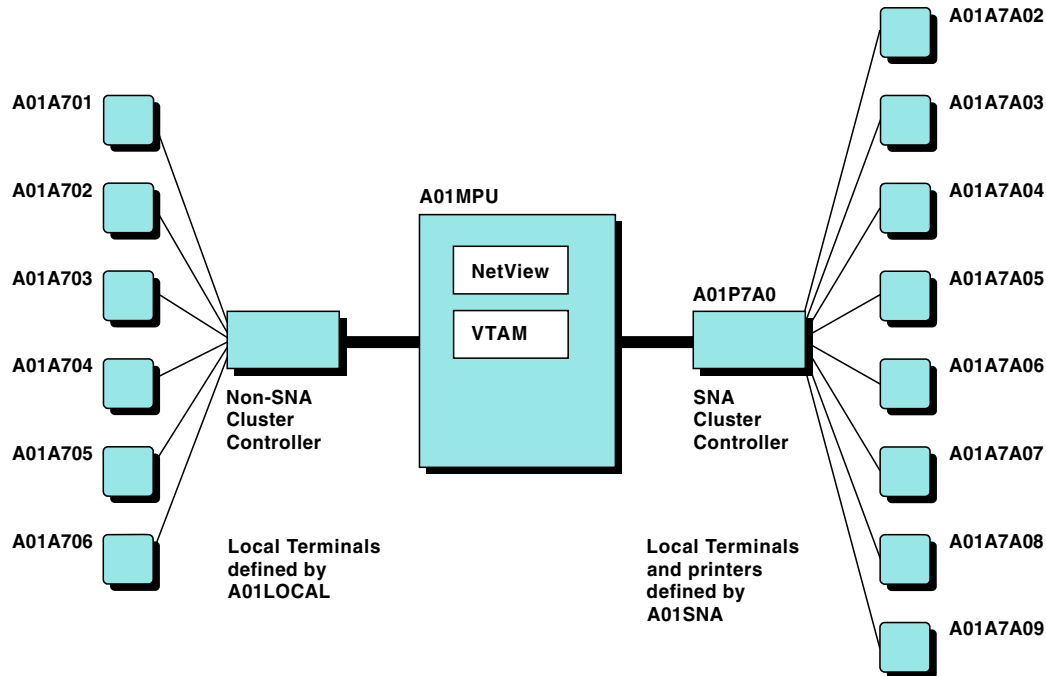


Figure 34. Local Devices Defined at A01MPU

Cross-Domain Resources

The logical units (LUs) in other domains that the VTAM program accesses are called *cross-domain resources* (CDRSCs). Some cross-domain resources are applications that reside in other hosts, such as CICS and TSO. A terminal in another domain can also be a CDRSC.

CDRSC definitions are not required. CDRSCs can dynamically define the owner of a CDRSC by polling other hosts in the network.

If you wish to define CDRSCs, you can do so in one or more CDRSC major nodes. The major node consists of a VBUILD statement (TYPE=CDRSC), followed by a CDRSC statement for each cross-domain resource.

The following sections define the cross-domain resources for A01MPU. The NETB equivalent cross-domain resource major node is B01CDRSC (CNMS0107).

A01CDRSC (CNMS0015) (Cross-Domain Resource Definitions):

In this example, only the NetView programs at subareas 2 (CNM02) and 99 (CNM99) are defined as CDRSCs. In this example, the owning CDRM for the NetView program is not defined. The VTAM program uses an adjacent SSCP table to locate the owning CDRM. See "Adjacent SSCP Table" on page 92 for additional information.

A01CDRS1 (CNMS0089) (Independent LU CDRSC Major Node): The cross-domain resource sample A01CDRS1 (CNMS0089) contains the independent LUs defined in the sample network for host A01MPU.

Cross-Domain Resource Managers

The owner of a CDRSC is called a *cross-domain resource manager* (CDRM). A CDRM is the VTAM program in another host.

Identify the CDRMs in all the other hosts with which your host needs to communicate. For A01MPU, these are A02M, A99M, and B01M. You should also define the CDRM component of the VTAM program in your local host. Therefore, a CDRM statement is required for A01M, A02M, A99M, and B01M for A01MPU.

Define each CDRM with a CDRM definition statement. The statement is coded in a CDRM major node that begins with a VBUILD statement (TYPE=CDRM).

It is not necessary to define B01M to A02MPU and A99MPU because they communicate through A01MPU.

If you wish to use the sample definitions, you should complete the following steps:

1. Change the NETID in the NETWORK statement to the name of your network. You can omit NETWORK statements if you do not use SNA Network Interconnection (SNI).
2. In each CDRM statement, change the label to match the name of the host being defined. In this sample and in most circumstances, use a CDRM name to match the host's SSCP name. (Hosts define their SSCP names as a start option.)
3. Also, in each CDRM statement, ensure the SUBAREA number is correct.
4. Change the ISTATUS statements for the external CDRMs (A02M, A99M, and B01M in the sample) from ACTIVE to INACTIVE if you do not want SSCP-to-SSCP sessions to be automatically established when this major node is activated.
5. Adjust the VPACING operand if it does not meet the needs of your network.
6. Note that CDRDYN=YES and CDRSC=OPT are coded for each CDRM. Code these operands to enable dynamic definition of cross-domain resources.
7. If necessary, add more CDRM statements. In most circumstances you should have a CDRM statement for every host you wish to communicate with (inside or outside your network).

A01CDRM (CNMS0014) (Cross-Domain Resource Manager): A01CDRM defines the cross-domain resource managers in NETA to A01MPU. The CDRM major node coded for B01MPU in NETB is B01CDRM (CNMS0106).

A01CDRM1 (CNMS0088) (Cross-Domain Resource Manager): This major node defines the cross domain-resource manager in NETB to A01MPU. A01CDRM1 is an example of an interconnected network CDRM.

When these major nodes and the CDRM minor nodes are activated, A01M sets up sessions with the remote VTAMs—A02M, A99M, and B01M.

If you want information about...	Refer to...
To see examples of CDRM definitions in interconnected networks	VTAM library

Adjacent SSCP Table

An adjacent SSCP table is a list of SSCPs you want the VTAM program to query when the resource is unknown. If you code an adjacent SSCP table, you can omit all or selected CDRSC statements. You can also code a CDRSC statement without

defining the CDRM that owns the resource. The CDRSC would be particularly useful in networks where applications are frequently moved from one host to another.

The adjacent SSCP table filed at B01MPU in NETB is B01ADJ (CNMS0104). If you want to use this table in your own network, modify it as follows:

1. As with CDRMs, you can delete the NETWORK statement if you do not intend to use SNA Network Interconnection (SNI). If you do use SNI, you may want to add another adjacent SSCP table for each network. The sample network, however, uses one adjacent SSCP table for all CDRMs in NETA and NETB.
2. For each ADJCDRM, change the name to match the name of the hosts that are adjacent to your host.

A01ADJ (CNMS0012) (Adjacent SSCP Table):

When the VTAM program searches for a CDRSC whose location is unknown, the VTAM program queries every SSCP in the list, in sequence, until it locates the resource or until it reaches the end of the list. If the VTAM program does not have a current session with an SSCP in this list, that SSCP is not queried.

Figure 35 on page 94 illustrates how default SSCP routing works. This figure illustrates what occurs when a user at host A01MPU enters:

cnm99

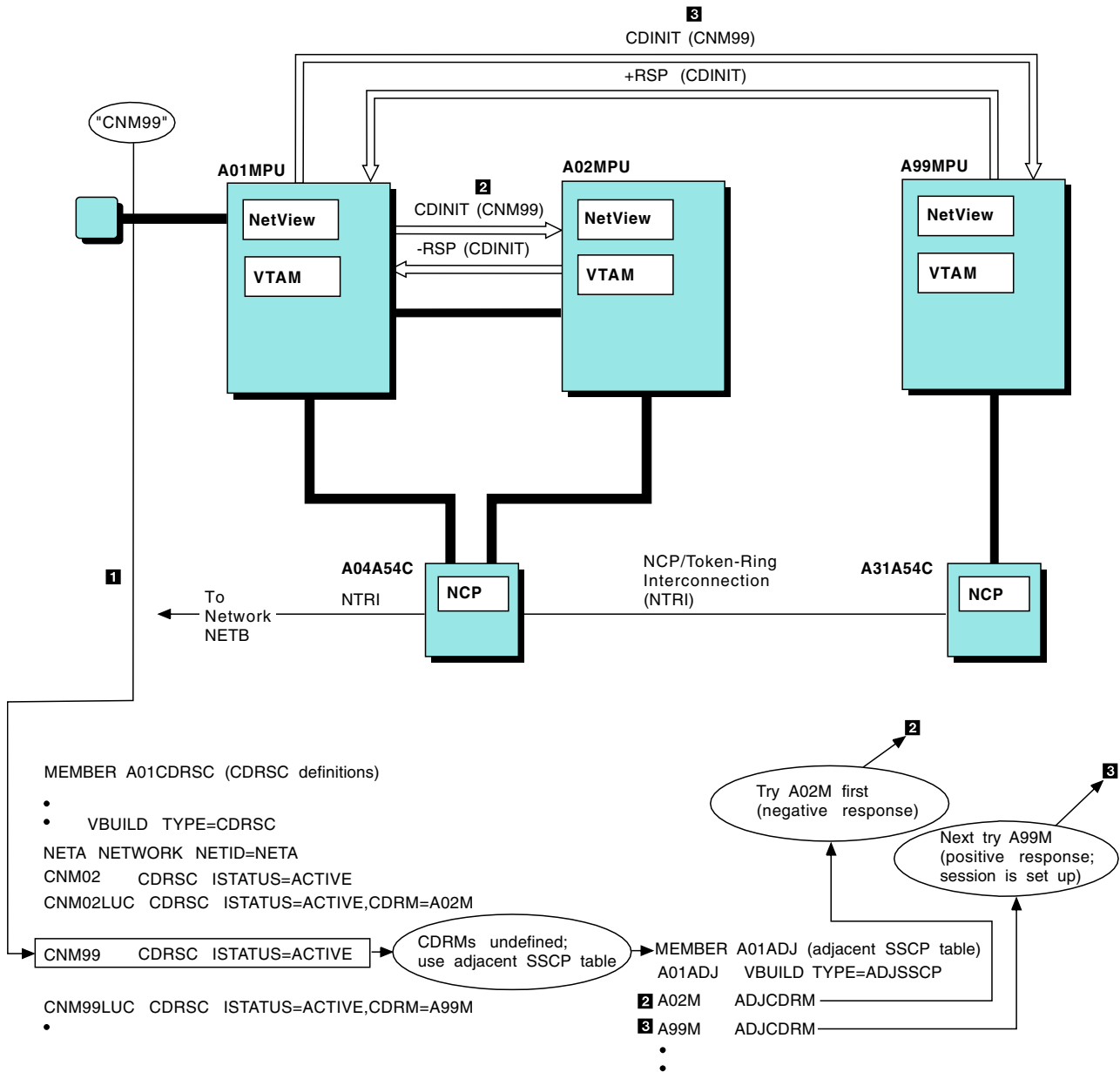


Figure 35. Use of Adjacent SSCP Table

When the VTAM program searches its list of cross-domain resources, it tries to locate an entry for CNM99. Because there is no indication where this resource resides, the VTAM program checks an adjacent SSCP table to see if adjacent SSCPs are defined. The VTAM program then uses the table to try to locate CNM99.

The first SSCP in the list is A02M. The VTAM program sends a session initiation request (CDINIT) to A02M, seeking to establish a session with CNM99. A01M has already established an SSCP-to-SSCP session with A02M. Therefore, the VTAM program knows which route to use.

A02M reacts to the initiation request by searching its resources for CNM99. A02M does not find CNM99 and returns a negative response. After receiving the negative response, the VTAM program tries the next SSCP in the list, which is A99M. Here, A99M owns CNM99 and accepts the initiation request.

Dynamic CDRSC definition increased the overhead required to set up this session. Explicit definition of the cross-domain resource allows the VTAM program to route the session request to its owner immediately.

This increase in overhead is minor and occurs only during initial session setup. For many sites, the increased overhead might be offset by the advantages of dynamic CDRSC definition. With fewer CDRSCs, less coding is required. Also, when an application moves to a different host you do not have to change CDRSC statements.

If you want information about...	Refer to...
Adjacent SSCP tables	The VTAM library

NCP/Token-Ring Interconnection (NTRI)

The NCP/Token Ring Interconnection (NTRI) allows a communication controller to attach to an IBM token-ring network using one physical medium, one attachment scheme, and one communication protocol.

The NCP at subarea 04, A04A54C, has NTRI line links that allow an NCP to connect to a token ring. To the VTAM program, NTRI links appear to be switched links and require a switched major node.

A04NTRI (CNMS0061) (Dial Deck for A04A54C NTRI Lines): A04NTRI defines the dial deck coded for A01MPU in NETA for A04A54C NTRI lines. The dial deck coded for B01MPU in NETB for B30A54C NTRI lines is B30NTRI (CNMS0110).

Channel-to-Channel Attachment

As the illustration of the network in Figure 36 on page 96 shows, A01MPU has a channel-to-channel attachment to A02MPU. This attachment must be defined to the VTAM program in a channel attachment major node. The channel attachment major node consists of a VBUILD statement (TYPE=CA) followed by GROUP, LINE, and PU statements.

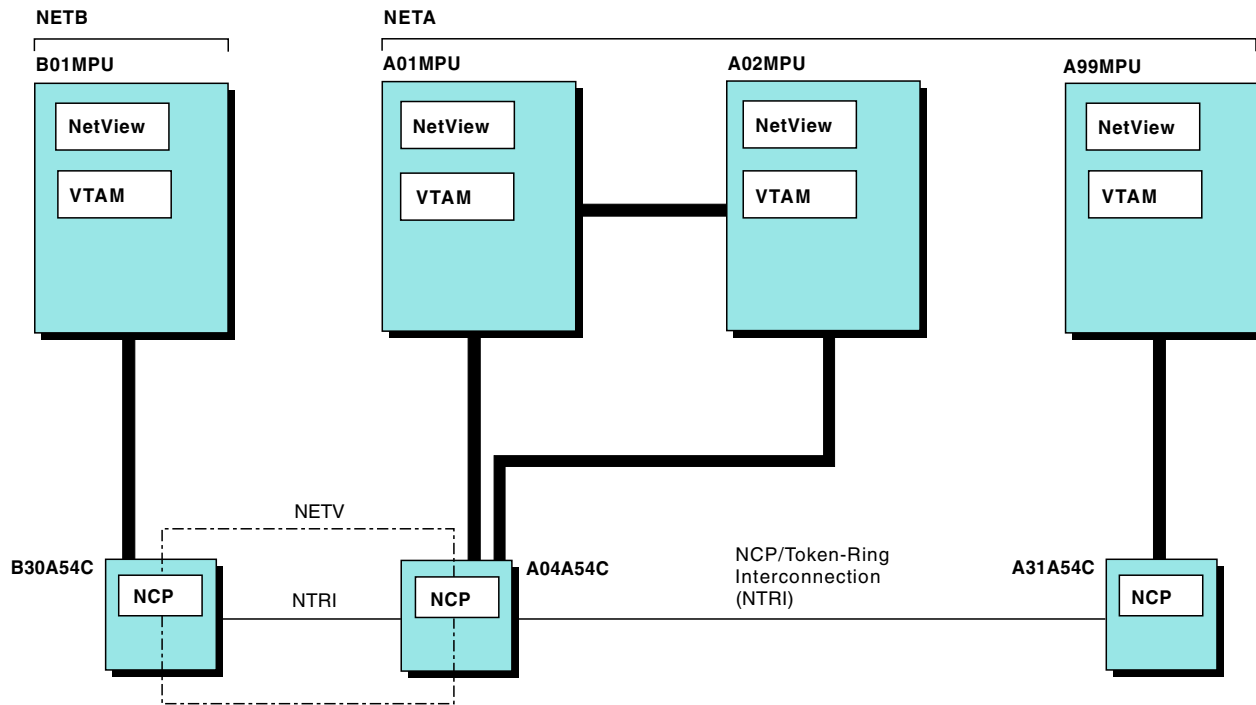


Figure 36. Network (NETA and NETB)

The major nodes named CTCA0102 (CNMS0038) and CTCA0201 (CNMS0039) define the channel attachment between A01MPU and A02MPU. If you wish to use these definitions in your own network, complete the following steps:

1. Check the DELAY and REPLYTO operands. Defaults were coded for each. The value specified in the DELAY operand is the time in seconds that the VTAM program in A01MPU must wait before sending low-priority data to the VTAM program in A02MPU. The value specified in the REPLYTO operand is the time the VTAM program must wait after completing a channel program.
2. Change the ADDRESS operand on the LINE statement to match the correct value for your network.
3. Verify the CTC address has been defined to the other host.

Channel-to-NCP Attachment

As the illustration of the sample network shows (see Figure 36), A01MPU has a channel-to-NCP attachment to A04A54C as coded in CTNA0104 (CNMS0081). If you wish to use these definitions in your own network, change the ADDRESS operand on the LINE statement to match the correct value for your network.

CTNA0104 (CNMS0081) (Channel-to-NCP Definition): The NCP channel attachment between A02MPU and A04A54C in NETA is coded in CTNA0204 (CNMS0082) and the NCP channel attachment between A99MPU and A31A54C in NETA is coded in CTNA9931 (CNMS0083). The channel-to-NCP definition coded for B01MPU in NETB is CTNB0130 (CNMS0115)

Paths

Once you know the physical structure of your network, remember to define the paths between the different nodes with PATH definition statements.

Figure 37 on page 97 illustrates the meaning of the PATH statement.

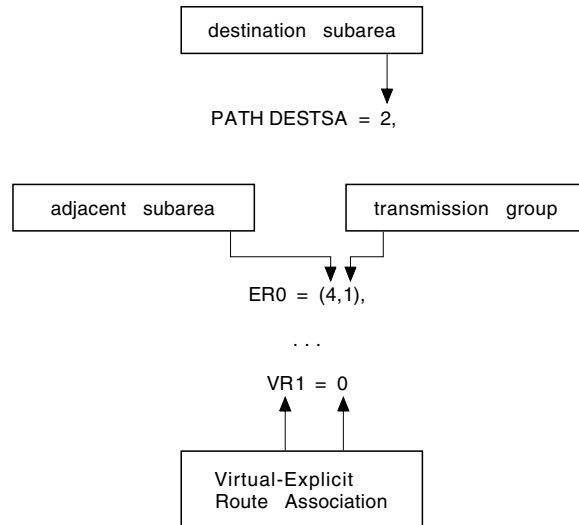


Figure 37. Meaning of PATH Statement

Code a PATH statement for each subarea (host or NCP) the VTAM program needs to communicate with. Generally, you should code a PATH statement for every subarea except your own.

The statement first defines the destination subarea. You can use the same statement for more than one subarea. Next, the statement must define the explicit routes and virtual routes (ERs and VRs) that make up that path.

An ER defines the physical elements that connect two subareas. A VR is the logical connection between two subareas. A VR uses the physical route defined by an ER. Associate each ER with a VR. ERs have the additional characteristic of a transmission priority.

The VTAM program needs both ERs and VRs to set up a session. The VTAM program begins with a VR and then uses the PATH table to determine which ER to which the VR maps. A class-of-service table tells the VTAM program which VR to use. In Figure 37, if the class of service says that VR1 is to be used, then the VTAM program uses the PATH statement and map that statement into ER0.

The PATH table defines the paths from A01MPU to subareas 2, 4, 31, and 99, which are the other subareas in the network. Figure 36 on page 96 illustrates these nodes. If you wish to use this table, make the following modifications to each statement:

1. Change the DESTSA operands to match the subareas in your network.
2. Change the ER and the VR operands to fit your own routing scheme.
3. Add the VRPWS xy operands to fit your network or use the defaults. The VRPWS xy operand defines the minimum and maximum window size for the specified virtual route and transmission priority.

You may want to use the Network and Design Analysis (NETDA) program to create your path tables.

A01PATH (CNMS0018) (Path Definitions for Host Subarea): The following defines the paths for A01MPU in NETA. The sample path tables A02PATH

(CNMS0025) and A99PATH (CNMS0036) for A02MPU and A99MPU in NETA are also included on the tape. The path definition for B01MPU in NETB is B01PATH (CNMS0105).

The arrows in Figure 38 show host A01MPU's explicit routes to destination subarea 2.

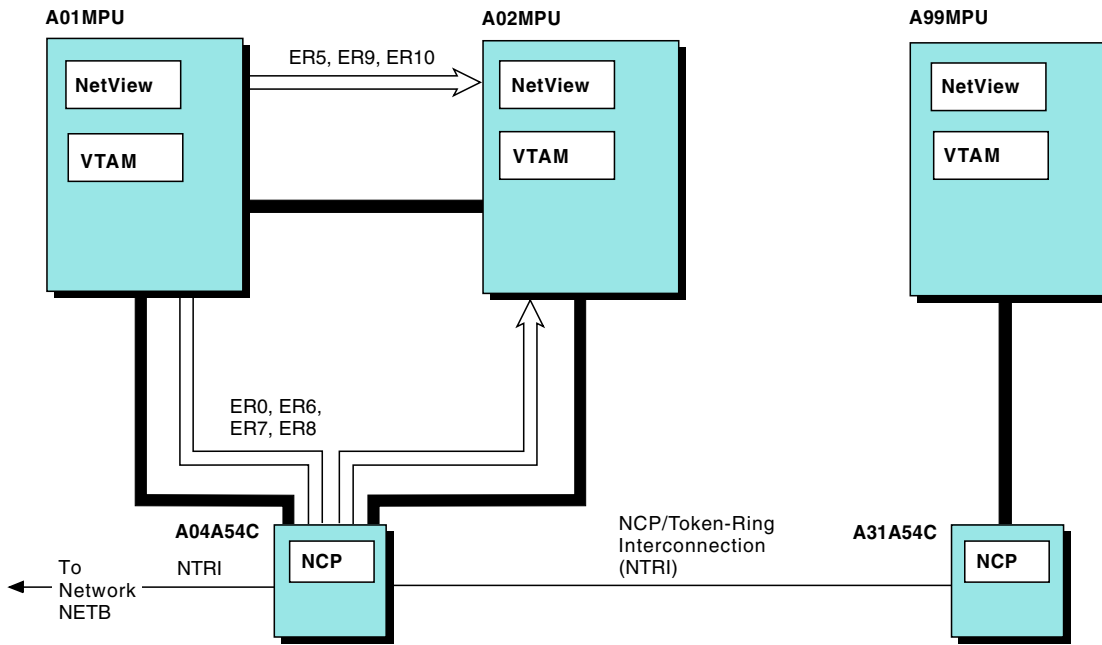


Figure 38. Explicit Routes Defined at A01MPU to Subarea 2

How a Route Is Selected

Figure 39 on page 99 shows how the logon mode and the class-of-service (COS) table, along with the VTAM definitions, are used to select the route for a session.

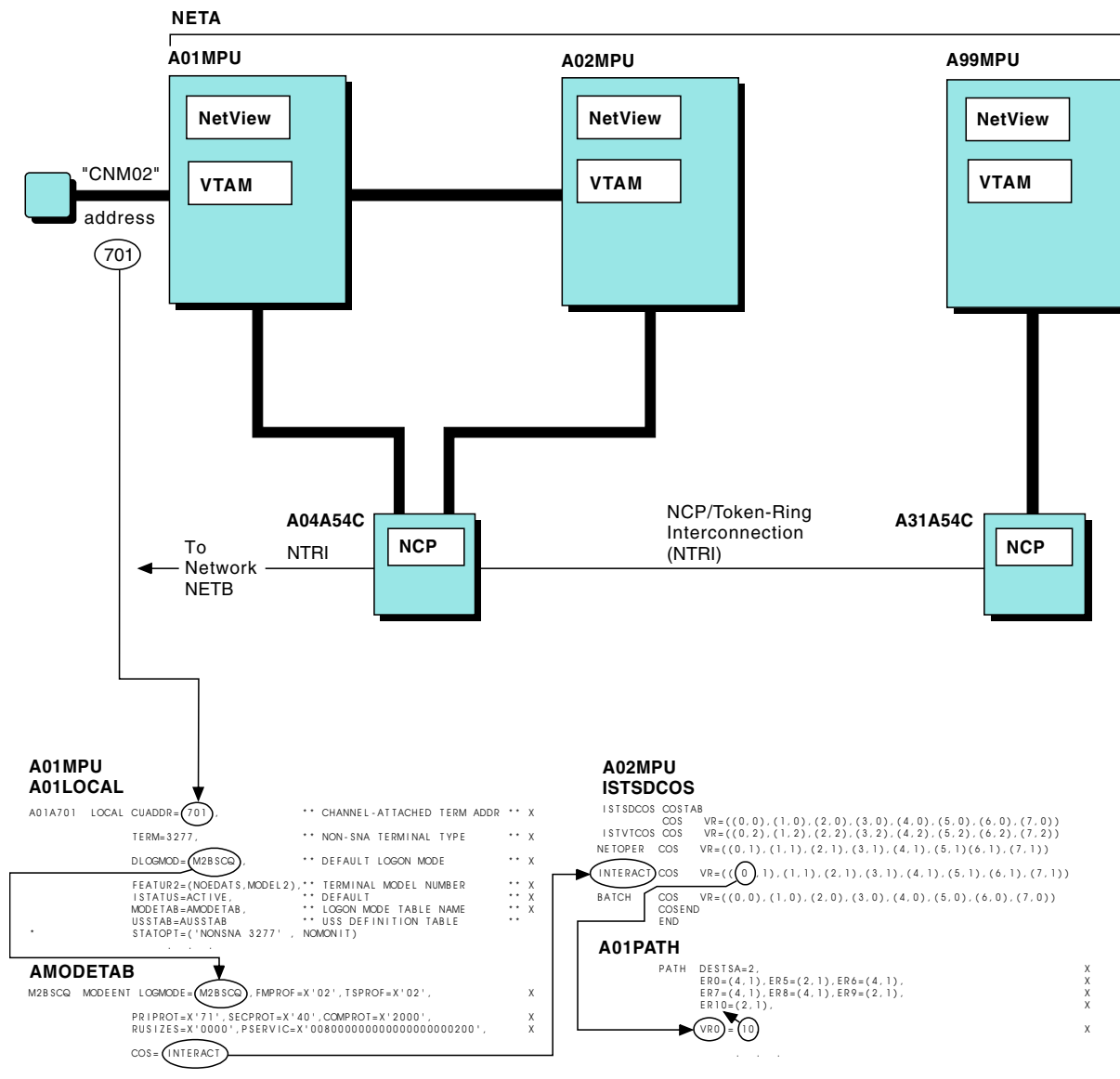


Figure 39. How an Explicit Route Is Selected

In this example, a user of a terminal at address 701 in A01MPU's domain wishes to establish a cross-domain session with the NetView program at A02MPU.

The user at A01MPU requests the session by entering:

```
cnm02
```

This is equivalent to entering:

```
logon applid(cnm02)
```

CNM02 is a USS command defined in A01MPU's USS table (AUSSTAB (CNMS0011)). The formatted logon specifies an application name of CNM02, which is the name given to the NetView program at subarea 2. CNM02 is also defined to the VTAM program at A02MPU as an APPL and to the VTAM program at A01MPU as a CDRSC.

The VTAM program at A01MPU carries out these steps when selecting the route:

1. A01MPU uses the USS table (not shown in Figure 39 on page 99) to format the logon.
2. The VTAM program locates the LOCAL definition statement for the terminal at address 701. As the figure illustrates, the definition for the terminal defines the logon mode table entry as M2BSCQ (DLOGMOD=M2BSCQ). The name of the logon mode table that contains this entry is also given (MODETAB=AMODETAB).
3. The VTAM program searches AMODETAB for the logon mode table entry named M2BSCQ. In Figure 39 on page 99, the class of service specified for that entry is INTERACT.

The VTAM program at A02MPU carries out these steps in selecting the route:

1. The VTAM program searches the class-of-service table for the class named INTERACT. The class-of-service entry defines the virtual route and transmission priority for the session. As Figure 39 on page 99 shows, the entry for INTERACT specifies that virtual route 0 with a transmission priority of 1 should be used.
2. The VTAM program uses its path table, specifically the definition of the path to subarea 2. As the COS entry specifies, virtual route 0 should be used for this session. The PATH statement maps virtual route 0 into explicit route 10 (VR0=10). Therefore the VTAM program uses explicit route 10 for this session. The PATH statement specifies the adjacent subarea and transmission group number for this ER. As the figure shows, for this ER, the VTAM program uses transmission group 1 (a channel) to subarea 2. The VTAM program activates the VR if required, completing the routing from A01MPU to A02MPU.

A02MPU accepts the session initiation request, and the user logs on to the NetView program.

VTAM Tables

Besides VTAM major nodes, you need to code the CNM routing table. You have the option of also coding the following tables:

- Unformatted system services (USS)
- Logon mode
- Class of service (COS)

The VTAM and NetView programs provide defaults or samples for each table. You can use the tables as they exist or customize them to fit the requirements of your network.

If you plan to modify one or more of these tables, copy the IBM sample and change it to meet your requirements. Then assemble and link-edit the table.

The following sections provide an overview of each table and include a description of how to assemble and link-edit a VTAM table.

If you want information about...

Refer to...

Coding these tables

The VTAM library

USS Table

You can use the USS table to define user (and operator) commands and messages. As an MVS user at subarea 01, remember to remove the VM and the VM01

commands. If hosts A02MPU and A99MPU are MVS hosts, remove the VM02 and VM99 commands. AUSSTAB (CNMS0011) is the USS table that is used for the sample network.

A default USS table is also shipped with the VTAM program.

If you want information about...	Refer to...
A listing of the default USS table that is shipped with the VTAM program	The VTAM library

Logon Mode Table

A logon mode table defines the session protocols for the different devices and applications in the network. The table also defines the terminal screen size and the class of service used for the session. Two logon mode tables were coded for the sample network:

AMODETAB (CNMS0001)

Supports various devices for the sample network

INTERCOS (CNMS0050)

Supports various applications

If you wish to use these tables you should assemble and link-edit them into NETVIEW.V5R1USER.VTAMLIB.

Note: You should not change RU sizes for a bisynchronous session.

Class-of-Service Table

A class-of-service (COS) table allows you to define the routes that the VTAM program uses in establishing cross-domain sessions. High-priority sessions might be assigned a class of service that uses high-speed routes. Conversely, low-priority sessions might be assigned a class of service that uses slower routes.

You define a class of service by creating an entry for it in a class-of-service table. You then specify the name of the entry in the logon mode table entry to create the routes used for the session.

The VTAM program does not require a class-of-service table. However, if you do not code one, the only classes of service you can use are the unnamed class of service and ISTVTCOS, which is the name the VTAM program uses to set up its own sessions.

For the sample network, the COS table has five classes of service:

- Unnamed class of service
- ISTVTCOS
- NETOPER
- INTERACT
- BATCH

All the classes of service have the same selection order but different transmission priorities. ISTVTCOS is always the class of service with the highest transmission priority. The selection order for the unnamed class of service is as follows:

1. Virtual Route 0, Transmission Priority 0
2. Virtual Route 1, Transmission Priority 0
3. Virtual Route 2, Transmission Priority 0
4. Virtual Route 3, Transmission Priority 0

5. Virtual Route 4, Transmission Priority 0
6. Virtual Route 5, Transmission Priority 0
7. Virtual Route 6, Transmission Priority 0
8. Virtual Route 7, Transmission Priority 0

Assembling and Link-Editing the Tables

Sample CNMSJ006, shipped with the NetView program, assembles and link-edits the USS, logon mode, and class-of-service tables.

Modifying the VTAM Start Procedure

A VTAM start procedure (CNMSJ008) is included in the samples. You can use the procedure as it is shipped, or you can modify it to meet the needs of your installation. The comments in the start procedure list some considerations for modifying the start procedure. Symbolic parameters are used for the region size, data set name qualifier, and the SYSABEND output class. For example, the data set name on the VTAMLST statement results in a data set name of NETVIEW.V5R1USER.CNM01.VTAMLST.

Appendix C. NetView Samples Reference

Table 19 lists the samples that can be found in the following libraries:

- NETVIEW.V5R1M0.CNMSAMP
- NETVIEW.V5R1M0.DSIPARM
- NETVIEW.V5R1M0.DSIPRF
- NETVIEW.V5R1M0.SDSIMSG1
- NETVIEW.V5R1M0.SDSIOOPEN

The table includes a brief description of each sample and the data set name where each member resides once installation is complete.

Note: The National Language Support Feature contains CNMMSJPN, the NetView sample for the Kanji feature program definition file for the sample network.

For information on RODM, GMFHS, SNA Topology Manager, and MultiSystem Manager samples, refer to *Tivoli NetView for z/OS Installation: Configuring Graphical Components*.

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 system.

Table 19. List of Samples

Distributed As	Name	Description	Data Set Name
AAUCNMTD	same	This sample contains initialization values for the session monitor's task AAUTCNMI. AAUTCNMI collects data from other network components such as the VTAM and NCP programs.	DSIPARM
AAUKEEP1	same	This sample defines KEEP classes that control the amount of data kept by the session monitor component of the NetView program. Also included are the MAPSESS statements that map sessions into KEEP classes.	DSIPARM
AAUPRMLP	same	Initialization values for the session monitor's task AAUTSKLP	DSIPARM
AAURTM1	same	This sample defines response time performance classes for the response time monitor (RTM). The performance class sets a standard for performing a certain percentage of the transactions in less than a specified response time.	DSIPARM
BNJ36DST	same	Initialization values for the 4700 support facility	DSIPARM
BNJMBDST	same	Initialization values for the hardware monitor's task BNJDSESV. Values for BNJMBDST are specified in CNMSTYLE and BNJMBDST should not be modified.	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMB000	same	Sample sense code	DSIPARM
CNMB001	same	Sample sense code	DSIPARM
CNMB002	same	Sample sense code	DSIPARM
CNMB080	same	Sample sense code	DSIPARM
CNMB081	same	Sample sense code	DSIPARM
CNMB082	same	Sample sense code	DSIPARM
CNMB083	same	Sample sense code	DSIPARM
CNMB084	same	Sample sense code	DSIPARM
CNMB085	same	Sample sense code	DSIPARM
CNMB086	same	Sample sense code	DSIPARM
CNMB087	same	Sample sense code	DSIPARM
CNMB088	same	Sample sense code	DSIPARM
CNMB089	same	Sample sense code	DSIPARM
CNMB08A	same	Sample sense code	DSIPARM
CNMB08B	same	Sample sense code	DSIPARM
CNMB100	same	Sample sense code	DSIPARM
CNMB101	same	Sample sense code	DSIPARM
CNMB200	same	Sample sense code	DSIPARM
CNMB201	same	Sample sense code	DSIPARM
CNMB400	same	Sample sense code	DSIPARM
CNMB401	same	Sample sense code	DSIPARM
CNMB402	same	Sample sense code	DSIPARM
CNMB800	same	Sample sense code	DSIPARM
CNMB801	same	Sample sense code	DSIPARM
CNMB802	same	Sample sense code	DSIPARM
CNMBA00	same	Sample sense code	DSIPARM
CNMBA01	same	Sample sense code	DSIPARM
CNMBAK1	same	Sample back-up command authorization table	DSIPARM
CNMBFF0	same	Sample sense code	DSIPARM
CNMBFF1	same	Sample sense code	DSIPARM
CNMBFF2	same	Sample sense code	DSIPARM
CNMBFF3	same	Sample sense code	DSIPARM
CNMBFF5	same	Sample sense code	DSIPARM
CNMBFF6	same	Sample sense code	DSIPARM
CNMBFF7	same	Sample sense code	DSIPARM
CNMBFF8	same	Sample sense code	DSIPARM
CNMBFF9	same	Sample sense code	DSIPARM
CNMBFFC	same	Sample sense code	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMBINDF	same	Sample sense code	DSIPARM
CNMBUNBD	same	Sample sense code	DSIPARM
CNMCAU00	same	MVS command management exclusion/inclusion table	CNMSAMP
CNMETSO	same	TSO command server CLIST	CNMSAMP
CNMGNETV	same	JPEG - NetView logo	SDSIOPEN
CNMGSPCR	same	JPEG - graphic spacer	SDSIOPEN
CNMGTIVL	same	JPEG - Tivoli logo	SDSIOPEN
CNMHELPF	same	Helpmap for NetView	DSIPARM
CNMJSCL	same	Java™ language "count lines" example	CNMSAMP
CNMJSHW	same	Java language "Hello World" example	CNMSAMP
CNMKEYS2	same	Alternate programmable key definitions for NetView applications	SDSIOPEN
CNMKEYS	same	Programmable key definitions for NetView applications	SDSIOPEN
CNMMIGKA	same	Programmable key definitions for NetView applications	SDSIOPEN
CNMMIGKV	same	Programmable key definitions for View applications	SDSIOPEN
CNMMSENU	same	NetView date/time message translation sample	SDSIMSG1
CNMNEWS	same	Customizable sample containing text that can be displayed to operators during log on	SDSIOPEN
CNMS0001	AMODETAB	Logon mode table for the sample network	VTAMLST
CNMS0003	ATCCON01	Configuration start list for host subarea 01	VTAMLST
CNMS0004	ATCCON02	Configuration start list for host subarea 02	VTAMLST
CNMS0005	ATCCON99	Configuration start list for host subarea 99	VTAMLST
CNMS0006	ATCCON00	Default configuration start list	VTAMLST
CNMS0007	ATCSTR01	Start options for host subarea 01	VTAMLST
CNMS0008	ATCSTR02	Start options for host subarea 02	VTAMLST
CNMS0009	ATCSTR99	Start options for host subarea 99	VTAMLST
CNMS0010	ATCSTR00	VTAM start options	VTAMLST
CNMS0011	AUSSTAB	VTAM USS table	VTAMLST
CNMS0012	A01ADJ	Adjacent SSCP table	VTAMLST
CNMS0013	A01APPLS	NetView APPL definitions for subarea 01	VTAMLST
CNMS0014	A01CDRM	Cross domain resource manager	VTAMLST
CNMS0015	A01CDRSC	Cross domain resource definitions	VTAMLST
CNMS0016	A01LOCAL	Non-SNA channel attached terminals	VTAMLST
CNMS0018	A01PATH	Path definitions for host A01MPU	VTAMLST
CNMS0019	A02ADJ	Adjacent SSCP table	VTAMLST
CNMS0020	A02APPL	NetView APPL definitions for subarea 02	VTAMLST

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS0021	A02CDRM	Cross domain resource manager	VTAMLST
CNMS0022	A02CDRSC	Cross domain resource definitions	VTAMLST
CNMS0023	A02LOCAL	Non-SNA channel attached terminals	VTAMLST
CNMS0025	A02PATH	Sample VTAM list - path definition for A02MPU	VTAMLST
CNMS0030	A99ADJ	Adjacent SSCP table	VTAMLST
CNMS0031	A99APPL	NetView APPL definitions for subarea 99	VTAMLST
CNMS0032	A99CDRM	Cross domain resource manager	VTAMLST
CNMS0033	A99CDRSC	Cross domain resource definitions	VTAMLST
CNMS0034	A99LOCAL	Non-SNA channel attached terminals	VTAMLST
CNMS0036	A99PATH	Sample VTAM list - path definition for A99MPU	VTAMLST
CNMS0038	CTCA0102	Sample VTAM list - CTC definition for A01MPU	VTAMLST
CNMS0039	CTCA0201	Sample VTAM list - CTC definition for A02MPU	VTAMLST
CNMS0040	ISTSDCOS	Sample VTAM list - class of service for all networks	VTAMLST
CNMS0041	A01USER	APPL definitions for host A01MPU	VTAMLST
CNMS0042	A02USER	APPL definitions for host A02MPU	VTAMLST
CNMS0043	A99USER	APPL definitions for host A99MPU	VTAMLST
CNMS0047	A01MVS	MVS-unique APPL definitions	VTAMLST
CNMS0048	A02MVS	MVS-unique APPL definitions	VTAMLST
CNMS0049	A99MVS	MVS-unique APPL definitions	VTAMLST
CNMS0050	INTERCOS	Sample VTAM list - log on mode table for NETA	VTAMLST
CNMS0055	same	Assembles and link-edits the NetView constants module (DSICTMOD)	CNMSAMP
CNMS0061	A04NTRI	Dial deck for NTRI lines	VTAMLST
CNMS0065	A04A54C	NCP generation definition	VTAMLST
CNMS0066	A31A54C	NCP generation definition	VTAMLST
CNMS0073	A01SNA	SNA channel attached terminals	VTAMLST
CNMS0074	A02SNA	SNA channel attached terminals	VTAMLST
CNMS0075	A99SNA	SNA channel attached terminals	VTAMLST
CNMS0081	CTNA0104	NCP channel attachment major node for A01MPU	VTAMLST
CNMS0082	CTNA0204	NCP channel attachment major node for A02MPU	VTAMLST
CNMS0083	CTNA9931	NCP channel attachment major node for A99MPU	VTAMLST
CNMS0084	CNMCON01	Lists major nodes and major nodes listed in ATCCON01.	VTAMLST

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS0085	CNMCON02	Lists major nodes and major nodes listed in ATCCON02.	VTAMLST
CNMS0086	CNMCON99	Lists major nodes and major nodes listed in ATCCON99.	VTAMLST
CNMS0087	A01NVAS	NetView Access Services APPL statements	VTAMLST
CNMS0088	A01CDRM1	Cross domain resource manager	VTAMLST
CNMS0089	A01CDRS1	Independent LU CDRSC major node for A01MPU	VTAMLST
CNMS0090	A99CDRS1	Independent LU CDRSC major node for A99MPU	VTAMLST
CNMS0091	A02NVAS	NetView Access Services APPL statements	VTAMLST
CNMS0092	A99NVAS	NetView Access Services APPL statements	VTAMLST
CNMS0101	ATCSTRB1	VTAM start options for host B01MPU	VTAMLST
CNMS0102	ATCCONB1	Configuration start list for host B01MPU	VTAMLST
CNMS0103	CNMCONB1	Lists major nodes in B01MPU and major nodes listed in ATCCON01.	VTAMLST
CNMS0104	B01ADJ	Adjacent SSCP table for host B01MPU	VTAMLST
CNMS0105	B01PATH	Path definitions for host B01MPU	VTAMLST
CNMS0106	B01CDRM	CDRM for host B01MPU	VTAMLST
CNMS0107	B01CDRSC	CDRSC major node for host B01MPU	VTAMLST
CNMS0108	B01LOCAL	Non-SNA channel attached terminals for B01MPU	VTAMLST
CNMS0109	B30A54C	NCP generation definitions for host B01MPU	VTAMLST
CNMS0110	B30NTRI	Dial deck for B30A54C NTRI lines	VTAMLST
CNMS0111	B01MVS	MVS APPL definitions	VTAMLST
CNMS0113	B01APPLS	NetView APPL definitions for host B01MPU	VTAMLST
CNMS0115	CTNB0130	NCP channel attachment major node for host B01MPU	VTAMLST
CNMS1048	HELPMAP	Correlates arguments from the help command to the primary help panel	DSIPARM
CNMS1084	HELPMAPU	Correlates arguments from the help command to the primary help panel for user-defined commands	DSIPARM
CNMS1097	TSTCSCF	Full-screen automation sample that captures event log and response time log for hardware group 1	CNMSAMP
CNMS1098	TSOSDA	Full-screen automation example of a TAF session	CNMSAMP
CNMS1101	same	Contains PIPE samples included in <i>Tivoli NetView for z/OS Customization: Using Pipes</i> .	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS1A01	DSIELFCB	This sample is an assembler language program that you can use for external logging. DSIELFCB returns information about the output file being used for logging. Use this sample with DSIELLR, DSIELMEM, and DSIELXIT.	CNMSAMP
CNMS1A02	DSIELLR	This sample is an assembler language program that you can use for external logging. DSIELLR contains a sample log off routine. Use this sample with DSIELFCB, DSIELMEM, and DSIELXIT.	CNMSAMP
CNMS1A03	DSIELXIT	This sample contains a sample installation exit that you can use to log data into a file. You can use this program whenever the system management facilities (SMF) are not being used.	CNMSAMP
CNMS1A04	DSISDMA	Message string definitions	CNMSAMP
CNMS2VSM	same	Creates sample database for the session monitor	CNMSAMP
CNMS3VSM	same	Creates sample database for the hardware monitor	CNMSAMP
CNMS4200	PTMPPLT	HLL PL/I template	CNMSAMP
CNMS4201	CTMPPLT	HLL C template	CNMSAMP
CNMS4202	ATMPCMDP	ATMPCMDP control section	CNMSAMP
CNMS4210	PEXIT3	HLL PL/I DSIEX03 example	CNMSAMP
CNMS4211	PSNDDAT	HLL PL/I send data example	CNMSAMP
CNMS4212	PWATDAT	HLL PL/I wait for data example	CNMSAMP
CNMS4213	PEXIT2A	HLL PL/I DSIEX02A example	CNMSAMP
CNMS4214	PCNMI	HLL PL/I CNMI example	CNMSAMP
CNMS4215	PKEYIO	HLL PL/I VSAM example	CNMSAMP
CNMS4216	PSCOPCK	HLL PL/I example of command authorization	CNMSAMP
CNMS4217	PFLVIEW	HLL PL/I view example	CNMSAMP
CNMS4218	PACTLU	HLL PL/I wait/trap example	CNMSAMP
CNMS4219	PSEQLOG	HLL PL/I sequential logging example	CNMSAMP
CNMS4220	PXITDI	DST initialization exit for USERVSAM sample DST	CNMSAMP
CNMS4221	PXITVN	DST empty VSAM data exit for sample USERVSAM DST	CNMSAMP
CNMS4222	PSNDDST	Sends VSAM requests to sample USERVSAM DST	CNMSAMP
CNMS4223	PDOVSAM	Processes VSAM requests under sample USERVSAM DST sent by OPRSAM command processor	CNMSAMP
CNMS4224	PPRIME	HLL PL/I DSIXITVN example	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS4226	PHSNDMU	PL/I sample that shows how to use CNMHSEND in a PL/I program	CNMSAMP
CNMS4227	same	PL/I sample that sends generic alert through program-to-program interface	CNMSAMP
CNMS4228	same	PL/I sample that sends data buffer through program-to-program interface	CNMSAMP
CNMS4229	same	PL/I sample that receives data buffer through program-to-program interface	CNMSAMP
CNMS4230	PRODMCON	Begins the HLL PL/I CNMQAPI member connection to RODM	CNMSAMP
CNMS4231	PAUTOTB	HLL PL/I example	CNMSAMP
CNMS4232	PREGISTR	PL/I sample for CNMRGS	CNMSAMP
CNMS4233	PSENDMU	PL/I sample for CNMSMU	CNMSAMP
CNMS4236	PHREGSTR	PL/I sample for CNMHRGS	CNMSAMP
CNMS4239	PPRSMDB	PL/I sample for CNMPMDB	CNMSAMP
CNMS4240	CEXIT3	HLL C DSIEX03 example	CNMSAMP
CNMS4241	CSNDDAT	HLL C send data example	CNMSAMP
CNMS4242	CWATDAT	HLL C wait for data example	CNMSAMP
CNMS4243	CEXIT2A	HLL C DSIEX02A example	CNMSAMP
CNMS4244	CCNMI	HLL C CNMI example	CNMSAMP
CNMS4245	CKEYIO	HLL C VSAM example	CNMSAMP
CNMS4246	CSCOPCK	HLL C example of command authorization	CNMSAMP
CNMS4247	CFLVIEW	HLL C view example	CNMSAMP
CNMS4248	CACTLU	HLL C wait/trap example	CNMSAMP
CNMS4249	CSEQLOG	HLL C sequential logging example	CNMSAMP
CNMS4250	CXITDI	DST initialization exit for USERVSAM sample DST	CNMSAMP
CNMS4251	CXITVN	DST empty VSAM data exit for sample USERVSAM DST	CNMSAMP
CNMS4252	CSNDDST	Sends VSAM requests to sample USERVSAM DST	CNMSAMP
CNMS4253	CDOVSAM	Processes VSAM requests under the USERVSAM DST sent by the CSNDDST command processor	CNMSAMP
CNMS4254	CPRIME	HLL C DSIXITVN sample	CNMSAMP
CNMS4256	CHSNDMU	C sample that shows how to use CNMHSEND in a C program	CNMSAMP
CNMS4257	same	HLL C sample that sends generic alert through program-to-program interface	CNMSAMP
CNMS4260	CRODMCON	Begins the HLL C CNMQAPI member connection to RODM	CNMSAMP
CNMS4261	CAUTOTB	HLL C sample	CNMSAMP
CNMS4262	CREGISTR	C sample for CNMRGS	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS4263	CSENDMU	C sample for CNMSMU	CNMSAMP
CNMS4266	CHREGSTR	C sample for CNMHRGS	CNMSAMP
CNMS4269	CPRSMDBR	C sample for CNMPMDB	CNMSAMP
CNMS4270	AXITVN	Provides initial record for empty VSAM database	CNMSAMP
CNMS4271	AMSGMOD	Issues DSIMDS to build a user-defined message module	CNMSAMP
CNMS4272	AWRTLOG	Shows how to write a message to the NetView log using DSIWLS	CNMSAMP
CNMS4273	AMLWTO	Demonstrates title line output using DSIPSS type=output	CNMSAMP
CNMS4274	ADATTIM	Gets current data and time and displays message	CNMSAMP
CNMS4275	ASEQLOG	Logs text to a sequential log	CNMSAMP
CNMS4276	ALISTMEM	Reads and displays a member from NetView DSIPARM data set	CNMSAMP
CNMS4277	AOPTTSK	User-defined optional sub-task	CNMSAMP
CNMS4278	ABLDMSG	Shows how DSIMBS can be used to build user-defined messages	CNMSAMP
CNMS4279	APSSFULL	Uses DSIPSS TYPE=ASYPANEL to display full screen panel	CNMSAMP
CNMS4280	ACALLCMD	Calls another command processor (or command list) directly	CNMSAMP
CNMS4281	DSIUSR00	Sample user-defined message member	CNMSAMP
CNMS4282	ATMPUXIT	ATMPUXIT control section	CNMSAMP
CNMS4283	DSIEX02A	Installation exit invoked for standard output to the operator's terminal	CNMSAMP
CNMS4284	ALERTMSG	Generates automation message for NMVT (Network Management Vector Transport) Alerts	CNMSAMP
CNMS4286	AHSNDMU	Assembler sample for DSIHSNDS	CNMSAMP
CNMS4287	CNMSGENA	Assembler sample that sends generic alert through program-to-program interface	CNMSAMP
CNMS4288	CNMSEND	Assembler sample that sends the data buffer through program-to-program interface	CNMSAMP
CNMS4289	CNMRECV	Assembler sample that receives the data buffer through program-to-program interface	CNMSAMP
CNMS4290	ARODMCON	Activates the assembler DSINOR that connects to RODM	CNMSAMP
CNMS4291	AAUTOTB	Tests automation table statements by sending a MSU directly to the automation table	CNMSAMP
CNMS4292	AREGISTR	Assembler sample for DSI6REGS	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS4293	ASENDMU	Assembler sample for DSI6SNDS	CNMSAMP
CNMS4294	AGETDS	Assembler sample for DSIGETDS	CNMSAMP
CNMS4295	OPERID	Example of an Automation Table Function (ATF)	CNMSAMP
CNMS4296	AHREGSTR	Assembler sample for DSIHREGS	CNMSAMP
CNMS4297	DSIEX17	Installation exit invoked for MVS messages and delete operator messages (DOM)	CNMSAMP
CNMS4298	DSIEX18	Installation exit that can be used as a template	CNMSAMP
CNMS4299	APRSMDB	Assembler sample which builds a message data block (MDB) and a source object and invokes the process MDB service	CNMSAMP
CNMS4305	PACTPIP	HLL PL/I PIPE command example	CNMSAMP
CNMS4307	DSIEX19	Installation exit that provides command authority checking for the RUNCMD command	CNMSAMP
CNMS4308	DSIEX20	Installation exit that allows filtering of session awareness (SAW) data	CNMSAMP
CNMS4402	CNMSNIFF	GMFHS automation example	CNMSAMP
CNMS4403	EKGSNIFF	GMFHS automation example	CNMSAMP
CNMS4405	CACTPIP	HLL C PIPE command example	CNMSAMP
CNMS4406	same	C sample that sends an INIT or DOWN alert to complete the DOMP010/DOMS010 protocols for service points that do not send an adequate alert themselves.	CNMSAMP
CNMS4501	same	Assembler IPCS installation exit for PPI trace records.	CNMSAMP
CNMS4508	TESTQRSR	Command procedure that returns a list of active spans and resources that can be controlled	CNMSAMP
CNMS4VSM	same	Creates sample database for the 4700 support facility	CNMSAMP
CNMS6201	MPFLSTAC	Conservative MVS MPF message suppression	CNMSAMP
CNMS6202	MPFLSTAA	Aggressive MVS MPF message suppression	CNMSAMP
CNMS6205	ACOTABLE	NetView automation table entries	CNMSAMP
CNMS6206	same	CMDMDL statements for system automation command lists	CNMSAMP
CNMS6207	same	JES2 and JES3 system log analysis program	CNMSAMP
CNMS6211	CLRLOG	Clears SYS1.LOGREC for future recording	CNMSAMP
CNMS6212	CLRSMF	Clears SYS1.MANX for future recording	CNMSAMP
CNMS6213	LGPRNT	Prints SYS1.LOGREC	CNMSAMP
CNMS6214	DSIPRT	Prints the primary and/or secondary NetView logs	CNMSAMP
CNMS6221	\$CLRSMF	Input to the CLRSMF procedure	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS6222	\$SOFT	Input to the LGPRNT procedure, step name SOFT	CNMSAMP
CNMS6223	\$SYSEXN	Input to the LGPRNT procedure, step name SYSEXN	CNMSAMP
CNMS6224	\$SYSUM	Input to the LGPRNT procedure, step name SYSUM	CNMSAMP
CNMS62J1	same	Rename JCL	CNMSAMP
CNMS62J2	RDSIPARM	SYSLOG analysis program	CNMSAMP
CNMS6401	same	CMDMDL statements for MVS commands	CNMSAMP
CNMS6402	same	CMDMDL statements for JES2 commands	CNMSAMP
CNMS6403	same	CMDMDL statements for JES3 commands	CNMSAMP
CNMS6404	same	CMDMDL statements for advanced starter set command lists	CNMSAMP
CNMS6405	DSITBL11	NetView automation table	CNMSAMP
CNMS6406	AOPUMCMT	TSO command list to copy command lists without comments	CNMSAMP
CNMS6408	same	Automated operator definitions	CNMSAMP
CNMS6409	DSIPROFM	AUTOMGR AUTOTASK profile	CNMSAMP
CNMS6410	DSIPROFG	Generic AUTOTASK profile	CNMSAMP
CNMS64P0	same	Panel to display automation status of all products	CNMSAMP
CNMS64P1	same	Panel to display automation information for a specific product	CNMSAMP
CNMS64P2	same	Panel to display message response variable values	CNMSAMP
CNMS64P3	same	Help panel for panel CNMS64P0	CNMSAMP
CNMS64P4	same	Help panel for panel CNMS64P1	CNMSAMP
CNMS64P5	same	Help panel for panel CNMS64P2	CNMSAMP
CNMS7030	same	Retrieves focal point definitions for primary and back-up names and stores in global variables	CNMSAMP
CNMS7101	same	Alias table definitions for default information problem records	CNMSAMP
CNMS7102	same	Alias table definitions for default information configuration records	CNMSAMP
CNMS7103	same	Alias table definitions for Network Configuration Application (NCA) configuration records	CNMSAMP
CNMS7104	same	Program interface data table (PIDT) definitions for NCA inquiry transactions	CNMSAMP
CNMS7105	same	PIDT definitions for retrieve transactions for NCA SNA records	CNMSAMP
CNMS7106	same	PIDT definitions for retrieve transactions for NCA circuit records	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMS7107	same	PIDT definitions for retrieve transactions for NCA equipment records	CNMSAMP
CNMS7108	same	PIDT definitions for retrieve transactions for NCA software records	CNMSAMP
CNMS7109	same	Alias table definitions supporting SNA topology manager problem records	CNMSAMP
CNMS7110	same	Alias table definitions supporting SNA topology manager configuration records	CNMSAMP
CNMS7111	same	Alias table definitions supporting SNA topology manager configuration records	CNMSAMP
CNMS7112	same	Alias table definitions supporting SNA topology manager configuration records	CNMSAMP
CNMS8002	RXUFUNC	Demonstrates how to return a value to the REXX caller	CNMSAMP
CNMS8003	AUTODROP	Conditionally drops pre-loaded NetView command lists	CNMSAMP
CNMS8013	DELVSAM	Sample REXX procedure that deletes a VSAM data set	CNMSAMP
CNMS8014	OPENVSAM	Sample REXX procedure that creates or opens a VSAM data set	CNMSAMP
CNMS8015	INITREC	Sample REXX procedure that writes the first record in a VSAM data set	CNMSAMP
CNMS8016	PUTREC	Sample REXX procedure (DSIVSMX) that writes records to VSAM data sets	CNMSAMP
CNMS8017	VSAMGLOB	Sample REXX procedure that uses VSAM for permanent task global	CNMSAMP
CNMS8018	BULKPUT	Sample REXX procedure (DSIVSMX) that writes multiple records to VSAM data sets	CNMSAMP
CNMS8019	STARVSAM	Sample REXX procedure that starts sample VSAM tasks	CNMSAMP
CNMS8020	DEFVSAMS	Sample REXX procedure (IDCAMS) that define VSAM data sets used by VSAM tasks.	CNMSAMP
CNMS8021	SAMPBULK	Sample REXX procedure (DSIVSAM) that writes multiple records to VSAM data sets	CNMSAMP
CNMS8022	same	NetView sample - SAMPVSAM task initialization member	CNMSAMP
CNMS8023	SENDTSO	Sends TSO commands via a TAF LU1 session and produces correlated output	CNMSAMP
CNMS8024	same	Processes TSO commands and returns output that can be processed by NetView pipelines	CNMSAMP
CNMS8027	CSCFIML	NetView sample to IML a 3174 controller	CNMSAMP
CNMS8029	same	Sample to interact with NetView through PPI to issue console commands	CNMSAMP
CNMSAF2	same	Sets RACF definitions for NetView operators and commands	CNMSAMP

|
|

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMSBAK1	same	Backup command authorization table	DSIPARM
CNMSCAT2	same	Sample command authorization table	DSIPARM
CNMSCNFT	same	This sample lets you define screen colors, prefix data, and prefix display order for message formatting.	DSIPARM
CNMSHMAT	same	Automation sample for hardware monitor instrumentation	DSIPARM
CNMSI101	same	Allocates VSAM database for network and trace logs	CNMSAMP
CNMSI201	same	Allocates VSAM database for session monitor	CNMSAMP
CNMSI301	same	Allocates VSAM database for hardware monitor	CNMSAMP
CNMSI401	same	Allocates VSAM database for 4700 support facility	CNMSAMP
CNMSI501	same	Allocates CSCF VSAM databases of sample network	CNMSAMP
CNMSI601	same	Allocates save/restore VSAM database	CNMSAMP
CNMSID01	same	Deletes VSAM databases	CNMSAMP
CNMSIHSA	same	Tivoli Enterprise Console automation table entries	DSIPARM
CNMSJ000	same	Changes samples to reference user-defined subarea and domain	CNMSAMP
CNMSJ001	same	Defines user ICF catalog and alias	CNMSAMP
CNMSJ002	same	Allocates partitioned data sets	CNMSAMP
CNMSJ003	same	Compresses and copies partitioned data sets	CNMSAMP
CNMSJ004	same	Allocates logs and databases	CNMSAMP
CNMSJ005	same	REPRO sample databases	CNMSAMP
CNMSJ006	same	Assembles and link-edits VTAM tables	CNMSAMP
CNMSJ007	CNMNDEF	Status monitor preprocessor procedure that starts the status monitor preprocessor and creates the DSINDEF member	PROCLIB
CNMSJ008	CNMNET	VTAM start procedure	PROCLIB
CNMSJ009	CNMPROC	NetView application address space start procedure	PROCLIB
CNMSJ010	CNMPSSI	NetView subsystem address space start procedure	PROCLIB
CNMSHTSP	same	Displays a list of Web addresses which can be selected to access websites	CNMSAMP
CNMSJ023	same	Reproduces VSAM file for verification	CNMSAMP
CNMSJ032	same	Creates /etc/netview and /tmp/netview directories and copies /usr/lpp/tcpip/samples mibs to /etc/netview/mibs	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMSJBUP	same	Creates the installation JCL samples library	CNMSAMP
CNMSJH10	CNMGMFHS	GMFHS start procedure	PROCLIB
CNMSJH12	same	Loads the RODM data cache for GMFHS	CNMSAMP
CNMSJH13	same	Relink-edits the assembler table, DUIFSMT	CNMSAMP
CNMSJI00	same	Copy statement	CNMSAMP
CNMSJI01	same	Copy statement	CNMSAMP
CNMSJI02	same	Copy statement	CNMSAMP
CNMSJI03	same	IEBCOPY control statements for VTAMLST	CNMSAMP
CNMSJI07	same	IEBCOPY sample procedures to PROCLIB	CNMSAMP
CNMSJI12	same	IEBCOPY GMFHS procedure to PROCLIB	CNMSAMP
CNMSJI18	same	IEBCOPY RODM procedures to PROCLIB	CNMSAMP
CNMSJI21	same	IEBCOPY control statements for VTAMLST	CNMSAMP
CNMSJI23	same	IEBCOPY event automation service procedure to PROCLIB	CNMSAMP
CNMSJI24	same	IEBCOPY status monitor preprocessor procedure to PROCLIB	CNMSAMP
CNMSJM01	same	VSAM LSR buffer definition	CNMSAMP
CNMSJM04	CNMPRT	Prints network or trace logs	PROCLIB
CNMSJM10	same	Generates a list of sense codes and frequency of occurrence	CNMSAMP
CNMSJM11	same	Assembles and link-edits the NetView REXX parameters module (DSIRXPRM)	CNMSAMP
CNMSJM12	same	Symbol substitution utility	CNMSAMP
CNMSJM13	same	DATA REXX initialization parameters	CNMSAMP
CNMSJN01	same	Japanese only: Defines Japanese message translations to the NetView command facility.	SDSIMSG1
CNMSJSQL	same	SQL plan installation sample job	CNMSAMP
CNMSJTSO	same	TSO command server sample job	CNMSAMP
CNMSJUNX	same	UNIX/390 command server sample job	CNMSAMP
CNMSJV03	same	Reorganizes NLDM CNM.SA01 database	CNMSAMP
CNMSJV04	same	Reorganizes NPDA CNM.SA01 database	CNMSAMP
CNMSPAN2	same	Sample NetView span table	DSIPARM
CNMSRPLY	same	Multiple reply support for a single MVS command	CNMSAMP
CNMSSTSO	same	Sample JCL to run the NetView TSO command server as an MVS started job	CNMSAMP
CNMSUNIX	same	Sample JCL to run the NetView UNIX command server as an MVS started job	CNMSAMP
CNMSTASK	same	NetView-provided task statements	DSIPARM
CNMSTDAT	same	Topology Display Instrumentation automation table sample	CNMSAMP

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
CNMSTGEN	same	You can include additional or modified CNMSTYLE definition statements, including DATA REXX logic.	DSIPARM
CNMSTNXT	same	Includes NetView-supplied CNMSTYLE updates.	DSIPARM
CNMSTPWD	same	Includes VSAM and ACB passwords.	DSIPARM
CNMSTSOS	same	MVS start command sample to start the NetView TSO command server as a started task	DSIPARM
CNMSTTWR	same	Includes style statements from non-NetView towers.	DSIPARM
CNMSTYLE	same	Defines some of the NetView initialization parameters.	DSIPARM
CNMSUNXS	same	MVS start command sample to start the NetView UNIX command server as a started task	DSIPARM
CNMSURLS	same	Contains a list of Web addresses read by CNMSHTSP	CNMSAMP
CNMSVTET	same	VTAM monitor auto-table: message suppression	DSIPARM
CNMSVTFT	same	VTAM monitor auto-table entries	DSIPARM
CNMSXENT	same	Sample used to show that NetView is installed	CNMSAMP
CNMSXSYS	same	Sample used to show that NetView System Services is installed	CNMSAMP
CNMTRMSG	same	NetView message translation sample	SDSIMSG1
CNMTRUSR	same	Message translation sample for user-defined message translations	SDSIMSG1
CNMTRXMP	same	Message translation sample for examples of message translations	SDSIMSG1
DSI6INIT	same	Prototype of the LU 6.2 transport initialization member	DSIPARM
DSI6SCF	same	Defines to the sphere of control manager which entry points to acquire into a focal point's sphere of control.	DSIPARM
DSIALATD	same	Includes initialization parameters for alias name translation. <i>Alias name translation</i> is a function that allows you to translate the names of logical units, logon modes, and classes of service. You will need this function only if you have naming conflicts between interconnected networks.	DSIPARM
DSIALTAB	same	Includes alias translation definitions. You can use this sample to define new names for a logical unit (LU), a class of service (COS), or a logon mode (MODE).	DSIPARM
DSIAMMAT	same	Automation table for Application Management Instrumentation	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
DSIAMIE	same	Application Management Instrumentation for NetView on a host with an event automation service message adapter	DSIPARM
DSIAMII	same	Application Management Instrumentation autotask initialization and termination configuration file	DSIPARM
DSIAMIN	same	Application Management Instrumentation for NetView on a host not running the event automation service	DSIPARM
DSIAMIR	same	Application Management Instrumentation for NetView on a host without an event automation service message adapter	DSIPARM
DSIAMIT	same	Application Management Instrumentation for NetView on a host with an event automation service message adapter	DSIPARM
DSIAMLTD	same	Includes initialization parameters for the access method LU function	DSIPARM
DSIAPMI	same	APM autotask configuration file	DSIPARM
DSIAPML	same	APM instrumentation	DSIPARM
DSIAPMR	same	APM instrumentation	DSIPARM
DSIAUTB	same	Part list for usage of the AUTOBYPAS REXX or CLIST function	DSIPARM
DSIAUTBU	same	User defined part list for AUTOBYPAS REXX or CLIST function	DSIPARM
DSIBKMEM	same	Defines the initialization values for the task DSIAL2WS. The DSIAL2WS task sends alerts to workstations running AIX® SNA Manager/6000.	DSIPARM
DSICCDEF	same	Defines command termination conditions and is used with the CCDEF command.	DSIPARM
DSICCSYS	same	Defines command termination conditions and is used with the CCDEF command for NetView System Services.	DSIPARM
DSICMD ¹	same	NetView command model statements for all NetView commands.	DSIPARM
DSICMD1A	same	This non-executable sample identifies command model statements added between NetView V3R1 and TME® 10 NetView for OS/390 V1R1. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
DSICMD1B	same	This non-executable sample identifies command model statements added between TME 10 NetView for OS/390 V1R1 and V1R2. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD1C	same	This non-executable sample identifies command model statements added between TME 10 NetView for OS/390 V1R2 and Tivoli NetView for OS/390 V1R3. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD1D	same	This non-executable sample identifies command model statements added between TME 10 NetView for OS/390 V1R3 and Tivoli NetView for OS/390 V1R4. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD21	same	This non-executable sample identifies command model statements added to the NetView program between NetView V1R3 and V2R1. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD22	same	This non-executable sample identifies command model statements added between NetView V2R1 and V2R2. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD23	same	This non-executable sample identifies command model statements added between NetView V2R2 and V2R3. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD24	same	This non-executable sample identifies command model statements added since NetView V2R3. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD2G	same	This non-executable sample identifies command model statements added for the GMFHS feature.	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
DSICMD31	same	This non-executable sample identifies command model statements added between NetView V2R4 and V3R1. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMD51	same	This non-executable sample identifies command model statements added between NetView V1R4 and V5R1. Migrators to V5R1 can include this member in their existing DSICMD instead of manually adding the new command model statements.	DSIPARM
DSICMDU	same	Command model statements for user-defined commands	DSIPARM
DSICMENT	same	NetView command model statements.	DSIPARM
DSICMSYS	same	NetView command model statements for NetView System Services functions.	DSIPARM
DSICNM	same	Contains initialization parameters for the status monitor.	DSIPARM
DSICPINT	same	Contains definitions for the network product support communications network management interface (CNMI) function.	DSIPARM
DSICRTTD	same	CNM router initialization member	DSIPARM
DSIDB2DF	same	Sample initialization member for task DSIDB2MT	DSIPARM
DSIDMN	same	Provided for migration purposes. It includes DSIDMNU.	DSIPARM
DSIELMEM	same	Includes initialization parameters for task DSIELTSK	DSIPARM
DSIEX21	same	Assembler exit used with DSITCPRF for encryption	CNMSAMP
DSIGEMTR	same	Tivoli Ready for z/OS configuration sample	DSIPARM
DSIHINIT	same	Prototype of the LU 6.2 high performance definition statements	DSIPARM
DSIILGCF	same	Syslog task configuration	DSIPARM
DSIKINIT	same	Initializes values for the command facility's DSIKREM task	DSIPARM
DSILOGBK	same	Includes initialization parameters for NetView disk log support	DSIPARM
DSILUCTD	same	Contains initialization parameters for the CNM data-transfer task for subarea 01 only (A01MPU in the sample network).	DSIPARM
DSINDEF	same	Contains the status monitor network node definition	DSIPARM
DSIOPF	same	Operator definitions and passwords	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
DSIOPFU	same	Operator definitions and passwords for user-defined operators.	DSIPARM
DSIPRFGR	same	Initializes the generic receiver task DSINVGR	DSIPRF
DSIPRFLC	same	Provides the DSILCOPR profile statements for an operator ID to run as an unattended operator.	DSIPRF
DSIPROFA	same	Contains a profile for a network operator. An operator using this profile is not an authorized receiver and can enter class-2 commands only.	DSIPRF
DSIPROFB	same	Contains a profile for a network operator. An operator using this profile is an authorized receiver and can enter both class-1 and class-2 commands.	DSIPRF
DSIPROFC	same	Contains a profile for an automated operator (that is, an operator started with the AUTOTASK command to run as an unattended operator).	DSIPRF
DSIPROFD	same	Contains a profile for an automated operator (that is, an operator started with the AUTOTASK command to run as an unattended operator). This profile is used by unattended operators who want to run OVERRIDE CMD=LOW.	DSIPRF
DSIPROFE	same	Defines an automated operator profile for the NetView Bridge RTRINIT command This is a sample profile for an automated operator (that is, an operator started with the AUTOTASK command to run as an unattended operator). This profile is used by the BRIGOPER automated operator.	DSIPRF
DSIPROFF	same	Defines automated operator profile for the NetView Bridge REMOTEBR command. This is a sample profile for an automated operator (that is, an operator started with the AUTOTASK command to run as an unattended operator). This profile is used by the REMOPER automated operator.	DSIPRF
DSIPROFG	same	Automated operator profile that is functionally equivalent to DSIPROFD. It is provided for compatibility reasons.	DSIPRF
DSIPROFI	same	Automated operator profile for the event management autotask	DSIPRF
DSIPROFJ	same	Automated operator profile for the GMFHS common operations services (COS) gateway autotask	DSIPRF
DSIPROFK	same	Automated operator profile for instrumentation autotask	DSIPRF

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
DSIPROFV	same	Automated operator profile for the visual BLDVIEWS server	DSIPRF
DSIQTSKI ²	same	Defines RODMs and NetView tasks to the RODM access and control subtask (DSIQTSK).	DSIPARM
DSIREXCF	same	Rexec server task configuration	DSIPARM
DSIRHOST	same	RSH security file	DSIPARM
DSIROVSI	same	Defines the initialization values for the task DSIROVS	DSIPARM
DSIRSHCF	same	RSH server task configuration	DSIPARM
DSIRTTTD	same	TCP/IP alert receiver	DSIPRF
DSISCHED	same	CHRON command calendar schedule file	DSIPARM
DSISECUR	same	RMTCMD security table	DSIPARM
DSISVRTD	same	Initialization values for the command facility's save/restore (DSISVRT) task	DSIPARM
DSITBL01	same	Contains automation definitions for subarea 01.	DSIPARM
DSITPCPF	same	Defines the initialization values for DSITCPIP.	DSIPARM
DSITCPRF	same	Defines the operators that can log on to NetView using the NetView 3270 management console.	DSIPRF
DSITDSR	same	Application Management Instrumentation for NetView on a host not running the event automation service	DSIPARM
DSITRCBK	same	Includes NetView trace log support initialization parameters	DSIPARM
DSITSK	same	Defines a list of tasks that are allowed for NetView System Services	DSIPARM
DSIUINIT	same	Defines the initialization values for DSIUDST. Values for DSIUINIT are specified in CNMSTYLE and DSIUINIT should not be modified.	DSIPARM
DSIVPARM	same	Initialization parameters for VPD task supporting Network Asset Management	DSIPARM
DSIWBMEM	same	Initialization parameters for the NetView Web Server	DSIPARM
DSIW3PRF	same	Properties definitions for 3270 web sessions	DSIPARM
FKXSCM	same	Defines community names for IP resources to AON/TCP for proactive monitoring and commands	DSIPARM
FKXSDVPT	same	Used by DVIPA processing to prime the DVIPA status and definition table	DSIPARM
FKXVHTML	same	SNMPView HTML source code	DSIPARM
FKXWHTML	same	WEB IP trace HTML source code	DSIPARM

Table 19. List of Samples (continued)

Distributed As	Name	Description	Data Set Name
-------------------	------	-------------	------------------

Notes:

1. **More Information about DSICMD:** The CMDMDL statements provide the definitions for the commands. Some of the CMDMDL statements have command synonym (CMDSYN) statements. These statements provide a synonym for the command.
 This definition also provides the cross-domain log on definitions and the CMDMDL and CMDSYN statements for the terminal access facility (TAF) and the VTAM program.
 Files whose names begin with CNMS6 are included in NETVIEW.V5R1M0.CNMSAMP. Load these files into DSICMD to assist you in using the automation command lists that are also included on the distribution tape.
2. **More Information about DSIQTSKI:** Refer to *Tivoli NetView for z/OS Automation Guide* for an example of using the DSIQTSK task to manage your RODMs. The example illustrates using RODM to automate the recovery of a failed resource. For each RODM you specify in DSIQTSKI, you can define an initialization command processor to be invoked when the DSIQTSK subtask successfully connects to RODM. You can specify the name of this command processor within each RODM definition statement.
 DSIQTSKI also contains information about which RODMs you need to connect to.

Appendix D. Partitioned Data Sets and Members

This appendix lists some of the ddnames in the NetView start procedure CNMPROC and some of the data set members which NetView uses. NetView will look in the data sets specified on the given ddname's DD statement for the members listed in Table 20.

Table 20. List of ddnames and Data Set Members

ddname	Member	Description
BNJPNL1	Various	Source panel definitions (hardware monitor)
BNJPNL2	Various	Color maps (hardware monitor)
CNMPNL1	Various	Source panel definitions (NetView)
DSIARPT	Various	Stores the output reports produced from running tests of the automation table. There is a summary report and a detail report.
DSIASRC	Various	Testing automation tables
DSICLD	Various	Command list definitions
DSILIST	Various	Automation table listings
DSIMSG	Various	Disk-resident messages
DSIOPEN	CNMKEYS	Key settings for components
	CNMKEY2	Plus any non-protected members
	<i>User-defined</i>	
DSIPARM	AAUCNMTD	Session monitor CNMI details
	AAUKEEP1	Session monitor data wrap details
	AAUPRMLP	Session monitor initialization
	AAURTM1	Session monitor RTM classes
	BNJ36DST	4700 support facility initialization
	CNMSTYLE	Initialization parameters
	DSIALATD	ALIAS initialization
	DSIALTAB	ALIAS name translations
	DSIAMLTD	Session monitor cross-domain details
	DSICMD	Commands, command processors, command list names
	DSICNM	Status monitor initialization
	DSICPINT	NPS initialization
	DSICRTTD	CNM router task
	DSIELMEM	External log initialization (if no SMF)
	DSILOGBK	Network log initialization
	DSILUCTD	Session monitor data transfer task details
	DSINDEF	Status monitor network data base
	DSIOPF	Operator definitions
	DSISVRTD	Save/restore initialization

Table 20. List of ddnames and Data Set Members (continued)

ddname	Member	Description
	DSITBL01	Sample automation table
	DSITRCBK	Network trace initialization
	DSIVPARM	NAM initialization
	CNMBxxx	Sense code descriptors
	<i>User definitions</i>	User defined members
DSIPRF	DSIPROFA	Sample operator profile
	DSIPROFB	Sample operator profile
	DSIPROFC	AUTOTASK profile (basic)
	DSIPROFD	AUTOTASK profile (MVS console)
	DSIPRFLC	Unattended operator profile
DSIVTAM	VTAM details	For span of control

Index

Special Characters

%INCLUDE statement 46
&NV2I 39
&Q1 3
&SQ1 3
&SYSCONE 15, 39
&SYSNAME 13
&VQ1 4

Numerics

4700 support facility
 enabling 45
 initialization 123
 samples 81
 VSAM cluster 21

A

A01APPLS 25
 STATOPT statements 27
A01SNA and local terminal support 32
AAUCNMTD 103, 123
AAUKEEP1 103, 123
AAUPRMLP 103, 123
AAURTM1 103, 123
AAUTCNMI task 33, 77
AAUTSKLP task 77
AAUVSPL cluster 21
ACBpassword, changing 26
accessibility information ix
address space 7
address spaces, increasing 12
alerts panel 69
ALERTSH command 69
ALIAS
 initialization 123
 name translations 123
alias, defining
 names 28
 translations 29
ALIAS name for NetView data sets 17
ALIASAPL task 33, 77
ALIASMEM statement 28
allocating
 data sets 18
 VSAM clusters 20
AMODETAB (CNMS0001)
 assembling VTAM tables 31
 logmode table 27
AON
 enabling 44
 panel data set 18
 samples 81
AON/SNA
 samples 81
AON/TCP
 samples 81
APF authorization 9, 10
 dynamic changes 9, 11

APPL statements 25, 27
application job, NetView 43
Application Management Instrumentation
 enabling 45
ARM couple data set 13
ARM parameter
 CNMPSSI start procedure
 specifying NetView interface 35
assembler samples 81
assembling VTAM tables 31
ATCCON01 25
ATCSTR00 32
AUSSTAB (CNMS0011) table
 new installation 31
 updating 31
AUTH statement
 PPO value on CNM01PPT
 statement 26
AutoBridge
 enabling 45
automatic restart manager support 13
automatically run commands and
 command lists 47
automation table
 reports 18
 sample 124
 test listings 123
 testing 18, 123
AUTOTASK
 profile 124
auxInitCmd keyword
 CNMSTYLE 47

B

BLKSIZE operand and defining command
 lists 41
blocksize, allocating 41
BNJ36DST 103, 123
BNJ36PR cluster 21
BNJDSE36 task 77
BNJDSERV task 33, 77
BNJLGPR cluster 21
BNJMBDST 103
BNJMNPDA task 77
BNJPNL1 123
BNJPNL2 123
books
 feedback vii
 online vii
 ordering vii
buffer size, modifying 40

C

C&NV2I.STGEN 46
catalog, defining user 17
CBUF symbolic parameter in
 CNMPSSI 35

CDRM
 identifying 92
 major node, network name for 29
CDRSC 29, 91
central site control facility (CSCF)
 VSAM cluster 21
changing
 domain and subarea numbers 18
channel-attached
 terminals 32
checkpoint data sets, RODM 21
class of service (COS) names, aliases
 for 30
class-of-service table 101
cloning, z/OS function 15
CNM router task 123
CNM routing table 100
CNM01BRW 33
CNM01LUC 33
CNM01VMT 33
CNMCALRT task 77
CNMCAU00 105
CNMCSRVP in LPALST 8
CNMCSSIR
 task 77
CNMETSO 105
CNMEVTAM command list 33
CNMGMFHS 20
CNMGNETV 105
CNMGSPCR 105
CNMGTVL 105
CNMHELPH 105
CNMI 123
CNMJSCL 105
CNMJSHW 105
CNMKEYS 105, 123
CNMKEYS2 105
CNMMIGKA 105
CNMMIGKV 105
CNMMSENU 105
CNMNDEF 20
CNMNET 20, 33
 starting 51
CNMNEWS 57, 105
CNMPNL1 18, 123
CNMPROC 20, 39
 start procedures 35
 starting 53
CNMPRT 20
CNMPSSI 20
 start procedures 35
 starting the subsystem address
 space 52
CNMS0001
 logmode table 27
 sample 105
CNMS0003 25, 105
CNMS0010 32, 105
CNMS0013 26, 105
 modifying major nodes 25
 STATOPT statements 27

CNMS0055 (DSICTMOD) 106
 CNMS0073 (A01SNA) 32, 106
 CNMSAF2 113
 CNMSAMP library 5, 19
 CNMSBAK1 114
 CNMSCAT2 114
 CNMSCNFT 114
 CNMSHMAT 114
 CNMSHTSP 114
 CNMSI101 21, 114
 CNMSI201 21, 114
 CNMSI301 21, 114
 CNMSI401 21, 114
 CNMSI501 21, 114
 CNMSI601 21, 114
 CNMSID01 21, 114
 CNMSJ000 18, 114
 CNMSJ001 17, 114
 CNMSJ002 18, 114
 CNMSJ003 114
 new installation 19
 CNMSJ004
 allocating VSAM clusters 20
 sample 114
 CNMSJ005 23, 114
 CNMSJ006 31, 114
 CNMSJ007 114
 CNMSJ008 10, 33, 114
 starting VTAM 51
 CNMSJ009 10, 39, 114
 starting NetView 35, 53
 CNMSJ010 10, 114
 parameters 35
 starting subsystem address space 52
 CNMSJ023 114
 CNMSJ032 114
 CNMSJBUP
 job 17
 sample 115
 CNMSJH10 115
 CNMSJH12 115
 CNMSJH13 115
 CNMSJI00 115
 CNMSJI01 115
 CNMSJI02 115
 CNMSJI03 20, 115
 CNMSJI07 20, 115
 CNMSJI12 20, 115
 CNMSJI18 20, 115
 CNMSJI21 20, 115
 CNMSJI23 20, 115
 CNMSJI24 20, 115
 CNMSJM01 115
 CNMSJM04 115
 CNMSJM10 115
 CNMSJM11 115
 CNMSJM12 13, 115
 CNMSJM13 115
 CNMSJSQL 115
 CNMSJTSO 115
 CNMSJUNX 115
 CNMSPAN2 115
 CNMSRPLY 115
 CNMSSTSO 115
 CNMSSUNX 115
 CNMSTASK 46, 115
 CNMSTDAT 115

CNMSTGEN 116
 CNMSTNXT 116
 CNMSTPWD 21, 26, 46, 116
 ACB password 46
 CNMSTSOS 116
 CNMSTTWR 46, 116
 CNMSTYLE 116, 123
 %INCLUDE 46
 &CNMRODM 46
 &NV2I 39, 44
 ACBpassword, changing 26
 auxInitCmd keyword 47
 components, enabling 44, 46, 48
 DOMAINID, changing 26
 naming convention 39, 44
 NetView domain 46
 overview 44
 RODM name 46
 security 47
 tower statement 44
 CNMSUNXS 116
 CNMSURLS 116
 CNMSVTET 116
 CNMSVTFT 116
 CNMTAMEL task 77
 CNMTRMSG 116
 CNMTRUSR 116
 CNMTRXMP 116
 color maps 123
 command
 security 47
 command facility
 list of commands 59
 PF keys 61
 samples 81
 testing 55
 command list
 automatically run 47
 defining 40
 definitions 123
 commands 123
 concatenation, definition 41
 CONSNAME command 15
 console ID 15
 CONSOLxx member 14
 COS (class of service) statement 30
 COUPLExx member 13
 CPDLB2 volume 22
 CSCFIML 113
 CUADDR parameter 32
 Customer Support viii

D

data cache
 load procedure 20
 DATA REXX logic 46
 data set naming conventions 3
 defining
 ALIAS translations 29
 NetView
 domain 46
 VSAM clusters 20
 definition data set members in
 DSIPARM 41
 disability information ix
 dispatch priority, setting 40

distribution tape 4
 DLOGMOD operand 27
 domain
 changing 18
 defining 46
 DOMAINID, changing 26
 DPRTY parameter 40
 DSI6DST task 78
 DSI6INIT 116
 DSI6SCF 116
 DSIALATD 116, 123
 DSIALTAB 28, 29, 116, 123
 DSIAMIAT 116
 DSIAMIE 117
 DSIAMII 117
 DSIAMIN 117
 DSIAMIR 117
 DSIAMIT 117
 DSIAMLTD 117, 123
 DSIAMLUT task 33, 77
 DSIAPMI 117
 DSIAPML 117
 DSIAPMR 117
 DSIARPT 18, 123
 DSIASRC 18, 123
 DSIATOPT task 77
 DSIAUTB 117
 DSIAUTBU 117
 DSIBKMEM 117
 DSICCDEF 117
 DSICCSYS 117
 DSICLD 40, 123
 DSICMD 117, 123
 DSICMD1A 117
 DSICMD1B 118
 DSICMD1C 118
 DSICMD1D 118
 DSICMD21 118
 DSICMD22 118
 DSICMD23 118
 DSICMD24 118
 DSICMD2G 118
 DSICMD31 119
 DSICMD51 119
 DSICMDU 119
 DSICMENT 119
 DSICMSYS 119
 DSICNM 119, 123
 DSICPINT 119, 123
 DSICRTR task 33, 77
 DSICRTTD 119, 123
 DSIDB2DF 119
 DSIDB2MT task 77
 DSIDCBMT task 79
 DSIDMN 119
 DSIELMEM 119, 123
 DSIELTSK task 77
 DSIEX21 119
 DSIG symbolic parameter in
 CNMPSSI 36
 DSIGDS
 starting VTAM ACBs and NetView
 tasks 33
 task 33, 77
 DSIGEMTR 119
 DSIHINIT 119
 DSIHLLMT task 79

DSIHPDST task 78
 DSILGCF 119
 DSIKINIT 119
 DSIKPNL cluster 21
 DSIKREM task 78
 DSILIST 123
 DSILIST data set 18
 DSILOG task 78
 DSILOGBK 119, 123
 DSILOGMT task 79
 DSILOGP cluster 21
 DSILUCTD 119, 123
 DSIMONIT task 79
 DSIMSG 123
 DSINDEF 119, 123
 DSIOPEN 123
 DSIOPF 62, 119, 123
 DSIOPFU 120
 DSIPARM data set 18
 DSIPHONE
 LNKLSTxx updates 10
 DSIPRF
 data set 18
 members 124
 DSIPRFGR 120
 DSIPRFLC 120, 124
 DSIPROFA 120, 124
 DSIPROFB 120, 124
 DSIPROFC 120, 124
 DSIPROFD 120, 124
 DSIPROFE 120
 DSIPROFF 120
 DSIPROFG 120
 DSIPROFI 120
 DSIPROFJ 120
 DSIPROFK 120
 DSIPROFV 121
 DSIQTSK task 78
 DSIQTSKI 121
 DSIREXCF 121
 DSIRHOST 121
 DSIROVS task 78
 DSIROVSI 121
 DSIRQJOB task 78
 DSIRSHCF 121
 DSIRTTT task 78
 DSIRTTD 121
 DSISCHED 121
 DSISECUR 121
 DSISTMMT task 79
 DSISVRT
 cluster 21
 task 78
 DSISVRTD 121, 123
 DSITBL01 121, 124
 DSITCPCF 121
 DSITCPIP task 78
 DSITCPRF 121
 DSITDSR 121
 DSITIMMT task 79
 DSITRACE task 78
 DSITRCBK 121, 124
 DSITRCP cluster 21
 DSITSK 121
 DSIUDST task 78
 DSIUINIT 121
 DSIVPARM 121, 124

DSIVTAM 124
 DSIW3PRF 121
 DSIWBMEM 121
 DSIWBTSK task 78
 DSIWEB task 79
 DUIDGHB task 78
 DUIFSSCO task 78
 duplicate resource names, resolving 28

E

E/AS
 start procedure 20
 e-mail contact viii
 EKGLOADP 20
 EKGXRODM 20
 EMCS console
 assigning names 15
 estimates, decisions before calculating
 full-screen automation 5
 RODM checkpoint data sets 5
 session awareness record filtering 5
 extended multiple console support 15
 external log
 initialization (no SMF) 123
 EZLTCFG task 78
 EZLTDDF task 78
 EZLTLOG task 78
 EZLTSTS task 78

F

feedback about publications viii
 FKXSCM 121
 FKXSDVPT 121
 FKXVHTML 121
 FKXWHTML 121
 FLBTOPO task 79
 full-screen sessions 28

G

GETCONID command 15
 GMFHS
 installation samples 81
 samples 81
 SCHEDxx updates 7
 start procedure 20
 graphics
 enabling 45

H

hardware monitor
 color maps 123
 enabling 45
 Information/Management 42
 initialization 48
 panel definitions 123
 samples 81
 SMF log records 14
 testing 68
 VSAM cluster 21
 help facility
 testing 58

HLL samples 81

I

ICF catalog 17
 IEAAPF 8
 IEAAPFxx member 10
 IEASYM statement 13
 IEASYMxx member 13
 IEASYsxx member 12
 IEC161I messages from VSAM 53
 IEFSSNxx member 11
 IEFUSI exit 8
 IHSAEVNT 20
 index, naming conventions 3
 information, accessibility ix
 information, disability ix
 Information/Management 42
 initial command
 automatically run 47
 installation
 results 5
 task definition 5
 verification 23
 installing
 NetView 17
 ISTMGCO0 table 31
 ISTMGCO1 table 31
 ISTSDCOS class of service table 31

J

JCL symbolics in naming conventions 3
 JES 49

K

keyboard, shortcut keys ix

L

link-list, defining 10
 Link Pack Area (LPA)
 building pageable 8
 creating 8
 LIST command 60
 LNKLSTxx member, updating 10
 LNKLSTxx statements 9, 10
 loading partitioned data sets with
 samples
 new installation 19
 loading verification data sets 23
 LOADxx member 13
 log
 printing automatically 42
 logmode entries 55
 LOGMODE entry, using 55
 logmode table
 changing 27
 logon mode table 101
 LPALSTxx statements 8
 LU
 list 71
 LU 6.2 communication 25
 LU name translation statement 29

M

- manuals
 - feedback vii
 - online vii
 - ordering vii
- MBUF symbolic parameter in CNMPSSI 36
- messages
 - disk-resident 123
 - IEC161I from VSAM 53
- MODE statement 30
- mode table changes, reviewing for 31
- MSGIFAC parameter
 - CNMPSSI start procedure specifying NetView interface 36
- MSM
 - samples 82
- multiple concurrent NetView operators 27
- MultiSystem Manager
 - enabling 45
- MVS
 - ASID value, changing 12
 - CONSOLxx 14
 - COUPLExx 13
 - extended multiple console support 14
 - IDCAMS statement members 81
 - IEAAPFxx 10
 - IEBCOPY procedures 81
 - IEFSSNxx 11
 - IEFSYMxx 13
 - IEFSYSxx 12
 - LNKLSTxx 10
 - LOADxx 13
 - LPALSTxx 8
 - message traffic 14
 - preparation 7
 - PROGxx 9
 - restarting 16
 - RSVNONR value, changing 12
 - samples 81
 - SCHEDxx 7
 - secondary subsystems, defining 11
 - SMFPRMxx 14
 - subsystem allocatable consoles 14
- MVS command management
 - enabling 45
- MVS consoles
 - determining type used 14
- MVS exit
 - LNKLSTxx updates 10
- MVS MODIFY command 14
- MVS ROUTE command 14
- MVS sockets application 43

N

- NAM initialization 124
- name of resource 27
- naming conventions, data set 3
- NCP
 - coding 82
 - definition 83
 - samples 82
- NCP definitions 81

- NCP definitions (*continued*)
 - reviewing for mode table changes 31
- NETID start option, VTAM 29
- NETLOGA 66
- NETVAPPL 19
- NetView
 - address spaces 7
 - address spaces, increasing 12
 - alias name 17
 - application job 43
 - buffer size, modifying 40
 - CNMSTYLE 44
 - command lists, defining 40
 - command prefix designator 14
 - data set types 3
 - data sets, defining 41
 - defining domain 46
 - defining to VTAM 25
 - extended multiple console support
 - consoles 14
 - high-level qualifier 3
 - identifier 46
 - initialization 44
 - libraries, authorization 9, 10
 - logs, printing 42
 - naming conventions 3
 - organization 1
 - PDS members 123
 - production 73
 - program directory 4
 - SCHEDxx updates 7
 - sense codes 41
 - session information 60
 - setting dispatch priority 40
 - start procedure 20
 - start procedure, reviewing 35
 - starting 51, 53
 - starting before JES 49
 - starting the subsystem address space 52
 - startup procedure 39
 - subsystem allocatable consoles 14
 - subsystem interface start procedure 20
 - symbolics, setting 46
 - system symbolics, setting 13
 - tasks 33, 77, 78
 - tasks for multiple operators 27
 - TCP/IP, defining 43
- NetView subsystem interface
 - start procedure 20
- NETVIEW.V5R1M0.CNMSAMP library
 - new installation 19
- NETVIEW.V5R1USER.CNM01
 - .CNMPNL1 18
 - .DSIARPT 18
 - .DSIASRC 18
 - .DSILIST 18
 - .DSIPARM 18
 - .DSIPRF 18
 - .SDSIOPEN 18
- NETVIEW.V5R1USER.VTAMLIB 18
- network log 66
 - print procedure 20
- network log initialization 123
- network name 29
- network organization 1

- NETWORK statement 29
- network trace initialization 124
- NLDM command 70
- NMC
 - VTAM updates 25
- NPDA command 68
- NPS initialization 123
- NTRI 95
- NV2I, local system symbolic 39

O

- online publications viii
- operator
 - control sessions 28
 - multiple concurrent 27
 - security 47
- operator definitions 123
- operator profile
 - CONSNAM attribute 15
 - sample 124
- ordering publications viii
- organization, NetView and network 1
- ORIGNET statement 29

P

- P256BUF operand 39
- P4000BUF operand 39
- panels
 - source 123
- partitioned data sets
 - allocating 18
 - loading
 - new installation 19
- password
 - VSAM 21
- PATH statement 96
- PDS members 123
- PF key settings 61
- PFXREG operand 38
- POI task, defining VTAM 26
- PPIOPT operand 38
- PPO value on AUTH operand 26
- PPOLOG statement 32
- preinstallation tasks
 - applying required PTFs and APARs 4
 - estimating resources 5
 - getting the PSP Bucket 4
 - program directory 4
 - preventive service planning 4
 - primary POI task 26
- printing logs
 - automatically 42
- priority, setting dispatch 40
- program directory 4
- program properties table 7
- PROGxx member 9
- PSP 4
- PTF 4
- publications
 - feedback vii
 - online vii
 - ordering vii

R

- region size
 - specifying 40
- RELATE value to define user catalog 17
- resource
 - defining 27
 - name, defining to status monitor 27
 - names, using alias names to resolve
 - duplicate 28
- resource estimates 5
- REXX samples 81
- RODM
 - allocating 21
 - checkpoint data set, allocating 21
 - deleting 22
 - IEFUSI exit 8
 - LNKLSTxx updates 10
 - load data cache procedure 20
 - log, allocating 21
 - samples 81
 - SCHEDxx updates 7
 - start procedure 20
- root name 26
- RU sizes
 - logmode table 55

S

- SAA DM members 81
- sample network 1
- samples
 - domain 18
 - NCP 82
 - network 1
 - subarea 18
- save/restore function
 - VSAM cluster 21
- save/restore initialization 123
- SCHEDxx statements 7
- SCNMLPA1 library 8
- SDSIOPEN 18
- secondary subsystem, defining 11
- SECOPTS statement 47
- security 47
- security migration tool 18
- sense code
 - CNMBxxx 124
- samples 81
- specifying 41
- session
 - full-screen 28
 - operator-control 28
- session information 60
- session monitor
 - CNMI details 123
 - cross-domain details 123
 - data transfer task details 123
 - data wrap details 123
 - enabling 45
 - initialization 123
 - network data base 123
 - RTM classes 123
 - samples 81
 - SMF log records 14
 - testing 70
 - VSAM cluster 21

- SETCONID command 15
- SETPROG command 9, 11
- setting dispatch priority 40
- SEZALINK 43
- SEZALNK2 43
- SEZALNKG 43
- shortcut keys, keyboard ix
- slot size, modifying 40
- SMF records 14
- SMFPRM 14
- SMFPRMxx 14
- SMP/E format 4
- SMP target library name 3
- SNA terminal
 - channel-attached 32
- SNA topology manager
 - APPL name 26
 - VTAM updates 26
- span of control
 - security 47
- SPO value on the AUTH operand 26
- SSCP table 92
- start procedure 20
- starting
 - NetView 53
 - NetView subsystem address space 52
 - option, VTAM NETID 29
 - procedure, reviewing NetView 35
 - VTAM 51
- STATOPT statement 27
- status monitor
 - defining resources to 27
 - preprocessor procedure 20
- STEPLIB 9, 10
- storage
 - key 7
- storage estimates 5
- subarea, changing 18
- subsystem allocatable consoles 14
- subsystem application procedure,
 - modifying 35
- subsystem designator character 14
- SWITCH command 67
- symbolic naming conventions, JCL 3
- symbolics
 - CNMSTYLE 46
- SYS1.LPALIB 8
- SYS1.PARMLIB 10
- SYSDEF statement 13
- sysplex environment 14
- System Automation for OS/390 44
- system symbolic
 - &SYSCLONE 15
- system symbolics
 - CNMSJM12 13
 - IEASYMxx 13
 - IEFSYMxx, updating 13
 - NetView procedure 39
 - user-defined 13

T

- tables, assembling VTAM 31
- TAF
 - sessions 63
- TAF (terminal access facility)
 - user tasks 28

- target library name, System Modification Program 3
- tasks 77
 - AAUTCNMI 33
 - ALIASAPL 33
 - BNJDSESV 33
 - CNM01BRW 33
 - CNM01LUC 33
 - CNM01VMT 33
 - DSIAMLUT 33
 - DSICRTR 33
 - DSIGDS 33
 - NetView supplied 78
 - restarting VTAM, considerations 33
 - starting and stopping 78
 - statements 46
 - VPDTASK 33
- TCP/IP
 - configuration data 43
 - defining 43
- Tivoli Customer Support viii
- TOWER statement 44, 46
- trace log 21
 - print procedure 20

U

- unattended operator profile 124
- unique identifier, assigning NetView 46
- user catalog, defining 17
- USS table 100
- USS table changes, reviewing 31

V

- verifying
 - installation 55
 - of NetView 55
- VPDTASK task 33, 78
- VSAM
 - allocating 20
 - data sets, naming conventions 3
 - deleting 22
 - password, defining 21
 - passwords 21
 - sample databases 23
- VTAM
 - ACBs, and corresponding NetView tasks 33
 - APPL major node 18
 - application programs 88
 - assembling tables 31
 - CDRSC definition statement 29
 - channel-to-channel attachment 95
 - channel-to-NCP attachment 96
 - class-of-serve table 101
 - CNM routing table 100
 - configuration list 25
 - configuration lists 86
 - cross-domain resource managers 92
 - cross-domain resources 91
 - defining NetView to 25
 - definitions 85
 - reviewing for mode and table changes 31
 - load modules for sample network 18

VTAM (*continued*)
 local non-SNA terminals 89
 local SNA devices 89
 logon mode table 101
 major node names 83
 NETID start option 29
 NMC updates 25
 NTRI 95
 PATH definition 96
 SNA topology manager 26
 span of control 124
 SSCP table 92
 start options 32, 86
 start procedure 9, 10, 20, 33, 102
 starting 51
 USS table 100
VTAM definitions 81
VTAMLIB data set 18
VTAMLST data sets
 NETVIEW.V5R1USER.CNM01.VTAMLST 18

W

web browser
 access 47



File Number: S370/4300/30XX-50
Program Number: 5697-ENV



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC31-8872-00

