Tivoli NetView for z/OS MultiSystem Manager User's Guide

Copyright Notice

© Copyright IBM Corporation 1997, 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, BookManager, CICSPlex, FFST, IBMLink, NetView, Nways, OS/2, OS/390, TME, Tivoli Enterprise, 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.

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

Notices

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

Programming Interfaces

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

<table>
<thead>
<tr>
<th>Preface</th>
<th>vii</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who Should Read This Document</td>
<td>vii</td>
</tr>
<tr>
<td>What This Document Contains</td>
<td>vii</td>
</tr>
<tr>
<td>Publications</td>
<td>viii</td>
</tr>
<tr>
<td>Prerequisite and Related Documents</td>
<td>viii</td>
</tr>
<tr>
<td>Accessing Publications Online</td>
<td>viii</td>
</tr>
<tr>
<td>Ordering Publications</td>
<td>viii</td>
</tr>
<tr>
<td>Providing Feedback about Publications</td>
<td>ix</td>
</tr>
<tr>
<td>Contacting Customer Support</td>
<td>ix</td>
</tr>
<tr>
<td>Accessibility Information</td>
<td>ix</td>
</tr>
<tr>
<td>Keyboard Access</td>
<td>ix</td>
</tr>
<tr>
<td>Conventions Used in This Document</td>
<td>x</td>
</tr>
<tr>
<td>Platform-specific Information</td>
<td>x</td>
</tr>
<tr>
<td>Terminology</td>
<td>x</td>
</tr>
<tr>
<td>Reading Syntax Diagrams</td>
<td>xi</td>
</tr>
<tr>
<td>Required Syntax</td>
<td>xi</td>
</tr>
<tr>
<td>Optional Keywords and Variables</td>
<td>xii</td>
</tr>
<tr>
<td>Default Values</td>
<td>xii</td>
</tr>
<tr>
<td>Long Syntax Diagrams</td>
<td>xii</td>
</tr>
<tr>
<td>Syntax Fragments</td>
<td>xii</td>
</tr>
<tr>
<td>Commas and Parentheses</td>
<td>xiii</td>
</tr>
<tr>
<td>Highlighting, Brackets, and Braces</td>
<td>xiv</td>
</tr>
<tr>
<td>Abbreviations</td>
<td>xv</td>
</tr>
</tbody>
</table>

## Part 1. What Is MultiSystem Manager

<table>
<thead>
<tr>
<th>Chapter 1. MultiSystem Manager Overview</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MultiSystem Manager Benefits</td>
<td>3</td>
</tr>
<tr>
<td>How MultiSystem Manager Communicates with Your Networks</td>
<td>3</td>
</tr>
<tr>
<td>About Topology Agents</td>
<td>8</td>
</tr>
<tr>
<td>The Role of the Topology Agent</td>
<td>8</td>
</tr>
<tr>
<td>The Role of the Topology Manager</td>
<td>8</td>
</tr>
<tr>
<td>Dynamic Topology Discovery</td>
<td>8</td>
</tr>
<tr>
<td>Automatic Topology and Status Updates</td>
<td>9</td>
</tr>
<tr>
<td>Easy-to-Use Command Interface</td>
<td>9</td>
</tr>
<tr>
<td>Creating Views</td>
<td>10</td>
</tr>
<tr>
<td>Resolving Network Problems</td>
<td>11</td>
</tr>
<tr>
<td>Automating Network Management</td>
<td>11</td>
</tr>
<tr>
<td>Automation Table Automation</td>
<td>11</td>
</tr>
<tr>
<td>AON Automation</td>
<td>11</td>
</tr>
<tr>
<td>RODM Automation</td>
<td>11</td>
</tr>
<tr>
<td>Creating Applications to Manage New Topologies</td>
<td>11</td>
</tr>
<tr>
<td>Creating Workstation Topology Agents</td>
<td>12</td>
</tr>
<tr>
<td>Creating MVS Topology Managers</td>
<td>12</td>
</tr>
<tr>
<td>External Documentation and Tools</td>
<td>12</td>
</tr>
<tr>
<td>Learning More About MultiSystem Manager Tools</td>
<td>13</td>
</tr>
<tr>
<td>Online Help</td>
<td>13</td>
</tr>
</tbody>
</table>

## Part 2. Using MultiSystem Manager

<table>
<thead>
<tr>
<th>Chapter 2. Customizing the Initialization Statements</th>
<th>19</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample Initialization Files</td>
<td>19</td>
</tr>
<tr>
<td>Defining RODM to MultiSystem Manager</td>
<td>19</td>
</tr>
<tr>
<td>Defining the RUNCMD Retry Count</td>
<td>20</td>
</tr>
</tbody>
</table>
### Table of Contents

#### Defining the Default Autotask
- Defining Exception Views
- Defining Service Points
  - Specifying the SP Parameter for TCP/IP Protocol
  - Specifying the SP Parameter for SNA Protocol
- Additional GETTOPO Parameters
  - HEARTBEAT Parameter
  - Additional Parameters for IP
  - Additional Parameters for LNM
  - Additional Parameters for NetFinity
  - Additional Parameters for the Open Topology Interface
  - Additional Parameters for Tivoli Management Region
- Defining NetView Management Console Views
  - Network Views
  - Using the Default Network Objects
  - Internet Protocol Network Objects
  - LNM Network Objects
  - NetFinity Network Objects
  - Open Topology Interface Network Objects
  - Tivoli Management Region Network Objects
- Creating Network Objects
  - Changing the Name of the Networks Aggregate
  - Changing the Name of a Network Aggregate
  - Displaying Individual Feature Networks
- Network Configuration Diagrams
  - Working with Networks

#### Chapter 3. Operation Overview
- Getting Started
  - Initializing Network Topology and Status
  - Processing Topology Requests
- MultiSystem Manager Views
  - Navigating Views
  - Topology Correlation Across Different Types of Networks
  - Viewing Correlated Resources
- Issuing Commands
  - Example of Issuing a Command in the NetView Management Console
- Resolving Network Problems
  - Finding a Failing Object
  - Finding the Status of an Object
- Understanding View Object Status
  - Setting Status Aggregation Thresholds
  - Setting Aggregation Priorities
- Updating Topology and Status
- Removing MultiSystem Manager Objects from Views
  - Removing Objects Meeting Criteria
  - Removing Real Objects
  - Removing Aggregate Objects
  - Preventing an Object from Being Removed
- Using Online Help
  - Using the NetView Management Console Online Help
  - Using NetView Help

#### Chapter 4. Internet Protocol Network Operation
- MultiSystem Manager Views
  - IP View Objects
  - Finding Resources
  - Navigating Network Views
  - Viewing IP Networks
  - Managing TN3270 Resources
| Chapter 5. LNM Network Operation ................................................. | 75 |
| MultiSystem Manager Views .................................................... | 75 |
| LNM View Objects .................................................................. | 75 |
| Finding Resources .................................................................. | 76 |
| Navigating Network Views ..................................................... | 76 |
| Viewing LNM Networks ......................................................... | 77 |

| Chapter 6. NetFinity Operation .................................................. | 83 |
| MultiSystem Manager Views ....................................................... | 83 |
| NetFinity View Objects .......................................................... | 83 |
| Finding Resources .................................................................. | 84 |
| Navigating Network Views ..................................................... | 84 |
| Viewing NetFinity Networks .................................................... | 84 |

| Chapter 7. Open Topology Interface Network Operation .................. | 91 |
| MultiSystem Manager Views ....................................................... | 91 |
| Open Topology Interface View Objects ....................................... | 91 |
| Finding Resources .................................................................. | 92 |
| Navigating Network Views ..................................................... | 92 |
| Viewing Open Networks .......................................................... | 93 |

| Chapter 8. Tivoli Management Region Operation ............................ | 97 |
| MultiSystem Manager Views ....................................................... | 97 |
| Tivoli Management Region View Objects ....................................... | 97 |
| Finding Resources .................................................................. | 97 |
| Navigating Network Views ..................................................... | 98 |
| Viewing Tivoli Management Region Networks ................................ | 98 |

| Part 3. Appendixes ................................................................. | 105 |

| Appendix A. Initialization Statements ......................................... | 107 |
| Syntax Rules for GETTOPO Statements in the Initialization File ....... | 108 |
| COMMON.FLC_RODMAPPL ......................................................... | 109 |
| COMMON.FLC_RODMNAME ........................................................ | 110 |
| GETTOPO RES and ONLY .......................................................... | 111 |
| (MSM)COMMON.FLC_DEF_NETW_VIEW ....................................... | 112 |
| (MSM)COMMON.FLC_EXCEPTION_VIEW_FILE ................................ | 113 |
| (MSM)COMMON.FLC_RODMINT .................................................. | 114 |
| (MSM)COMMON.FLC_RDMRETRY ............................................... | 115 |
| (MSM)COMMON.FLC_RNCDMTRY ............................................... | 116 |
| (MSM)COMMON.FLC_TCPNAME ................................................. | 117 |
| (MSM)COMMON.FLC_TN3270_FILE ............................................. | 118 |
| (MSM)function.autotask.MSMdefault ........................................ | 119 |

| Appendix B. Resource Information Window—Other Data Field ........... | 121 |
| Internet Protocol .................................................................. | 121 |
| LAN Network Manager .......................................................... | 122 |
| NetFinity ........................................................................... | 124 |
| Open Topology Interface ....................................................... | 125 |
| Tivoli Management Region ..................................................... | 125 |

| Appendix C. Global Variables .................................................... | 127 |

| Appendix D. MultiSystem Manager Traps and Alerts ..................... | 131 |
| Tivoli NetView Topology Manager Traps ..................................... | 131 |
| Tivoli Management Region Alerts ............................................. | 132 |

| Index .................................................................................. | 135 |
Preface

This document describes the MultiSystem Manager component of Tivoli® NetView® for z/OS™, and how it can be used in managing your networks. Also described are:

- How to create and tailor MultiSystem Manager views for your environment.
- How to use the NetView management console (NMC) workstation to navigate through your network views and issue commands to manage your network resources.
- How to resolve some of the most common problems you might encounter when MultiSystem Manager is initialized for the first time.

Who Should Read This Document

This book is intended for network operators, network planners, system designers, and system programmers, who use a NetView management console to monitor or control networks or customize the graphic views to better represent their network configurations. This book can also help those who plan network operations and train operators.

What This Document Contains

This book contains the following sections:

- "Chapter 1. MultiSystem Manager Overview” on page 3 contains an overview of the information in this book.
- "Chapter 2. Customizing the Initialization Statements” on page 19 defines how MultiSystem Manager is used to manage your network.
- "Chapter 3. Operation Overview” on page 45 explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage your networks.
- "Chapter 4. Internet Protocol Network Operation” on page 65 explains how to initialize MultiSystem Manager, start and stop processing, and use the views and command support to manage your Internet Protocol networks.
- "Chapter 5. LNM Network Operation” on page 73 explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage your LNM networks.
- "Chapter 6. NetFinity Operation” on page 83 explains how to initialize MultiSystem Manager, start and stop processing, and use the views and command support to manage your NetFinity networks.
- "Chapter 7. Open Topology Interface Network Operation” on page 91 explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage open networks.
- "Chapter 8. Tivoli Management Region Operation” on page 97 explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage your Tivoli management region networks.
- "Appendix A. Initialization Statements” on page 107 contains the syntax descriptions for the MultiSystem Manager initialization statements. It includes the general MultiSystem Manager definition statements.
Preface

- “Appendix B. Resource Information Window—Other Data Field” on page 121 lists the MultiSystem Manager resource types for each MultiSystem Manager feature, and the contents of the Other data field for each resource type.
- “Appendix C. Global Variables” on page 127 lists the global variables used by the MultiSystem Manager topology manager to store information about MultiSystem Manager.
- “Appendix D. MultiSystem Manager Traps and Alerts” on page 131 provides a numeric listing of the MultiSystem Manager alerts that are sent to the NetView program by the MultiSystem Manager agents.

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>
<thead>
<tr>
<th>Table 1. Resource Web sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tivoli NetView for z/OS</td>
</tr>
</tbody>
</table>

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:


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:

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 Web site:

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.

**Italics** Variables and new terms appear like this, in italics. Words and phrases that are emphasized also appear like this, in italics.

**Monospace** Code examples, output, and system messages appear like this, in a monospace font.

**ALL CAPS** Tivoli NetView for 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](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
- 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/racfss.html](http://www-1.ibm.com/servers/eserver/zseries/zos/security/racfss.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>
<thead>
<tr>
<th>Element Position</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the command line</td>
<td>Required</td>
</tr>
<tr>
<td>Above the command line</td>
<td>Default</td>
</tr>
<tr>
<td>Below the command line</td>
<td>Optional</td>
</tr>
</tbody>
</table>

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

**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.

*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**

```
   RID TASK=opid
      ,STEP
      ,CONTINUE
      ,END
      ,RUN
      ,MODNAME=*
      ,MODNAME= *name
      ,OPTION=*name
      ,OPTION=
      ,HAPIENTR
      ,HAPIEXIT
```

*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 for a syntax with the fragments ReMote and FromTo.

**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.
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 \textit{sessid} variable, enter:

\texttt{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.

\textbf{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.

\textit{Table 3. Syntax Elements Examples}

<table>
<thead>
<tr>
<th>This element...</th>
<th>Looks like this...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keyword</td>
<td>CCPLOADF</td>
</tr>
<tr>
<td>Variable</td>
<td>\textit{resname}</td>
</tr>
<tr>
<td>Operand</td>
<td>MEMBER=membername</td>
</tr>
<tr>
<td>Default</td>
<td>\textit{today} or INCL</td>
</tr>
</tbody>
</table>
Abbreviations

Command and keyword abbreviations are described in synonym tables after each command description.
Part 1. What Is MultiSystem Manager

Chapter 1. MultiSystem Manager Overview

MultiSystem Manager Benefits .................................................. 3
How MultiSystem Manager Communicates with Your Networks .......... 3
About Topology Agents ............................................................ 8
The Role of the Topology Agent ................................................. 8
The Role of the Topology Manager .............................................. 8
  Dynamic Topology Discovery .................................................. 8
  Automatic Topology and Status Updates .................................. 9
  Easy-to-Use Command Interface ............................................. 9
Creating Views ........................................................................ 10
Resolving Network Problems .................................................... 11
Automating Network Management .............................................. 11
  Automation Table Automation ............................................... 11
  AON Automation ................................................................ 11
  RODM Automation ................................................................ 11
Creating Applications to Manage New Topologies ......................... 11
Creating Workstation Topology Agents ...................................... 12
Creating MVS Topology Managers ............................................ 12
External Documentation and Tools ............................................ 12
Learning More About MultiSystem Manager Tools ....................... 13
Online Help ............................................................................ 13
MultiSystem Manager Overview

MultiSystem Manager simplifies the task of network management by “harnessing” the power of Tivoli NetView for z/OS (NetView) to centrally manage your network resources.

MultiSystem Manager features enable you to manage the following types of networks:
- IP networks managed by Tivoli NetView for UNIX®, NetView for AIX®, Tivoli NetView for NT, Tivoli NetView for Sun, HP Network Node Manager for Sun, or HP Network Node Manager for HP-UX
- TCP/IP Stack and TN3270 resources managed by CS/390
- IBM OS/2® LAN Network Manager (LNM) networks
- NetFinity network workstations
- Any network supported by MultiSystem Manager Open topology agents
- Tivoli Management Regions

MultiSystem Manager Benefits

MultiSystem Manager provides an integrated and centralized network management facility that enables you to manage your networks from a NetView management console workstation. To help you manage your networks, MultiSystem Manager provides benefits such as:
- Dynamic topology and status discovery of your networks
- Instant access to graphical views of the topology and status of your networks, all from a single NetView management console workstation
- Quick notification of changes in network topology and status:
  - A token-ring has excessive errors.
  - A bridge is off line.
  - A monitored adapter is not responding.
  - A node is down.
  - A node joined the network.
- The ability to send commands to network resources simply by selecting the resource in a NetView management console view and then selecting the command from a pull-down window
- The ability to integrate the management of various types of networks

In addition, the Open feature enables you, or a vendor, to create a topology agent that manages and monitors your resources.

How MultiSystem Manager Communicates with Your Networks

MultiSystem Manager uses a topology manager-topology agent relationship to manage your network resources. MultiSystem Manager provides a topology manager, which runs on NetView. The topology agent resides in the managed environment and is responsible for forwarding topology and status for all managed network resources to the topology manager.

The topology manager sends queries and commands to the topology agents. Topology agents use alerts and command responses to communicate with the topology manager. Alerts and resolutions are received by the NetView automation
table and processed by the topology manager. Figure 7 shows topology manager and agent communications.

![Diagram](image)

**Figure 7. Communication between MultiSystem Manager and Topology Agents**

The topology manager can use SNA LU 6.2, SNA SSCP-PU, or IP sessions to communicate with the topology agents. If using SNA sessions, the topology manager uses NetView RUNCMD and RMTCMD commands over SNA sessions to communicate with the topology agents. If using IP sessions, the topology manager uses commands over IP sessions. The communication protocol is dependent upon the topology agent with which the topology manager is communicating. Table 4 lists the MultiSystem Manager topology feature and the protocol it uses for communication.

<table>
<thead>
<tr>
<th>MultiSystem Manager Feature</th>
<th>Communication Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Protocol</td>
<td>• IP and SNA LU 6.2 for Tivoli NetView for AIX&lt;br&gt;• IP for Tivoli NetView for NT, Sun, HP&lt;br&gt;Network Node Manager for Sun, and HP-UX</td>
</tr>
<tr>
<td>Internet Protocol TN3270</td>
<td>SNA LU 6.2 and PPI</td>
</tr>
<tr>
<td>LNM</td>
<td>SNA LU 6.2 and SSCP-PU</td>
</tr>
<tr>
<td>NetFinity</td>
<td>• SNA LU 6.2 for OS/2&lt;br&gt;• IP for Windows NT&lt;sup&gt;®&lt;/sup&gt;</td>
</tr>
<tr>
<td>Open Topology Interface</td>
<td>SNA LU 6.2</td>
</tr>
<tr>
<td>Tivoli management region</td>
<td>IP</td>
</tr>
</tbody>
</table>

For SNA communications, the *service point* is the LU or PU of the workstation that communicates with the topology manager. For IP communications, the *service point* is the IP host name of the workstation that communicates with the topology manager.

**Note:** In this book, the term *service point* includes the topology agents communicating with topology managers using TCP/IP.
Figure 8 on page 6 shows the MultiSystem Manager features that use an SNA communication protocol. LNM B communicates through an SNA SSCP-PU connection to a remote NetView. The TN3270 Manager discovers information via communication with CS/390 across a 390 network. The other agents communicate directly to NetView through an SNA LU 6.2 connection.
Figure 9 on page 7 shows the MultiSystem Manager features that use an IP communication protocol.
Figure 9. MultiSystem Manager Environment with IP Connection
About Topology Agents

Topology agents for MultiSystem Manager are available in a variety of ways. In cases such as Internet Protocol (IP) and Tivoli Management Region, the topology agents are shipped with the MultiSystem Manager component of NetView. NetFinity and LAN Network Manager (LNM) topology agents are shipped as part of those products. Open agents can be written by customers or provided by vendors.

The run time and installation characteristics of the topology agents are dependent upon the operating system on which they run. For example, Tivoli management region topology agents are provided in PERL scripts.

Each of the MultiSystem Manager features require that you install the topology agent on the service point workstation.

The Role of the Topology Agent

The role of the topology agent residing on the service point is to monitor the network in which it resides and to dynamically communicate information about changes in network topology or resource status to the topology manager.

When the topology manager issues a command to gather topology and status, the topology agent collects the information and sends it back as part of the command’s response.

The topology agent sends an alert or resolution to the topology manager to indicate changes in topology and status.

The topology manager updates the status of the resource in RODM and reflects this status change in your NetView management console views. Alerts and resolutions are then stored in the event viewer for the resource and can be displayed on your NetView management console workstation.

The Role of the Topology Manager

To monitor and manage your networks, the topology manager performs the following functions:

- dynamically discovers the topology and status and stores it in RODM
- processes the topology and status updates from the topology agents
- integrates with AON to automate recovery of failing resources
- provides an easy-to-use command interface based on Distributed Manager Command Support (DMCS)

Dynamic Topology Discovery

The topology manager begins the process of network management by dynamically discovering the initial topology and status of the resources in your network and storing this information in RODM. After the information is in RODM, you can view your network resources from your NetView management console workstation.

Figure 10 on page 9 shows an example of the types of NetView management console views created by MultiSystem Manager for the LNM feature.
Automatic Topology and Status Updates

After MultiSystem Manager is initialized and the initial topology and status of the network is stored in RODM, the topology manager keeps topology and status up-to-date by receiving updates from the topology agents. The status changes are reflected in your views, and alerts are stored in the NetView management console event viewer.

Easy-to-Use Command Interface

MultiSystem Manager provides an easy-to-use command interface, distributed manager command support (DMCS), that enables you to send commands to the topology agents. DMCS enables you to issue commands from your NetView
management console workstation. DMCS automatically retrieves RODM information that is required to send the command. You can use DMCS in an automation routine, from the NetView operator command line, or from the NetView management console workstation. Figure 11 shows how MultiSystem Manager presents a list of commands on your NetView management console workstation for the IP agent.

Figure 11. A MultiSystem Manager NetView management console Command Menu Example

Creating Views

MultiSystem Manager dynamically builds views that meet the majority of your network management needs, but you might also want to create unique views.

You can monitor your network from a single NetView management console view, or you can create multiple views, with each view reflecting a different grouping of your resources.

You can also integrate your MultiSystem Manager networks with other network views. For example, if you have an SNA network view, you can add your MultiSystem Manager networks to that view.

The NetView program provides several options that enable you to build customized views. The RODM Collection Manager enables you to create dynamic custom views and aggregate objects. For more information, see the Tivoli NetView for z/OS NetView Management Console User's Guide. You can also use BLDVIEWS or Visual BLDVIEWS to create static customized views. For more information, see the VBV statement in the Tivoli NetView for z/OS Administration Reference and the VBVSERV command in the Tivoli NetView for z/OS Command Reference.
Resolving Network Problems

You can use NetView management console menus and facilities to navigate between views and to locate failing resources. Once the failing resource is located, you can simply select the object on your view and send a command to resolve the problem.

Automating Network Management

You can use MultiSystem Manager to automate many network management procedures. In general, there are three types of automation: Automation table, AON, and RODM.

Automation Table Automation

This type of automation is based on user-written applications that react to information received by NetView from the topology agents. MultiSystem Manager adds statements to the NetView automation table to capture alerts and resolutions, and messages to react to them. The NetView automation table provides this information for user-written programs. You can also add statements to the NetView automation table, enabling you to receive updates from the topology agents.

AON Automation

MultiSystem Manager has been integrated with Automated Operations Network (AON) using AON policy to recover failing resources. For more information, refer to the Tivoli NetView for z/OS Automated Operations Network User’s Guide.

RODM Automation

This type of automation is based on applications that access information stored in RODM. RODM automation applications can process within RODM, using RODM methods, or externally using MultiSystem Manager Access or the RODM API. You can write automation applications that react to status changes made by MultiSystem Manager alert processing. You can also write applications that correlate resources reported upon by different topology managers. These applications can react to problems affecting multiple resources, which might have been reported in multiple alerts.

Note: An example of a topology manager, other than MultiSystem Manager, is the SNA topology manager.

MultiSystem Manager uses RODM-based automation and the topology correlation function to dynamically correlate different managed resources to the same aggregate objects.

Creating Applications to Manage New Topologies

You can write applications to monitor and manage resources that are not supported by other MultiSystem Manager features. These applications can be either workstation-based topology agents or MVS-based topology managers.

Workstation-based topology agents are agents that reside on SNA service point workstations. These agents use the MultiSystem Manager Open topology manager to create and manage objects in RODM. MVS-based topology managers reside on the NetView host. MVS-based topology managers either perform the function of a
topology agent at the host or communicate with a remote topology agent that is not at an SNA service point. MVS-based topology managers use the MultiSystem Manager Access facility to create and manage objects in RODM, rather than using the Open API and the Open topology manager.

The documentation and tools you need depend on the type of topology application you want to create.

In addition to writing topology applications, several Tivoli Partner organizations provide workstation-based topology agents that manage resources that are not discovered by MultiSystem Manager. Refer to the web address, http://www.developer.ibm.com/ for more information.

You can also obtain MVS-based NetView topology managers that use MultiSystem Manager Access. An example is CICSPlex® Systems Manager for MVS/ESA.

Creating Workstation Topology Agents

The following documentation and tools will help you create workstation-based topology agents:

- Tivoli NetView for z/OS Data Model Reference
- Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide
- MultiSystem Manager: Topology Agent Developer’s Guide
- MultiSystem Manager Topology Agent Developer’s Toolkit
- MultiSystem Manager Sample Topology Agent

For a description of these items and how you can obtain them, see External Documentation and Tools.

Creating MVS Topology Managers

The following documentation and tools will help you create MVS-based topology managers:

- Tivoli NetView for z/OS Data Model Reference
- Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide
- MultiSystem Manager Access
- MultiSystem Manager Sample MVS Data Model Application

For a description of these items, and how you can obtain them, see External Documentation and Tools.

External Documentation and Tools

Documentation and tools for building a MultiSystem Manager Open Topology Interface agent application can be uploaded from an internet web browser by pointing to the Tivoli NetView for z/OS Tools Download page at http://www.tivoli.com/support/downloads/netview_390/tools/ and then selecting downloads from the MSMTOOLK package.

Tools and selected patches for MultiSystem Manager agents can be downloaded from an FTP site. Use the ANONYMOUS password at ftp.tivoli.com and look in the support/netview390/msm/ directory. You can also upload files from this FTP site by pointing your Web browser at ftp://ftp.tivoli.com/support/netview390/msm/
and opening a file and saving it to your disk. You can also download from this web page at http://www.tivoli.com/nv390_supported.

Here is an overview of the publications:

- **MultiSystem Manager: Topology Agent Developer’s Guide** describes how to write an Open topology agent application program. This document is available from the MSMTOLK package at Web site: http://www.tivoli.com/support/downloads/netview_390/tools/.
- MultiSystem Manager Sample Topology Agent contains the "C" language source code for a Topology Agent. This sample application demonstrates use of the Topology Agent Developer's Toolkit. This sample application is available from the MSMTOLK package at Web site: http://www.tivoli.com/support/downloads/netview_390/tools/.
- **Tivoli NetView for z/OS Data Model Reference** describes the NetView data models.
- **Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide** describes writing RODM applications and methods, as well as creating objects in RODM that use the GMFHS data model. See the **Tivoli NetView for z/OS User’s Guide** for the order number for printed copies of this publication.

**Note:** Most of the publications are also available on the web at [http://www.tivoli.com/nv390](http://www.tivoli.com/nv390).

- MultiSystem Manager Topology Agent Developer’s Toolkit contains a set of C language functions that create data streams used to build NetView management console views. This toolkit is available from the MSMTOLK package at the following Web site: http://www.tivoli.com/support/downloads/netview_390/tools/.
- MultiSystem Manager Sample MVS Data Model Application contains REXX samples that enable you to create and update objects in RODM from an MVS application using the Open data model. The sample application is provided on the MultiSystem Manager product media in the CNMSAMP data set as FLCSOX01.
- MultiSystem Manager Access provides a fast and efficient REXX interface to RODM. This document also describes how to create a topology manager application on MVS if you do not want to use the Open API to support a remote topology agent. Refer to **Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide** for more information about RODM tools.

### Learning More About MultiSystem Manager Tools

You can use the World Wide Web to learn more about Tivoli NetView for z/OS MultiSystem Manager, view sample graphic displays and upload tools and documentation. Visit the Tivoli NetView for z/OS home page at [http://www.tivoli.com/nv390/](http://www.tivoli.com/nv390/).

For an overview of Tivoli network computing products, and a link to the Tivoli NetView for z/OS MultiSystem Manager home page, visit: [http://www.tivoli.com/](http://www.tivoli.com/)

### Online Help

MultiSystem Manager provides online help for commands and messages. This help is available on your NetView management console workstation, and from your NetView command line.

You can obtain contextual help for MultiSystem Manager supported commands and messages from the NetView management console help facility. You can obtain
help for the topology commands and all MultiSystem Manager messages at your NetView management console by entering the NetView HELP command.

For information about using the MultiSystem Manager online help, refer to "Using Online Help" on page 62.
## Part 2. Using MultiSystem Manager

### Chapter 2. Customizing the Initialization Statements

- Sample Initialization Files ........................................ 19
- Defining RODM to MultiSystem Manager ......................... 19
- Defining the RUNCMD Retry Count ............................... 20
- Defining the Default Autotask .................................... 20
- Defining Exception Views .......................................... 20
- Defining Service Points ........................................... 20
  - Specifying the SP Parameter for TCP/IP Protocol .............. 21
  - Specifying the SP Parameter for SNA Protocol ................ 21
    - Defining Topology Agents in the Same NetView Domain .... 23
    - Defining Agents in the Same Network but Different Domains 24
    - Defining Agents in Different Networks ....................... 24
- Additional GETTOPO Parameters ................................ 24
  - HEARTBEAT Parameter ........................................... 24
- Additional Parameters for IP ..................................... 24
  - IPRES and IPONLY Parameters ................................. 24
  - HOSTS Parameter ................................................. 25
  - HIDDEN Parameter ............................................... 25
  - UNMANAGED Parameter .......................................... 25
  - Selecting a Specific Tivoli NetView Map ...................... 26
- Additional Parameters for LNM ................................... 26
  - LNMRES and LNMONLY Parameters ............................. 26
  - Optional Parameters ............................................ 26
- Additional Parameters for NetFinity ............................. 26
  - NFRES and NFONLY Parameters ............................... 26
  - Optional Parameters ............................................ 27
- Additional Parameters for the Open Topology Interface ........ 27
  - OPENRES Parameter .............................................. 27
  - HOSTONLY Parameter ............................................ 27
  - ELEMENT Parameter .............................................. 27
- Additional Parameters for Tivoli Management Region .......... 27
  - TMERES and TMEONLY Parameters ............................. 27
  - Optional Parameters ............................................ 28
- Defining NetView Management Console Views ..................... 28
  - Network Views ................................................... 29
    - Using the Default Network Views ............................ 29
    - Creating and Customizing Network Views .................... 30
  - Using the Default Network Objects ............................ 32
  - Internet Protocol Network Objects ............................ 33
    - Using the Default Objects ................................... 33
  - LNM Network Objects .......................................... 34
    - Using the Default Objects ................................... 34
  - NetFinity Network Objects .................................... 35
    - Using the Default Objects ................................... 35
  - Open Topology Interface Network Objects ...................... 35
    - Using the Default Objects ................................... 35
  - Tivoli Management Region Network Objects .................... 36
    - Using the Default Network Objects ......................... 36
- Creating Network Objects ........................................ 36
  - Changing the Name of the Networks Aggregate ................ 36
  - Changing the Name of a Network Aggregate .................... 37
  - Displaying Individual Feature Networks ....................... 37
- Network Configuration Diagrams ................................ 37
  - Working with Networks .......................................... 37
    - A Network Aggregate .......................................... 37
    - A New View and a Network Aggregate ........................ 40
## Chapter 3. Operation Overview

### Getting Started
- Initializing Network Topology and Status
  - Initialization Steps
  - Issuing the Focal Point Commands
  - Issuing the INITTOPO Command
- Processing Topology Requests
  - Suspending Processing of Topology Requests
  - Resuming Processing of Topology Requests
  - Displaying Topology Processing Information

### MultiSystem Manager Views
- Navigating Views
- Topology Correlation Across Different Types of Networks
- Viewing Correlated Resources
  - Workstation Aggregate Objects Created by MultiSystem Manager Agents
  - Customization Options
  - Setting Free-Form Correlation Values
- Issuing Commands
  - Example of Issuing a Command in the NetView Management Console
- Resolving Network Problems
  - Finding a Failing Object
  - Finding the Status of an Object
- Understanding View Object Status
  - Setting Status Aggregation Thresholds
  - Setting Aggregation Priorities
- Updating Topology and Status
- Removing MultiSystem Manager Objects from Views
  - Removing Objects Meeting Criteria
  - Removing Real Objects
  - Removing Aggregate Objects
  - Preventing an Object from Being Removed
- Using Online Help
  - Using the NetView Management Console Online Help
  - Using NetView Help

## Chapter 4. Internet Protocol Network Operation

### MultiSystem Manager Views
- IP View Objects
- Finding Resources
- Navigating Network Views
- Viewing IP Networks
  - IP Networks View
  - IP Network Views
  - IP Subnetwork Views
  - Location Views
  - Segment Views
  - Router, Bridge, Hub, Host, and Link Views
- Managing TN3270 Resources

## Chapter 5. LNM Network Operation

### MultiSystem Manager Views
- LNM View Objects
- Finding Resources
- Navigating Network Views
- Viewing LNM Networks
  - LNM LAN Networks View
  - LNM Views
  - Segment Views
### Chapter 6. NetFinity Operation

- MultiSystem Manager Views .................................................. 83
- NetFinity View Objects ............................................................ 83
- Finding Resources ..................................................................... 84
- Navigating Network Views ......................................................... 84
- Viewing NetFinity Networks ....................................................... 84
  - NetFinity Networks View .......................................................... 85
  - NetFinity Network Views .......................................................... 86
  - NetFinity Group Views ............................................................. 87
  - NetFinity System Views .......................................................... 87

### Chapter 7. Open Topology Interface Network Operation

- MultiSystem Manager Views .................................................. 91
  - Open Topology Interface View Objects ..................................... 91
  - Finding Resources ..................................................................... 92
  - Navigating Network Views ......................................................... 92
  - Viewing Open Networks .......................................................... 93
  - Open Networks View .............................................................. 93

### Chapter 8. Tivoli Management Region Operation

- MultiSystem Manager Views .................................................. 97
  - Tivoli Management Region View Objects ................................. 97
  - Finding Resources ..................................................................... 97
  - Navigating Network Views ......................................................... 98
  - Viewing Tivoli Management Region Networks .......................... 98
    - Tivoli Networks View ............................................................ 99
    - Tivoli Network Views ............................................................ 100
    - Tivoli Management Region Views ........................................... 101
    - Tivoli Management Region Lightweight Client Framework Gateways and End Points 102
    - Tivoli Management Region Managed Node Views ...................... 103
Chapter 2. Customizing the Initialization Statements

The initialization statements in the CNMSTYLE file define how MultiSystem Manager is to manage your network. You can customize the initialization statements to do the following:

- Define the Resource Object Data Manager (RODM).
- Set the number of times a failed RUNCMD command is retried.
- Specify the name of the NetView autotask that is used for default processing.
- Design network views.

Many of the initialization statements have default values. You can customize the initialization statements to specify values that meet the needs of your enterprise.

In addition to the initialization statements specified in CNMSTYLE, you can code GETTOPO statements in a MultiSystem Manager initialization file. This enables you to retrieve topology and status for your managed networks when you initialize MultiSystem Manager.

If you have multiple topology features, you can include multiple initialization files in a single file by using the %INCLUDE statement.

Sample Initialization Files

MultiSystem Manager provides a sample initialization file named FLCAINP. This file resides in the DSIPARM data set.

MultiSystem Manager also provides sample initialization files that contain examples of GETTOPO statements for each feature. These files are included by the sample file FLCAINP. Table 5 shows each MultiSystem Manager feature and its sample initialization file. Your initialization files must reside in a DSIPARM data set.

<table>
<thead>
<tr>
<th>MultiSystem Manager Feature</th>
<th>Sample Initialization File</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Protocol</td>
<td>FLCSIIP</td>
</tr>
<tr>
<td>Internet Protocol TN3270</td>
<td>FLCS3270</td>
</tr>
<tr>
<td>LNM</td>
<td>FLCSILNM</td>
</tr>
<tr>
<td>NetFinity</td>
<td>FLCSINF</td>
</tr>
<tr>
<td>Open Topology Interface</td>
<td>FLCSIOPN</td>
</tr>
<tr>
<td>Tivoli management region</td>
<td>FLCSITME</td>
</tr>
</tbody>
</table>

See “Appendix A. Initialization Statements” on page 107 for more information about initialization statement syntax and coding restrictions.

Defining RODM to MultiSystem Manager

Specify the following initialization statements in CNMSTYLE to define RODM to MultiSystem Manager:
**COMMON.FLC_RODMNAME**

The name of the RODM you specify should be the same name you used to define this RODM to GMFHS. This statement is required.

**(MSM)COMMON.FLC_RODMINT**

The amount of time (in seconds) between retries of a RODM request that failed because RODM is checkpointing. When RODM is checkpointing a disk, it cannot process certain transactions, so MultiSystem Manager must wait until the checkpoint process is finished. If MultiSystem Manager tries but cannot access the information because RODM is checkpointing, it waits and tries again.

This statement is optional. If you do not code a value for this statement, MultiSystem Manager waits five seconds between each retry.

**(MSM)COMMON.FLC_RODMRETRY**

The number of times MultiSystem Manager retries a RODM request that failed because RODM is checkpointing. When RODM is checkpointing a disk, it cannot process certain transactions, so that MultiSystem Manager must wait until the checkpoint process is finished. If MultiSystem Manager cannot access the information because RODM is checkpointing, it waits and tries again.

This statement is optional. If you do not code a value for this statement, MultiSystem Manager retries three times.

---

### Defining the RUNCMD Retry Count

Use the **(MSM)COMMON.FLC_RUNCMDRETRY** statement in the CNMSTYLE file to specify the number of times you want MultiSystem Manager to retry a RUNCMD that could not be sent because the SNA session was busy (SENSE 0851). This statement is optional. If you do not code a value for this statement, MultiSystem Manager retries three times.

---

### Defining the Default Autotask

Use the **(MSM)function.autotask.MSMdefault** statement to specify the name of the default autotask that MultiSystem Manager should use during GETTOPO processing. When the autotask keyword is not specified on the GETTOPO command and the service point object for the GETTOPO command is not stored in RODM then the default autotask is used.

This statement is required. If you do not code a value for this statement, MultiSystem Manager fails to initialize. MultiSystem Manager supplies a sample autotask, AUTOMSMD, in sample FLCSOPF. Assign AUTOMSMD to be the default autotask.

---

### Defining Exception Views

To specify the name of the file that contains the information about processing exception views for MultiSystem Manager resources, use the **(MSM)COMMON.FLC_EXCEPTION_VIEW_FILE** statement. This statement is optional.

---

### Defining Service Points

Service points can be defined to MultiSystem Manager by specifying a GETTOPO statement in the initialization file.
The workstation where the topology agent is installed is called the service point. The service point workstation communicates with the topology manager. As used in this book, the term service point extends past the traditional SNA definition to include TCP/IP. The service point name is specified in the SP parameter on the GETTOPO command. This parameter is required on the GETTOPO statement.

The topology manager can communicate with topology agents through SNA or TCP/IP protocol. The communication protocol is dependent upon the MultiSystem Manager feature that you are implementing. Table 6 lists each MultiSystem Manager topology feature and the associated communication protocols that each feature supports.

Table 6. MultiSystem Manager Topology Feature and Communications Protocol

<table>
<thead>
<tr>
<th>MultiSystem Manager Feature</th>
<th>Communication Protocol</th>
</tr>
</thead>
</table>
| Internet Protocol           | • IP and SNA LU 6.2 for Tivoli NetView for AIX  
   | • IP for Tivoli NetView for NT, Sun, HP  
   | Network Node Manager for Sun, and HP-UX |
| Internet Protocol TN3270    | SNA LU 6.2 and PPI     |
| LNM                         | SNA LU 6.2 and SSCP-PU |
| NetFinity                   | • SNA LU 6.2 for OS/2  
   | • IP for Windows NT     |
| Open Topology Interface     | SNA LU 6.2             |
| Tivoli management region    | IP                     |

In some cases, it is necessary to code additional parameters, such as REMOTE and APPL, that are closely related to the SP parameter. These related parameters are discussed in the following sections, where applicable.

**Specifying the SP Parameter for TCP/IP Protocol**

For topology agents that communicate with the topology manager using an IP connection, the SP parameter is the TCP/IP host name of the service point workstation. This can be either the simple TCP/IP host name or the fully-qualified host name. Even if you specify a fully-qualified host name, the host name must be unique. You cannot specify two host names that are the same, even though they reside in different subnets.

The host name for NetFinity for Windows NT and IP must be preceded by an ampersand (&) to specify a TCP/IP connection.

**Specifying the SP Parameter for SNA Protocol**

For topology agents that communicate with the topology manager using an SNA connection, the SP parameter is the LU or PU name of the service point workstation. The LU name is specified when an LU6.2 connection is used to communicate with the topology manager. The PU name is used when an SSCP-PU session is used.

If the service point does not reside in the same SNA domain or network as MultiSystem Manager, and the connection to the service point workstation is LU6.2, specify the fully-qualified SNA name when coding the SP parameter. The
format of a fully qualified SNA name is \textit{sp\_netid.sp\_domain.sp\_name}, where \textit{sp\_netid} and \textit{sp\_domain} are the names of the remote SNA network and domain where the service point resides.

If the service point does not reside in the same SNA domain or network as MultiSystem Manager and the connection to the service point workstation is SSCP-PU, you must also code the \textit{REMOTE} parameter.

See \textbf{Figure 12} for guidelines on coding the SP and REMOTE parameters.

<table>
<thead>
<tr>
<th>MultiSystem Manager and the agent are in:</th>
<th>Same Domain</th>
<th>Different Domain but Same Network</th>
<th>Different Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same Domain</td>
<td>Code:</td>
<td>Code:</td>
<td>Code:</td>
</tr>
<tr>
<td></td>
<td>\textit{SP=LU name}</td>
<td>\textit{SP=LU name}</td>
<td>\textit{SP=remote_network_name.remote_domain_name.LU name}</td>
</tr>
<tr>
<td>Different Domain but Same Network</td>
<td>Code:</td>
<td>Code:</td>
<td>Code:</td>
</tr>
<tr>
<td></td>
<td>\textit{SP=PU name}</td>
<td>\textit{SP=PU name}</td>
<td>\textit{SP=PU name}</td>
</tr>
<tr>
<td></td>
<td>\textit{REMOTE=remote_domain_name}</td>
<td>\textit{REMOTE=remote_network_name.remote_domain_name}</td>
<td></td>
</tr>
</tbody>
</table>

\textbf{Figure 12. Coding the SP and REMOTE Parameters}

If you are using the IP or Open features, the \textit{APPL} parameter is also required. The \textit{APPL} parameter is used to define the name of the application on the service point machine that communicates with the topology manager.

\textbf{Figure 13 on page 23} shows various ways your network might be configured. This sample LNM network consists of two SNA networks (NET10 and NET20). NET10 consists of two NetView domains (CNM01 and CNM02). NET20 consists of a single NetView domain (CNM03). Because MultiSystem Manager is in NetView domain CNM02, domains CNM01 and CNM03 are considered remote domains. Because MultiSystem Manager resides in network NET10, NET20 is considered a remote network.
Defining Topology Agents in the Same NetView Domain

Figure 13 shows two LNM topology agents residing in the same NetView domain (CNM02) as MultiSystem Manager. These agents are contained in LNM B and LNM C. LNM B communicates over an LU 6.2 session and LNM C communicates over an SSCP-PU session.

To define these topology agents to MultiSystem Manager, create an initialization file statement for each LNM:

GETTOPO LNMRES, SP=LUB, ...(other parameters)
Defining Agents in the Same Network but Different Domains

Figure 13 on page 23 shows two LNM topology agents that are in a different domain than MultiSystem Manager, but are still in the same SNA network (NET10). These are in LNM A and LNM E.

Code the REMOTE parameter for LNM E, because it communicates with the remote NetView CNM01 by means of an SSCP-PU session. LNM A is not in the same domain as MultiSystem Manager, but it is not necessary to specify the REMOTE parameter because it is connected by LU6.2.

For LNM E, code:
GETTOPO LNMRES, SP=PUE, REMOTE=CNM01, ...(other parameters)

For LNM A, code:
GETTOPO LNMRES, SP=LUA, ...(other parameters)

It is not necessary to code the network name on the SP and REMOTE parameters because both LNMs are in the same SNA network (NET10) as MultiSystem Manager.

Defining Agents in Different Networks

Figure 13 on page 23 also shows LNM agents that are in a different network than MultiSystem Manager. In our example, this agent is in LNM D. Because LNM D is using an LU 6.2 session, code the remote network name, the remote domain name, and the LU name on the SP parameter shown in the following example:
GETTOPO LNMRES, SP=NET20.CNM03.LUD, ...(other parameters)

Additional GETTOPO Parameters

You can use additional GETTOPO parameters within the initialization file to define your topology agents. Some of these parameters are optional. Specify, in your GETTOPO statement, the parameters needed to best manage the resources in your network.

HEARTBEAT Parameter

The HEARTBEAT parameter provides a means of notification if you lose communication with a service point. MultiSystem Manager polls the agent at a specified interval. If the agent fails to respond, MultiSystem Manager sets the status of the agent to unsatisfactory and sets the status of all the objects owned by that agent to unknown.

Additional Parameters for IP

This section describes the additional GETTOPO parameters for IP.

IPRES and IPONLY Parameters

When you define the service point for a network managed by the MultiSystem Manager IP agent, you can specify that MultiSystem Manager gathers topology and status on the topology agent and all of its managed IP resources or only the topology agent.

MultiSystem Manager uses initialization statements, which are in the form of GETTOPO IPRES and GETTOPO IPONLY topology commands, to gather the initial network topology and status. If you code the IPRES parameter, MultiSystem
Manager gathers topology and status on the topology agent and all its managed IP resources such as subnetworks, segments, routers, bridges, hubs, hosts, and interfaces.

If you specify the IPONLY parameter, MultiSystem Manager gathers topology and status only on the topology agent. Although you can specify either of the parameters in the initialization file, specify the IPRES parameter on each statement to gather the most complete initial information.

**HOSTS Parameter**

When you define your topology agents, the initialization file enables you to choose the types of resources you want to manage. You can choose whether you want to include the hosts and host adapters of the network. If you want to include hosts and host adapters in your managed IP resources, specify HOSTS=YES (the default) on your GETTOPO initialization file statements.

If you specify HOSTS=NO, MultiSystem Manager gathers topology and status and manages only subnetworks, routers, bridges, hubs, and segments. Hosts and host interfaces are not managed.

If you specify HOSTS=NO, but later need information on your hosts to do problem determination, you can issue a GETTOPO IPRES or IPDETAIL command with HOSTS=YES to gather topology and status on the hosts and host interfaces.

If you want to decrease the network traffic flow and the amount of NetView and RODM storage that MultiSystem Manager uses during initialization, specify HOSTS=NO on the GETTOPO IPRES statement.

**HIDDEN Parameter**

As part of managing your IP environment, you can choose to hide resources on your local map. MultiSystem Manager gives you the ability to manage these hidden resources. If you want to manage hidden resources, specify HIDDEN=YES on your GETTOPO IPRES initialization file statements.

If you specify HIDDEN=YES, MultiSystem Manager gathers topology and status on the hidden resources, stores this information in RODM, and creates these objects in your views.

If you specify HIDDEN=NO, which is the default, MultiSystem Manager does not collect information about hidden resources and does not display these resources in your views.

**UNMANAGED Parameter**

As part of managing your IP environment, you can choose to display unmanaged resources. The UNMANAGED parameter on your GETTOPO IPRES initialization file statements determines if IP unmanaged network resources are displayed.

If you specify UNMANAGED=YES, which is the default, MultiSystem Manager gathers topology and status on the unmanaged resources, stores this information in RODM, and creates these objects in your views.

If you specify UNMANAGED=NO, MultiSystem Manager does not collect information about unmanaged resources or display them.

If you want to decrease the amount of processing done by the IP agent during initialization, specify UNMANAGED=NO on the GETTOPO IPRES statement.
Selecting a Specific Tivoli NetView Map
When you initialize your network, MultiSystem Manager gathers topology and status on the map currently being used by Tivoli NetView for UNIX, Sun, NT, or HP Network Node Manager. This map may not be the map that accurately reflects the IP resources you want to manage. To ensure management of a consistent picture of your managed IP resources, specify the specific map you want to initialize by using the MAP parameter on your GETTOPO IPRES statement. If the specified map is not active when you initialize MultiSystem Manager, the GETTOPO command fails and views for that IP network are not created. By activating the map and reissuing the GETTOPO IPRES statement for the specified map, you can create network views.

Additional Parameters for LNM
This section describes the additional GETTOPO parameters for LNM.

LNMRES and LNMONLY Parameters
When you define LNM networks, you can specify that MultiSystem Manager gathers topology and status on the LNM and all of its managed resources, or only on LNM.

MultiSystem Manager uses initialization file statements, which are in the form of GETTOPO LNMRES and GETTOPO LNMONLY topology commands, to gather the initial network topology and status. If you define the LNMRES parameter, MultiSystem Manager gathers topology and status for LNM and its managed resources.

If you specify the LNMONLY parameter, MultiSystem Manager gathers topology and status only on the LNM. Although you can specify either parameter in the initialization file, to gather the most complete initial information, specify the LNMRES parameter on each statement.

Optional Parameters
When you define topology agents in the initialization file, you can choose the types of resources you want to monitor. You can choose to monitor only the LNMs or the LNMs and their monitored resources.

If you do not specify what resources should be monitored, the MultiSystem Manager default is to monitor the LNMs and all the resources that the LNMs monitor. By using GETTOPO LNMONLY on an LNM initialization statement, you can request that the topology agent monitor only the LNM.

ADAPTERS Parameter: If you want to decrease the network traffic flow during initialization, specify ADAPTERS=NONE on the GETTOPO statement.

If you specify ADAPTERS=NONE, but later want to diagnose problems on a particular segment, you can issue a GETTOPO LNMSEG command with ADAPTERS=MONITORED or ADAPTERS=ALL to gather topology and status information.

Additional Parameters for NetFinity
This section describes the additional GETTOPO parameters for NetFinity.

NFRES and NFONLY Parameters
For a NetFinity network, you can define parameters to enable MultiSystem Manager to gather topology and status of the NetFinity agent and all of its managed resources, or only the NetFinity service point.
MultiSystem Manager uses initialization file statements, which are in the form of GETTOPO NFRES and GETTOPO NFONLY topology commands, to gather the initial network topology and status. To gather the topology and status from your NetFinity service point, and its managed resources, specify the NFRES parameter.

Use the NFONLY parameter to gather topology and status only for the NetFinity service point workstation. Although you can specify either parameter in the initialization file, specify the NFRES parameter on each statement to gather the most complete initial information.

**Optional Parameters**

NFPROCESS, NFSECURITY, and NFSYSMONITOR Parameters: To decrease network traffic during initialization, specify NFPROCESS=NO, NFSECURITY=NO and NFSYSMONITOR=NO on the GETTOPO statement or use the default. If you later want to diagnose problems on a particular workstation, issue a GETTOPO NFWKST command to that workstation with NFSYSMONITOR=ALERTSONLY, or NFSYSMONITOR=ALL, for example, to gather topology and status information.

**Additional Parameters for the Open Topology Interface**

This section describes the additional GETTOPO parameters for Open.

**OPENRES Parameter**

MultiSystem Manager uses initialization file statements, which are in the form of GETTOPO OPENRES topology commands, to gather the initial network topology and status.

**HOSTONLY Parameter**

MultiSystem Manager also uses GETTOPO HOSTONLY commands to retrieve status and information for Open topology agents and update them in RODM.

**ELEMENT Parameter**

If your topology agent (defined by the APPL parameter) supports multiple sub-applications, you can use the ELEMENT parameter to send the GETTOPO commands to that element manager.

For example, if your Open topology agent, APPLA, is composed of several sub-applications that perform different functions, and if you designed a sub-application to handle GETTOPO command processing, called sub-application GTAPPL, you can code APPL=APPLA and ELEMENT=GTAPPL. When the initialization file is processed, MultiSystem Manager sends the GETTOPO command to APPLA, but indicates that this command should be routed to sub-application GTAPPL.

**Additional Parameters for Tivoli Management Region**

This section describes the additional GETTOPO parameters for Tivoli management region.

**TMERES and TMEONLY Parameters**

When you define Tivoli management region topology agents, you can enable the MultiSystem Manager to gather topology and status for the topology agent and all of the resources it manages, or only the topology agent.

MultiSystem Manager uses initialization statements, that are in the form of GETTOPO TMERES and GETTOPO TMEONLY topology commands to gather the initial network topology and status. If you code the TMERES parameter,
MultiSystem Manager gathers topology and status on the topology agent and all of the Tivoli management region resources it manages, such as managed nodes and monitors.

If you code the TMEONLY parameter, MultiSystem Manager gathers topology and status only on the topology agent. Although you can specify either of the parameters in the initialization file, specify the TMERES parameter on each statement to gather the most complete initial information.

You can also configure the MSMAGENT.cfg file to discover topology for the resources on the Tivoli management region that contains the MultiSystem Manager Tivoli management region agent (hereafter denoted as the local Tivoli management region.) The MSMAGENT.cfg file contains the TMR= parameter. The default value is TMR=ALL which causes the Tivoli management region agent to discover topology for resources on the local Tivoli management region and on all connected Tivoli management regions. Change this parameter to TMRI=Local to discover topology for the local Tivoli management region only.

**Optional Parameters**

When defining topology agents in the initialization file, you can choose the resources you want to monitor. Optional parameters, such as ENDPOINT, POLICY, and MONITORS enable you to choose which resources you want to monitor.

**ENDPOINT Parameter:** This parameter enables you to specify whether or not to display endpoint information. If you choose the default, ENDPOINT=NO, no information will be displayed. Specify ENDPOINT=YES to display endpoint information. Specifying ENDPOINT=YES increases the time for the GETTOPO command to complete.

**POLICY Parameter:** This parameter enables you to specify whether or not to display managed nodes under the policy regions in the view. If you choose the default, POLICY=YES, the managed nodes under the policy regions will be displayed. If you do not want to display the managed nodes, you must specify POLICY=NO.

**MONITORS Parameter:** This parameter enables you to specify whether or not to display Tivoli management region distributed monitoring information gathered for each managed node with your topology data. If you choose the default, MONITORS=YES, then the information will be displayed. Specify MONITORS=NO if you do not want to display information about monitors. Specifying MONITORS=YES increases the time for the GETTOPO command to complete.

---

**Defining NetView Management Console Views**

With MultiSystem Manager, you can create graphical views showing your networks at various levels of detail. These views are based on the information specified in the initialization file. You can create and name network views and objects that are to be displayed in these network views.

This section describes how to customize network views and network objects by using the NETWORK_VIEW, NETWORK_AG_OBJECT, and NETWORK_NAME keywords. If you customize network views or network objects, specify these keywords on the first GETTOPO command that you issue to each service point. In most cases, the first GETTOPO command to each service point should be specified in the initialization file.
If the customized information is stored in RODM, it is not necessary to specify the customized values again in subsequent GETTOPO commands to each service point. This simplifies the task of writing additional automation routines for these managed resources.

**Network Views**

The Details NetView management console window lists the highest level MultiSystem Manager network views. The Details window, as shown in Figure 14, contains the default network view, MultiSysView.

Each MultiSystem Manager feature will be added to the NetView management console component list during initialization. The feature does not appear in the details window until it is defined in the initialization file. Figure 14 shows the NetView management console component list.

**Note:** The NetView management console, by default, displays views using icons. To change your symbols to match those displayed in this book, select the View pull-down menu, and select Show Icons to switch to shapes.

![NetView management console Window](image)

**Using the Default Network Views**

A single network view named MultiSysView is created by default. All defined networks are displayed in this view. Figure 15 on page 30 shows the MultiSysView. These views show the default high level networks object for all of the MultiSystem Manager features. To simplify the customization examples, the remaining portion of this chapter refers to a subset of MultiSystem Manager features.
Creating and Customizing Network Views

You can create unique views by using GETTOPO commands or the view customization utilities, BLDVIEWS and Visual BLDVIEWS. This chapter describes using GETTOPO commands to create your views. For information about BLDVIEWS, see the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer's Guide. For more information about Visual BLDVIEWS refer to the online help.

You can customize network views by changing the name of the default network view (MultiSysView) or by creating additional network views to manage network operations more efficiently.

Changing the Name of the Default View: The MultiSystem Manager default is a single network view named MultiSysView. To change the name of the default network view, modify the (MSM)COMMON.FLC_DEF_NETW_VIEW statement in the CNMSTYLE file. If you want a NetView management console operator to select a view called 1st_Shift_View rather than MultiSysView for the LNM agent, specify this statement:

(MSM)COMMON.FLC_DEF_NETW_VIEW=1st_Shift_View/LAN Networks for 1st Shift

1st_Shift_View can be created for each MultiSystem feature. When you initialize MultiSystem Manager using this initialization statement, an entry called 1st_Shift_View is displayed in the list of views on your NetView management console. Open this view to see the networks monitored by the first shift. Figure 16 on page 31 shows how your Details window might look for the LNM feature on NetView management console.
Creating Additional Network Views: You can also create additional network views. For example, if you want to view a smaller subset of networks during your off-shift operations, create another view called Off_Shift_View. To create the off-shift view for the LNM feature, specify the NETWORK_VIEW parameter on the GETTOPO statement as follows:

```
NETWORK_VIEW=Off_Shift_View/LAN Networks for Off Shift
```

When you initialize MultiSystem Manager using the NETWORK_VIEW parameter, the name of the new off-shift view is added to the list shown in Figure 17 on page 32 on the NetView management console.
For additional examples of customizing views, see "Network Configuration Diagrams" on page 37.

**Using the Default Network Objects**

MultiSystem Manager groups all networks by type, for example, all IP networks will be grouped under one network aggregate object called IP_Networks, by default, shown in [Figure 15 on page 30](#).

By default, MultiSystem Manager creates a network object named `spname_MSMfeature_Network` for each network. The `spname` string is either the LU portion of the SNA network address or the TCP/IP host name coded on the SP parameter of the GETTOPO statement for the network. The `MSMfeature` string names the MultiSystem Manager feature to which this network belongs (such as IP or NetFinity). [Figure 18 on page 33](#) shows an example of the default network objects that may appear in your NetView management console views.
If you do not want to use the default network object created by MultiSystem Manager, define the NETWORK_NAME or the NETWORK_AG_OBJECT parameters in the GETTOPO statements in the initialization file.

Internet Protocol Network Objects

MultiSystem Manager creates the following aggregate objects that can be customized:

- Networks aggregate object
  Represents a group of IP networks and has a resource type of Internet networks.
- Network object
  Represents a single IP network and has a resource type of Internet network aggregate.

Using the Default Objects

MultiSystem Manager, by default, groups all IP networks together under a single network aggregate object named IP_Network, as shown in Figure 15 on page 30.

By default, MultiSystem Manager creates a network object, spname_IP_Network, for each IP network. The spname string represents the value coded on the SP parameter on the GETTOPO statement for that network.

If you do not want to use the default objects created by MultiSystem Manager, define the NETWORK_NAME or the NETWORK_AG_OBJECT parameters in the GETTOPO statements in the initialization file. See Figure 18 for an example of the default network object.

If you installed the TN3270 Manager extension of MultiSystem Manager IP, then additional objects are added. They appear under the IP objects in views, as shown in Figure 19 on page 34.
LNM Network Objects

MultiSystem Manager creates the following aggregate objects that can be customized:

Networks aggregate object
- Represents a group of LNM networks and has a resource-type of LAN network.

Network object
- Represents a single LNM network (an LNM and all its monitored resources) and has a resource-type of Local area network.

Using the Default Objects

MultiSystem Manager, by default, groups all LNM networks together under a single network aggregate object, LNM_Networks, shown in Figure 15 on page 30.

By default, MultiSystem Manager creates a network object, spname_LNM_Network, for each LNM network. The spname string is the value coded on the SP parameter of the GETTOPO statement for that LNM network.

Figure 19. TN3270 (IP Stack) objects

V1R3

RALVMR Sysplex
Internet Subnet

MDL

TVT2015 Internet Host

AIX44AC Internet Router

NMPIPL14.raleigh.ibm.Com Internet Host

MDL

9.67.42.331

V1R4 and V5R1

RALVMR Sysplex
Internet Subnet

MDL

TVT2015 Internet Host

AIX44AC Internet Router

NMPIPL14.raleigh.ibm.Com Internet Host

MDL

9.67.42.331

Interface

IP Stack Agg

NMP1PL14 TN3270 Server

148.47.122.96 TN3270 Client
9.67.33.201 TN3270 Client
9.67.33.205 TN3270 Client
If you do not want to use the default objects created by MultiSystem Manager, define the NETWORK_NAME or the NETWORK_AG_OBJECT parameters in the GETTOPO statements in the initialization file. See Figure 18 on page 33 for an example of the default network object.

**NetFinity Network Objects**

MultiSystem Manager creates the following aggregate objects that can be customized:

**Networks aggregate object**
Represents a group of NetFinity networks and has a resource type of *NetFinity networks*.

**Network object**
Represents a single NetFinity network and has a resource type of *NetFinity network aggregate*.

**Using the Default Objects**
MultiSystem Manager, by default, groups all NetFinity networks together under a single network aggregate object, *NetFinity_Networks*, as shown in Figure 15 on page 30.

By default, MultiSystem Manager creates a network object, *spname_NetFinity_Network*, for each NetFinity network. The *spname* string is the value coded on the SP parameter on the GETTOPO statement for that network.

If you do not want to use the default objects created by MultiSystem Manager, define the NETWORK_NAME or the NETWORK_AG_OBJECT parameters in the GETTOPO statements in the initialization file. See Figure 18 on page 33 for an example of the default network object.

**Open Topology Interface Network Objects**

MultiSystem Manager creates the following aggregate objects that can be customized:

**Networks aggregate object**
Represents a group of Open networks and has a resource-type of *Group*.

**Network object**
Represents a single Open network (an Open topology agent and all its managed resources) and has a resource-type of *Network aggregate*.

**Using the Default Objects**
MultiSystem Manager creates a networks aggregate object, *applname_Networks*, for each topology agent. The *applname* string is the value of the APPL parameter on the associated GETTOPO statement. MultiSystem Manager groups all of the resources in this network under this object.

By default, MultiSystem Manager creates a network object, *spname_applname_Network*, for each Open network. The *spname* string is either the PU or LU portion of SNA network address specified on the SP parameter on the associated GETTOPO statement. The name *applname* can either be the name specified on the APPL parameter or, if the ELEMENT parameter is specified, the name is the concatenated values of the APPL and ELEMENT parameters (*appl.element*).

See Figure 18 on page 33 for an examples of the default network object.
If you do not want to use the default objects created by MultiSystem Manager, define the NETWORK_NAME or the NETWORK_AG_OBJECT parameters in the GETTOPO statements in the initialization file.

**Tivoli Management Region Network Objects**

MultiSystem Manager creates the following aggregate objects that can be customized:

- **Networks aggregate object**
  - Represents a group of Tivoli management region networks and has a resource-type of TME 10 networks.

- **Network object**
  - Represents a single Tivoli management region network and has a resource-type of TME 10 network.

**Using the Default Network Objects**

MultiSystem Manager groups all your Tivoli management region networks into one network aggregate object called *Tivoli_Networks Group*, by default.

By default, MultiSystem Manager creates a network object, `spname_TME10_Network`, for each Tivoli management region network. The `spname` string is the IP host name coded on the SP parameter of the GETTOPO statement for that network. See Figure 18 on page 33 for an example of the default network object.

If you do not want to use the default network object created by MultiSystem Manager, define the NETWORK_NAME or the NETWORK_AG_OBJECT parameters in the GETTOPO statements in the initialization file.

---

**Creating Network Objects**

You can create network objects as described in the following sections:

- "Changing the Name of the Networks Aggregate"
- "Changing the Name of a Network Aggregate" on page 37
- "Displaying Individual Feature Networks" on page 37

---

**Changing the Name of the Networks Aggregate**

By default, MultiSystem Manager names the networks object using the feature name as a prefix, as shown in Figure 15 on page 30.

To change the name of the default networks aggregate object, use the NETWORK_AG_OBJECT parameter. For example, to define all first-shift networks under an aggregate object that is displayed only in your first-shift view, and similarly your off-shift networks to be displayed only in your off-shift view, define your first-shift aggregate object with a different name than that of your off-shift aggregate object.

To manage networks on both shifts that are displayed in your first-shift and off-shift views, define them under the same network aggregate object, and define that network aggregate object to be displayed in each view. To do this, define at least one of the networks under the network aggregate object to be displayed in each view.

If you define a network setting NETWORK_AG_OBJECT to NONE (NETWORK_AG_OBJECT=NONE), the network is displayed in only one view: the view that is defined by the GETTOPO initialization file statement for that network.
Note that all networks shown under the same NETWORK_AG_OBJECT object must be the same type of network such as Tivoli management region, NetFinity and Internet Protocol (IP). For example, if you issue the following GETTOPO commands from your initialization file, you create two different MAINSITE objects:

```
GETTOPO TMERES,NETWORK_AG_OBJECT=MAINSITE,......
GETTOPO LNMRES,NETWORK_AG_OBJECT=MAINSITE,......
```

Only Tivoli management region objects can be placed under the MAINSITE object created by the GETTOPO TMERES command, and only LAN Network Manager (LNM) objects can be placed under the MAINSITE object created by the GETTOPO LNMRES command.

**Changing the Name of a Network Aggregate**

By default, MultiSystem Manager names your network aggregate `spname._Feature_Network`, The feature string represents of the network, such as NetFinity, IP, or Tivoli management region. To change the name of a network aggregate, use the NETWORK_NAME parameter.

**Displaying Individual Feature Networks**

By default, MultiSystem Manager groups all of your networks in a single aggregate object, `Feature_Networks`. The Feature string represents the type of network, such as NetFinity, IP or Tivoli management region. To display a single network on a network view, do not aggregate it with other networks. Use the NETWORK_AG_OBJECT parameter and specify NONE.

See ‘Network Configuration Diagrams’ for additional examples of how network objects can be displayed.

**Network Configuration Diagrams**

This section shows a series of network configurations, sample initialization files, NetView management console windows, MultiSystem Manager network views, and network objects that are displayed in those views.

**Working with Networks**

The enterprise in Figure 20 consists of a Tivoli management region network and a NetFinity for Windows NT network.
In your initialization file, specify topology and status for both networks and their managed resources (TMRRES and NFRES). Also, specify the default network view and the default network objects to be created.

Figure 21 shows the resulting NetView management console window for the Configuration Diagrams shown in Figure 20 on page 37.

When you select MultiSysView from the NetView management console window, you see two cluster objects, TME10_Networks and NetFinity_Networks, shown in Figure 22 on page 39.
If you select **TME10_Networks**, and then select **More Detail**, the objects representing Tivoli management region Networks are displayed.

**A Network Aggregate**

Now your enterprise consists of the same two networks shown in Figure 20 on page 37 but, for example, you want your NetFinity for Windows NT network to be grouped under a NetFinity network aggregate named **Backup_Network**.

In your initialization file, code the Tivoli management region network as before, but for the NetFinity for Windows NT network, code **NETWORK_AG_OBJECT=Backup_Network**, as shown in Figure 23.
The resulting NetView management console window is identical to the window in Figure 21 on page 38 because a new network view was not created.

When you select MultiSysView from the NetView management console window, you see two cluster objects, TME10_Networks and Backup_Network, as shown in Figure 24.

If you select TME10_Networks and select More Detail, you only see objects representing the Tivoli management region network. If you select Backup_Network and select More detail, you only see objects representing the NetFinity for Windows NT network.

A New View and a Network Aggregate

Your enterprise now consists a Tivoli management region network, a NetFinity for Windows NT network, and an LNM network. You could then have the LNM network to appear in its own view called Test_View and be grouped under an aggregate object named Test_Networks.

To do this, in your initialization file, code the Tivoli management region and NetFinity for Windows NT networks as before, but for LNM, code NETWORK_AG_OBJECT=Test_Networks and NETWORK_VIEW=TEST_VIEW/TEST LNM NETWORKS, as shown in Figure 25 on page 41.
The NetView management console window displays two network views. A network view named Test_View has been added to the list of views, shown in Figure 26.

Selecting MultiSysView from the NetView management console window shows two cluster objects, TME10_Networks and Backup_Network, as before. These are shown in Figure 24 on page 40.

If you select Test_View, one cluster object named Test_Networks is displayed. If you select Test_Networks and select More Detail, only objects representing the LNM network are displayed.

Figure 25. Network Configuration and Initialization Files

The NetView management console window displays two network views. A network view named Test_View has been added to the list of views, shown in Figure 26.

Figure 26. NetView management console Window with Test_View added

Selecting MultiSysView from the NetView management console window shows two cluster objects, TME10_Networks and Backup_Network, as before. These are shown in Figure 24 on page 40.

If you select Test_View, one cluster object named Test_Networks is displayed. If you select Test_Networks and select More Detail, only objects representing the LNM network are displayed.
An Individual Network
Your enterprise now consists of a Tivoli management region network, a NetFinity for Windows NT network, an LNM network, and an Open network. You could then display the Open network in the same view (MultiSysView) as both the Tivoli management region and NetFinity for Windows NT network, but display it as an individual network and not under an aggregate object. This results in one less level of hierarchy while navigating through the NetView management console views.

In your initialization file, define the Tivoli management region, NetFinity for Windows NT, and LNM networks the same, but for the Open Network, define NETWORK_AG_OBJECT=None, as shown in Figure 27.

The resulting NetView management console window is identical to the window in Figure 26 on page 41 because no new network views were created. You added only the Open network to the default view MultiSysView.

When you select MultiSysView from the NetView management console window, you see four objects, two of which are connected, as shown in Figure 28. Networks TME10_Networks and Backup_Network are displayed as before, but the Open network is displayed in the view. The Open network consists of a real object named LUO_APPLA_Mgr, which represents the Open topology agent, and an aggregate object named LUO_APPLA_Network, which represents the Open network.

LUO_APPLA_NETWORK is the default name created by adding the Open network LU name (LUO) to _Open_Network.
Naming an Individual Network

Your enterprise now consists of Tivoli management region, NetFinity for Windows NT, LNM, and Open networks. You could display the IP network in the same view as the LNM (Test_View) but as an individual network that is not under an aggregate object. You could name the IP network TESTNET.

To do this, in the initialization file, define the previous networks as before, but for the IP network, define the NETWORK_VIEW keyword to be TEST_VIEW/TEST IP NETWORKS and the NETWORK_NAME as TESTNET, as shown in Figure 29.

The resulting NetView management console window is identical to the window in Figure 26 on page 41 because no new network views were created. Only the IP network was added to the view Test_View.

Selecting Test_View from the NetView management console window displays the view with the IP network added, as shown in Figure 30. IP is similar to the Open network, except that the aggregate object representing Network E’s network is named TESTNET.
Chapter 3. Operation Overview

This chapter explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage your networks.

Getting Started

The following is a list of items to check before using MultiSystem Manager. You need to ensure that:

- You have completed the Tivoli NetView for z/OS installation steps listed in the Tivoli NetView for z/OS Installation: Configuring Graphical Components.
- You have created or modified the MultiSystem Manager initialization statements as described in "Chapter 2. Customizing the Initialization Statements" on page 19.
- RODM is active and the MultiSystem Manager data model is loaded.
- GMFHS and the NetView management console are active and operational for displaying graphic views.
- The NetView RUNCMD task (DSIGDS) is active and operational to support RUNCMDs if you are using SNA sessions. TCP/IP must be operational if you are using it.
- The NetView SAVE/Restore task (DSISVRT) is active and operational to support the GETTOPO heartbeat function.
- Autotask AUTOMSM is active.
- NetView hardware monitor is active.
- NetView automation table is active.
- The event receive (IHSAEVNT) is active and configured properly. This is required only by the Tivoli management region feature.

Initializing Network Topology and Status

Network topology and status information must be stored in RODM before MultiSystem Manager can manage your networks. This is done during MultiSystem Manager initialization, which is initiated by issuing the INITTOPO command.

When the INITTOPO command is issued, MultiSystem Manager reads your initialization file, which contains the names and SNA network addresses or IP host names of the topology agents. MultiSystem Manager sends topology requests to each topology agent defined in the initialization file. Topology and status are returned to MultiSystem Manager and stored in RODM. When the initialization process is complete, you can view the networks from your NetView management console workstations. Topology and status are dynamically updated for the managed resources in your network.

Initialization Steps

After the INITTOPO command is issued, the initialization process proceeds in three steps:

1. Verifies the Multisystem Manager statements coded in the CNMSTYLE file.
2. Creates a topology manager object in RODM for each MultiSystem Manager feature. These objects represent the MultiSystem Manager topology features and are displayed on the Graphic Monitor—Details window.

   After the CNMSTYLE statements have been verified and the topology manager class objects have been created successfully, you receive the following message:

   FLC059I MULTISYSTEM MANAGER INITIALIZATION FILE filename

   HAS BEEN READ SUCCESSFULLY. THE MULTISYSTEM MANAGER IS NOW ENABLED.

   The status of MultiSystem Manager is set to ENABLED, meaning that subsequent GETTOPO commands can now be processed.

3. Executes the GETTOPO statements in the initialization file. When MultiSystem Manager starts processing these statements, you receive the following message:

   FLC048I GETTOPO COMMANDS FROM MULTISYSTEM MANAGER INITIALIZATION

   FILE filename ARE NOW BEING PROCESSED.

   Once MultiSystem Manager has finished processing all of the GETTOPO statements, you receive the following message:

   FLC126I GETTOPO COMMANDS FROM MULTISYSTEM MANAGER INITIALIZATION

   FILE file_name HAVE NOW BEEN PROCESSED.

**Issuing the Focal Point Commands**

To enable MultiSystem Manager to communicate with any Open topology agents, you must issue focal point change commands to each of the service points on which Open topology agents are executing. Open topology agents are typically written by partners or customers to manage resources not managed by MultiSystem Manager agents. Issue the following commands to each of the service points:

   FOCALPT CHANGE TARGET=uname FPCAT=SPCS
   FOCALPT CHANGE TARGET=uname FPCAT=ALERT

**Issuing the INITTOPO Command**

The INITTOPO command can be issued from a NetView command line or command procedure at any time. This initializes topology and status processing. Topology agents notify MultiSystem Manager of subsequent topology and status changes by means of alerts and resolutions. MultiSystem Manager processes these alerts and resolutions using the NetView automation table.

To initialize MultiSystem Manager from your host NetView operator station task (OST), enter the INITTOPO command on the operator command line:

   INITTOPO filename

   The filename parameter is optional. If you do not specify a file name, MultiSystem Manager uses the default initialization file, FLCAINP.

   You can also use the following statements that are included in CNMSTYLE:

   - (MSM)AUTOTASK.?MSMdefault.Console = *NONE*
   - (MSM)AUTOTASK.?MSMdefault.InitCmd = INITTOPO

   The first statement starts the autotask defined in the MSMdefault statement, which is also in CNMSTYLE. The second statement runs the INITTOPO command on that autotask. Comment these statements if you do not want to run the INITTOPO command during NetView initialization.

   The INITTOPO command can also be issued from a NetView command procedure. Refer to the INITTOPO command in the Tivoli NetView for z/OS Command Reference or online help, for information on specifying an initialization file on the INITTOPO
command. See "Chapter 2. Customizing the Initialization Statements” on page 19 for more information about initialization file statements.

If you have just installed a MultiSystem Manager feature, issue the INITTOPO command to ensure that the global variables have been set correctly. This also ensures that the correct topology manager objects are created in RODM. See "Appendix C. Global Variables” on page 127 for more information about global variables.

Processing Topology Requests

One of the major functions of MultiSystem Manager is to process requests for topology and status of resources in your network. These requests are made by issuing GETTOPO commands for particular resources being managed by a topology agent. GETTOPO commands can be coded in the MultiSystem Manager initialization file or issued from a NetView command line or command procedure. GETTOPO commands can be processed by MultiSystem Manager only if MultiSystem Manager is enabled.

MultiSystem Manager receives the GETTOPO request, parses it, and builds and executes a series of RUNCMDs depending on the topology request. These RUNCMDs are executed from the specified autotask and resource topology and status is returned to MultiSystem Manager in the form of RUNCMD responses. These responses are then parsed and the topology and status information is stored in RODM.

In large, complex computing environments, you sometimes stop and restart multiple programs and processes to apply maintenance, backup data, upgrade, and so on. During such times, you might want to suspend the processing of topology requests, knowing that these requests fail until a particular process or program is restarted. MultiSystem Manager provides commands that enable you to suspend and resume MultiSystem Manager processing of topology requests.

Suspending Processing of Topology Requests

The SUSPTOPO command suspends MultiSystem Manager processing. The status of MultiSystem Manager must be enabled to suspend processing. After successful completion of this command, the status of MultiSystem Manager is set to SUSPENDED. This also causes the topology manager objects displayed on the Graphic Monitor—Details window to change to a status of Not available. GETTOPO commands cannot be processed while MultiSystem Manager is suspended.

GETTOPO commands issued before MultiSystem Manager processing was suspended, continue to process until complete. GETTOPO commands issued after MultiSystem Manager processing is suspended are ignored, and message FLC045E (processing is suspended) is issued.

Many of the alerts sent from the topology agents report new resources coming online. These alerts drive statements in the NetView automation table, which in turn, drive command procedures that issue topology and status requests for the newly discovered resources. If MultiSystem Manager processing is suspended when these topology and status requests are issued, these requests are ignored. Consequently, topology and status information for new resources is lost while MultiSystem Manager processing is suspended.

Resuming Processing of Topology Requests

If the status of MultiSystem Manager is suspended, issue the RESTOPO command to resume processing topology and status requests. After the RESTOPO command
successfully completes, MultiSystem Manager status is set to ENABLED. This also causes the topology manager objects displayed on the Graphic Monitor—Details window to change to a status of AVAILABLE. All GETTOPO commands issued while MultiSystem Manager processing is enabled are processed.

Depending on how long MultiSystem Manager processing has been suspended and depending on the size of your network and the amount of activity in your network, you might need to issue GETTOPO commands for specific networks to bring topology and status information up to date. If you want to update the topology and status information for all your networks, issue the INITTOPO command again.

**Displaying Topology Processing Information**

The DISPTOPO command is used to display the status and initialization parameters for MultiSystem Manager, as well as some general information about this component of Tivoli NetView for z/OS. Figure 31 shows the resulting display windows.

See the DISPTOPO command in the Tivoli NetView for z/OS Command Reference or the online help for a description of the displayed fields.

![Figure 31. DISPTOPO Status Window Example](image)

**MultiSystem Manager Views**

After you initialize network topology and status in RODM, your NetView management console provides views of your MultiSystem Manager networks.

Using the NetView management console pull-down menus, you can navigate among the views to isolate failures and send commands to resolve problems. Refer to chapters 4 through 10 for navigation information about each MultiSystem Manager feature. The following view navigation information is common to all MultiSystem Manager features.
Navigating Views

Common view navigation options are available across all MultiSystem Manager features. From an aggregate object, the More Detail and Configuration->Child navigations display resources that are contained in the aggregate object. Resource->More Detail will show just the immediate child resources. Resource->Configuration Child shows the immediate child resources and any lower-level (grandchild) resources. The Configuration->Parent navigation is also supported. This navigation option will show the selected object connected to all aggregate object parents of that object.

To identify where a real network resource is managed by more than one topology agent, operators can select Configuration->Parent (on the context menu of the network resource object). This enables operators to display a view that contains the parent of the resource or the path from a real resource to the host that owns it. This is useful for navigating between managed networks. It is also useful in identifying cases where your systems administrator has defined two agents to manage the same network resource.

Operators can also select Configuration->Logical and Configuration- >Physical (on the context menu of the network resource object) to view the logical and physical relationship between resources. However, dependent upon the configuration, these navigations may only re-display resources in the current view.

As you navigate through your views, you can select any object and get resource information on that object. Each Resource Information window has the following fields:

**Resource name**
Name of the selected object.

**Other data**
Detailed information unique to the selected object.

**Type**
Class of view object representing the resource.

**Customer data**
Reserved for customer use.

**Managed by**
Name of the MultiSystem Manager topology manager that manages this object.

**System status**
Status of the selected object and time status was last updated.

**Operator status**
Status initiated by operator, for example, if aggregation is suspended.

**Aggregation priority (real objects only)**
The number of exception status levels to be reported.

**Total resources (aggregate objects only)**
The number of resources contained in this aggregate.

**Unsatisfactory/Exception resources (aggregate objects only)**
The number of resources contained in this aggregate that have either unsatisfactory or exception status.

**Degraded (aggregate objects only)**
The number of resources that must report exception status before the status of this aggregate is changed to degraded.
Severely degraded (aggregate objects only)
The number of resources that must report exception status before the status of this aggregate is changed to severely degraded.

Unsatisfactory (aggregate objects only)
The number of resources that must report exception status before the status of this aggregate is changed to unsatisfactory.

Topology Correlation Across Different Types of Networks
Tivoli NetView for z/OS uses topology correlation to automatically tie together resources managed by different types of topology features such as IP and Tivoli management region. Topology correlation is provided for all MultiSystem Manager topology features, the Tivoli NetView for z/OS SNA Topology Manager, and customer or vendor applications that use the GMFHS data model.

Most managed resources participate in more than one network or system. For example, a workstation can serve as a station in a token ring LAN network and a Windows® 95 platform, and be present in the IP environment. Most management agents only recognize one type of network or system, as opposed to every resource running in the system. The IP agent, for example, only recognizes resources with an IP addresses. Therefore, an IP view of the example workstation would contain information related to IP, but not the token ring adapter and Windows 95 platform.

Topology correlation overcomes the agent-specific perspective of management agents and enables a graphic monitor operator to view all components of a network device. This single view includes the current status of all components.

Topology correlation is accomplished using a network address (LAN MAC address or TCP/IP address) or a customer-defined free-form string value (for example, Accounting or London).

Network topology correlation automatically creates or updates correlated aggregate objects that may contain the following resource objects, if the corresponding topology feature is running for that type of resource:
• IP Interfaces
• LNM Token Ring Adapters
• Vendor or customer created Open Node and Node aggregate resources
• NetFinity Operating System resources
• Tivoli management region Monitor resources
• SNATM PU resources
• GMFHS managed real resources
• TN3270 server and client resources

If a topology feature discovers both a LAN MAC address and a TCP/IP address for a managed resource, topology correlation typically creates just one correlated aggregate object and connects the resource to it.

You can include any resource in topology correlation by setting a free-form correlator value. This is useful when a resource is not automatically correlated, because no MAC address or IP address is discovered for it. It is also useful when you want to include managed objects in views that you define based upon organizational structure or geography. For example, if a customer or partner application specifies London for the field value on different managed resource objects in RODM, all of those objects would be contained in one aggregate object.
with a display name of London. This object would have a display type of Open System Aggregate. For an example, see Figure 35 on page 53.

You can also use support for multiple values, in the free-form Correlator field, to create or locate a hierarchy of correlated aggregate objects that match your organizational structure. Figure 33 is an example.

![Figure 32. Correlated Aggregate Objects Example](image)

Topology correlation enables you to isolate a failure that one managed system caused in a separately managed system. For example, operators can see that a LAN adapter failure caused a workstation managed by NetFinity to fail and solve the problem in the LAN adapter, thus preventing wasted time investigating workstation software status.

Topology correlation is an optional function that discovers correlations dynamically through RODM methods. To enable this function, your system administrator must include the FLCSDM8 file in the RODM structure load. Including SNA resources in correlation also requires a change to the FLBSYSD file. Refer to the Topology Correlation chapter of the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide or the FLBSYSD file.

You can override correlation default display characteristics and you can include additional RODM objects in network or free-form topology correlation. All customization can be done in the FLCSDM8 RODM load file. Refer to the Topology Correlation chapter of the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide for customization information.

**Viewing Correlated Resources**

To view correlated resources, select a Real resource object, then select **Configuration->Parent**. The resulting NetView management console view, Figure 33 on page 53, shows both the navigation hierarchy for the topology feature of that object and a branch to the correlated aggregate in which the object is contained.
Figure 33 shows how a SNA PU is part of a SNATM network, but is also contained in a workstation aggregate that is part of a NetFinity Network.

In your views, correlated aggregate objects created or located by network address have an object type of LAN Workstation aggregate or IP System aggregate. If a correlated aggregate object is created or located by free-form correlation, the object is given an object type of Open System Aggregate. To see the resources contained in a correlated aggregate object, select the aggregate object and then select More Detail. The status of the correlated aggregate object reflects the combined status of the resources contained within.

For example, the NetView management console view shown in Figure 34 shows all the resources contained in workstation NetFinFVT3.
The workstation (NetFinFVT3) contains resources discovered by LNM, SNA Topology Manager, and NetFinity agents.

If you select a correlated aggregate object, you can display the contained resources and the aggregate by selecting **Configuration Child**. The NetView management console view shown in Figure 35 is the result.

![Figure 35. Correlated Aggregate and Resources Example](image)

**Workstation Aggregate Objects Created by MultiSystem Manager Agents**

The MultiSystem Manager NetFinity and Tivoli management region features can manage workstations and selected resources contained in those workstations. Other MultiSystem Manager features manage network resources, some of which might be contained in a workstation. Because of these differing objectives, the NetFinity and Tivoli management region features create correlated aggregates and display them in the More-Detail NetView management console navigation hierarchy (inline); the other features do not. For the other features, if topology correlation has been loaded, you can access correlated aggregate by selecting **Configuration->Parent** from the context menu of a real resource.

Correlated aggregate objects gather and display more information about their contained resources as additional correlations are achieved. Figure 35 displays Resource Information Other Data discovered by the Open system aggregate from each of the resources it contains. At the NetView management console, Other Data is displayed in the Data1 field of the Resource Properties dialog.

**Note:** The maximum length of the Other Data field, that can be displayed, is 254 characters. If the string in the Other Data field is longer than 254 characters, the Other Data field is truncated when displayed.

Figure 35 also demonstrates how a customer can use free-form topology correlation to relate objects discovered by NetView to business entities, such as a location (for example, London or Building201) or a discipline (for example, Manufacturing). Free-form correlation also enables you to categorize objects you create in RODM, as shown for objects of the GMFHS and Open Data Models.
Customization Options
Figure 32 on page 51 illustrates the result of free-form correlation of a multi-valued string. This string can be used to locate or create a hierarchy of correlated aggregate objects that match an organizational structure. Setting the Correlator field value on three real RODM objects created and linked the eight correlated aggregate objects shown in Figure 32 on page 51. The Correlator field values set on the three real RODM objects follow:

Set Bridge1 Correlator = 'RmA206 Bldg300 Barcelona Europe'
Set Switch5 Correlator = 'RmD312 Bldg400 Barcelona'
Set PBX3 Correlator = 'OpCenter Europe World'

Topology correlation defaults to displaying each correlated aggregate by its workstation name (if available). This is the same name that the MultiSystem Manager Netfinity and Tivoli management region features use to name workstations. If a workstation name is not available, the next available name type on the Display Name Priority list is selected. However, you can select a different name type to display for all correlated aggregates by customizing the RODM Loader file FLCSDM8. Refer to the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer's Guide for the available customization options. Refer to the customization instructions in the FLCSDM8 file for customization instructions. Table 7 shows the default display name priority.

Table 7. Correlated Aggregate Object Default Display Name Priority

<table>
<thead>
<tr>
<th>Priority</th>
<th>Name Type</th>
<th>Discovered by</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Computer Name</td>
<td>Netfinity and Tivoli management region features</td>
</tr>
<tr>
<td>2</td>
<td>IP Host Name</td>
<td>IP, Tivoli management region, and Netfinity features</td>
</tr>
<tr>
<td>3</td>
<td>TCP/IP Address</td>
<td>IP and Tivoli management region features</td>
</tr>
<tr>
<td>4</td>
<td>SNA Node Name</td>
<td>SNATM</td>
</tr>
<tr>
<td>5</td>
<td>LAN MAC Address</td>
<td>LNM, SNATM, IP, and Netfinity features</td>
</tr>
</tbody>
</table>

Note: Altering the display name priority in file FLCSDM8 might cause a discrepancy between the name types displayed in the NetView management console, and the name types displayed in example views in this book for Netfinity and Tivoli management region.

If a resource is correlated to an aggregate based upon a free-form value, the display name for the aggregate will be the free-form string value.

The topology correlation method also enhances your ability to locate a correlated aggregate using the NetView management console locate function. Operators can locate a correlated aggregate using a value for any of the following network addresses, dependent upon what is discovered in your network:

- LAN MAC Address
- IP Address
- SNA PU or LU
- IP Host name
- Free-form correlator string
Setting Free-Form Correlation Values
You can set free-form Correlator field values with point-and-click menus using Visual BLDVIEWS running on a workstation. You can also set Correlator field values, using RODMView, by creating a command list (CLIST) or by creating a BLDVIEWS script. For more information, refer to the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide.

Using Visual BLDVIEWS has several advantages. It enables you to point-and-click to select resources to be included in correlation. To do this, you do not need to understand RODM or BLDVIEWS syntax. Visual BLDVIEWS creates a valid BLDVIEWS script. BLDVIEWS typically includes objects in views if those objects have a consistent naming scheme (for example, CPNRTR2 and CPNHST14). Views are built, hierarchically, from the top down.

Multiple free-form correlation does not require objects to have a similar object naming. It builds views from the bottom up. Using BLDVIEWS and Topology correlation together, it is easier to build custom views to match your enterprise. To set a Correlator field value on an object in Visual BLDVIEWS, select your target object in the left side of the Visual BLDVIEWS menu and drag it to the Build/Set Resources area on the right side of the menu. This is illustrated in Figure 36.

![Figure 36. Setting a Correlator Field Value](image)

After you drop the object into the Build/Set Resources area, double-click on the object to bring up the settings dialog for it. The settings dialog is shown in Figure 37 on page 56.
Fill in the Correlator field. You can set a single string value or a multi-valued string, delimited by blank spaces or commas. After you have set correlator values on all the managed objects that you want to include in free-form correlation, save your requests to the host and run them, as shown in Figure 36 on page 55. Specify a file name and location to save the BLDVIEWS script that will be generated. If RODM is recycled, you can restore your custom correlations by issuing this script from Visual BLDVIEWS or from BLDVIEWS on z/OS. If you want to modify the script, you can edit it from Visual BLDVIEWS or z/OS/TSO.
Issuing Commands

MultiSystem Manager provides an easy-to-use command interface based on the NetView command support that enables you to select objects from your NetView management console views and issue commands to these objects without requiring you to enter the object’s name or address.

Generic commands (such as Current Status, Activate, Inactivate and Recycle) are not implemented by MultiSystem Manager. They can be implemented using the BLDVIEWS utility. For more information about BLDVIEWS, see Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide.

Example of Issuing a Command in the NetView Management Console

To issue a command to the IP system shown in Figure 42 on page 68:

1. Right-click on the system and select IP resource specific commands.
2. Select the command you want to issue.

Note: Responses to commands you issued using the command support are returned to the Command Responses window available from your NetView management console view under the Options pull-down menu. To display the window, select Show Log from the Options pull-down menu. To display the entire log, select Tear Away Log from the Options pull-down menu.

For more information about refreshing, copying, or getting help for messages that are displayed in the NetView management console Command Responses window, refer to the Tivoli NetView for z/OS NetView Management Console User’s Guide or the online help.

Figure 38 on page 58 shows the expanded commands as you select the command on NetView management console.
Resolving Network Problems

To resolve network problems, first locate the object in your view that is having the problem, as discussed in “Finding a Failing Object”. Then understand the meaning of the status of the object as discussed in “Understanding View Object Status” on page 59 to diagnose the problem and take the appropriate action.

Finding a Failing Object

Suppose you are monitoring a MultiSystem Manager networks view, and you notice the network aggregate turns yellow, indicating a problem. To find the problem, you can do one of the following:

- Select the yellow aggregate object and then select More Detail and repeat this procedure until you find the failing resource.
- Select an object and then select Locate Failing Resource. This function automatically builds a detailed view showing all real objects under the aggregate object that have unknown or unsatisfactory status.

Refer to the Tivoli NetView for z/OS NetView Management Console User’s Guide for information about locating failing resources in your views.

Finding the Status of an Object

To view alert information on the NetView management console, right-click on the object and select Event Viewer. The event viewer window shows the alerts received for that object.
Understanding View Object Status

The status for both aggregate and real objects is displayed. The status of a real object reflects the operational capability in the network for that object. Table 8 shows a general description of real object status.

Table 8. Real Objects Status Summary

<table>
<thead>
<tr>
<th>Status</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satisfactory</td>
<td>Resource is fully operational.</td>
</tr>
<tr>
<td>Intermediate</td>
<td>Performance of resource is degraded.</td>
</tr>
<tr>
<td>Unsatisfactory</td>
<td>Resource is not operational.</td>
</tr>
<tr>
<td>Unknown</td>
<td>MultiSystem Manager cannot communicate with the resource, and therefore, does not know the current status of the resource.</td>
</tr>
</tbody>
</table>

Setting Status Aggregation Thresholds

The status for an aggregate object is derived or aggregated from the real resource objects below it. The aggregation algorithm is based on the degraded, severely degraded and unsatisfactory thresholds for each of the aggregate objects.

Setting aggregation is important. If the thresholds are set too high, failures are not seen on the high-level views that you are monitoring. If the thresholds are set too low, minor failures cause the status of the aggregate objects to have unsatisfactory status, which can hide major failures.

MultiSystem Manager sets default thresholds for each object type. You can use the defaults, or you can modify these thresholds by using NetView management console workstation menus or by changing the value of the threshold in RODM. Refer to the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide for information about the GMFHS aggregation algorithm for aggregate objects.

Note: You can also set aggregation thresholds by using the view customization utility, BLDVIEWS. For information on BLDVIEWS, see Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide.

Setting Aggregation Priorities

Aggregation priorities reflect the relative importance of a real object in your network. MultiSystem Manager sets a default aggregation priority for each type of real object in your view. You can use the defaults, or you can modify these aggregation priorities. To modify the aggregation priorities, use NetView management console workstation menus or change the value of the aggregation priority value in RODM. Refer to the Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide for information about the GMFHS aggregation algorithm for real objects.

Note: You can also set aggregation priorities by using the view customization utility, BLDVIEWS. For information on BLDVIEWS, see Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer’s Guide.
Updating Topology and Status

After the initial topology and status are stored in RODM, you can extend the set of managed resources by gathering topology and status for resources that were not previously managed. Use the GETTOPO command to get topology and status for specific resources or groups of resources. For example, if you are monitoring only the MultiSystem Manager topology agent, you can issue a GETTOPO IPRES (IP specifies the network) command to gather information about the resources managed by that agent.

Removing MultiSystem Manager Objects from Views

Under usual network operations, you might have objects in your views that you do not need to monitor. Perhaps you want to remove objects from a view because their status has been UNKNOWN or UNSATISFACTORY for an extended period of time. Any object that has had UNKNOWN or UNSATISFACTORY status for an extended period of time probably is no longer connected to your network, and can be removed from your views. You can remove objects from your views by one of the following methods:

- Issuing a command from the NetView operator command line
- Scheduling a user-written command procedure to run as part of your automated network operations

Removing Objects Meeting Criteria

MultiSystem Manager provides a command, REMVOBJS, to remove objects from your views if the objects meet certain criteria. This command, the REMVOBJS command, can best be thought of as a cleanup routine. You can issue the REMVOBJS command against an aggregate object or a real object.

When objects, both real and aggregate, are defined in RODM a Purge attribute is created for each object. This attribute defines whether or not the object can be removed from your views. The Purge attribute can be set to a value of zero, one, or two, as shown in [Table 9](#).

<table>
<thead>
<tr>
<th>Purge Attribute Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Object can be removed</td>
</tr>
<tr>
<td>1</td>
<td>Object cannot be removed but certain links can be removed</td>
</tr>
<tr>
<td>2</td>
<td>Neither the object nor any of its links can be removed</td>
</tr>
</tbody>
</table>

An object is defined as *purgeable* if the Purge attribute is zero. When MultiSystem Manager creates a new object in RODM, the value of the Purge attribute for the object is set to zero. This enables the object to be removed. To prevent an object from being removed from your views, use the SETREMV command to change the value of the Purge attribute. For more information about the Purge attribute and using the SETREMV command, refer to the *Tivoli NetView for z/OS Command Reference* or online help.

**Note:** The display of special connectivity relationships, such as in an IBM token-ring, might be affected by the removal of objects from the view. If the
NetView management console does not display the view after the REMVOBJS command runs, you must rebuild the view by issuing the appropriate GETTOPO command.

Removing Real Objects

Each real object must satisfy the following two criteria before it can be removed from your views:

- The object must be outdated, that is, it must have had UNKNOWN or UNSATISFACTORY status for a user-specified period of time.
  - When you issue the REMVOBJS command, specify whether you want to remove objects in UNKNOWN or in UNSATISFACTORY status.
  - When the command is issued, specify how long the objects to be removed should be in an UNKNOWN or UNSATISFACTORY status. If you do not specify a time period, the default is to remove objects after 1 day (24 hours).
- The object must be purgeable; its Purge attribute is zero.

If the real object is both outdated and purgeable, REMVOBJS removes the real object from your views. If neither, or only one of the criteria are met, the real object is not removed.

Note: The REMVOBJS command has completed successfully when all selected objects that are outdated and purgeable are removed. If no selected objects meet this criteria, the command still completes successfully, though no objects were removed.

Removing Aggregate Objects

Each aggregate object must satisfy the following two criteria before it can be removed from your views:

- The aggregate object must be defined in RODM as being purgeable; its Purge attribute is zero.
- All the real objects that comprise the aggregate object must also meet the criteria to be removed (outdated and purgeable).
  - If one or more of the real objects that comprise the aggregate object do not meet the criteria, the aggregate object is not removed.

When you issue a REMVOBJS command against an aggregate object, MultiSystem Manager first checks to determine if the aggregate object itself can be purged. If it can be purged, MultiSystem Manager searches through RODM to determine if any of the real objects that comprise the aggregate object are outdated and can be purged. These real objects are then removed from your view. If all the real objects are removed, then the aggregate object is removed.

Preventing an Object from Being Removed

MultiSystem Manager provides a command, SETREMV, to change the Purge attribute of any MultiSystem Manager-created object. By changing the Purge attribute, you can prevent objects from being removed from your views.

For more information about the SETREMV command, refer to the Tivoli NetView for z/OS command help or online help.
Using Online Help

MultiSystem Manager provides online help for commands and messages. This help is available on your NetView management console and your NetView operator station task (OST). If you cannot access help, verify that the installation procedures were followed correctly.

Using the NetView Management Console Online Help

Help is available for each command in the NetView management console command support.

- From the command support, you can obtain help on each of the following:
  - Using the command support
  - Using the command help
  - Issuing commands from the command support
  - Coding commands
- At the command dialog window displayed for topology commands that have variable parameters, you can obtain help for each parameter or the entire command, including a detailed syntax diagram. See Figure 39 on page 63 for an example NetView management console command dialog window.
- At the command dialog window, select Help and search for the command for which you need help. Context sensitive help is not available from this window.
- After right-clicking an object, select Help from the menu to obtain help on all commands.
Using NetView Help

Help for the topology commands and all messages is available from the NetView command line. Refer to the Tivoli NetView for z/OS Command Reference or online help for more information about the NetView HELP command.

Table 10. Getting help for commands and messages

<table>
<thead>
<tr>
<th>If you want help on...</th>
<th>Then...</th>
</tr>
</thead>
<tbody>
<tr>
<td>A specific command</td>
<td>Enter HELP command_name, where command_name is the name of the command such as GETTOPO IPRES.</td>
</tr>
<tr>
<td>A specific message</td>
<td>Enter HELP message_id, where message_id is the identifier of the message such as FLC000.</td>
</tr>
</tbody>
</table>

Help is available for the following topology commands:
DISPTOPO
Displays information about MultiSystem Manager.

DMCS
Sends a command to a network manager.

GETTOPO
Retrieves topology and status information from a topology agent.

GETTOPO OPENRES
Adds topology and status for a specified Open topology agent and its managed resources.

GETTOPO HOSTONLY
Retrieves and add status updates for only the specified Open topology agent.

INITTOPO
Initializes MultiSystem Manager.

REMVOBJS
Removes objects from your views.

RESTOPO
Resumes processing of MultiSystem Manager topology requests.

SUSPTOPO
Suspends processing of MultiSystem Manager topology requests.
Chapter 4. Internet Protocol Network Operation

This chapter explains how to initialize MultiSystem Manager, start and stop processing, and use the views and command support to manage your Internet Protocol (IP) networks.

MultiSystem Manager Views

After you initialize network topology and status in RODM, the NetView management console provides views of IP networks. Using the NetView management console pull-down menus, you can navigate among the views to isolate failures and send commands to resolve problems.

IP View Objects

The following IP objects are displayed in various IP views:

- Aggregate objects that represent:
  - A group of IP networks
  - A single IP network
  - A user-defined logical group of IP resources
  - Subnetworks
  - Segments
  - Routers
  - Bridges
  - Links between routers and subnetworks or segments
  - Hubs
  - Hosts

- Real objects that represent:
  - Topology agents
  - Interfaces

If you installed the TN3270 Manager, then the following IP objects are displayed in various IP views:

- Aggregate objects that represent:
  - TCP/IP Stack aggregates

- Real objects that represent:
  - Telnet 3270 servers
  - Telnet 3270 clients

Figure 40 on page 66 shows the various types of objects that are displayed in IP network views.
Finding Resources

To display resources, access the Locate Resource window. From this window, you can locate an adapter by its MAC address and other IP resources by their names. You can also locate a resource by its display name.

Navigating Network Views

This section illustrates what you see when you navigate through the views of an IP network. The following GETTOPO command, sent to a topology agent with an LU name of AIXI8911, was used to build the NetView management console views:

GETTOPO IPRES, SP=AIXI8911, APPL=FLCIP01

To navigate the NetView management console views, begin by selecting the default MultiSystem Manager network view, MultiSysView, from the business view, as shown in Figure 41 on page 67.
**Note:** This chapter assumes you are using the default views created by MultiSystem Manager. For more information on creating network views, see "Defining NetView Management Console Views" on page 28.

Figure 41 shows the NetView management console network view named MultiSysView.

![Figure 41. NetView management console Default Network View (MultiSysView) Example](image)

MultiSysView consists of an object (a NetView management console cluster aggregate object) that represents all the IP networks defined in the initialization file used to initialize MultiSystem Manager. This aggregate object is named IP_Networks and its resource type is Internet networks. From MultiSysView, you can get more detailed views that display your IP networks.

**Viewing IP Networks**

IP views consist of the following levels:

**IP Networks View**

A single view that shows all your IP networks

**IP Network Views**

Views showing the subnetworks and connecting routers, bridges, and hubs within a network

**IP Subnetwork Views**

Views showing the segments and connecting routers, bridges, and hubs within subnetworks

**Location Views**

Views showing collections of user-defined IP managed resources

**Segment Views**

Views showing the routers, bridges, hubs, and hosts within a segment

**Router Views**

Views showing the interfaces within a router

**Bridge Views**

Views showing the interfaces within a bridge
Hub Views
Views showing the interfaces within a hub

Host Views
Views showing the interfaces within a host

Link Views
Views showing the router interface connecting a router to a subnetwork or segment

IP Networks View
To create the IP networks view with the NetView management console, select the IP_Networks object displayed in Figure 41 on page 67 and select More Detail. The resulting view, as shown in Figure 42, shows the IP networks that comprise the aggregate object IP_Networks.

This NetView management console view shows a single IP network, AIX18911_IP_Network. MultiSystem Manager displays a separate IP network for each Tivoli NetView for UNIX (Tivoli NetView) or NetView for UNIX network defined in your initialization file. A separate IP network is shown for each GETTOPO statement in the initialization file.

Each IP network is represented by two connected symbols:
- a node aggregate object, an Internet network aggregate
- an application real object, an Internet manager

In Figure 42, the aggregate object, AIX18911_IP_Network, represents all the managed IP resources in the network. The real object, AIX18911_IP_Mgr, represents the MultiSystem Manager topology agent. The name of the agent is comprised of the SNA service point address and the type of network the agent is managing (Internet Protocol). To find additional information on the topology agent, select the Internet manager and select Resource Properties. Figure 43 on page 69 shows the resulting NetView management console Resource Properties window for the selected Internet manager AIX18911_IP_Mgr.
IP Network Views

To view the managed IP resources in the AIXI8911 network, for example, select the aggregate object as shown in Figure 42 on page 68, then select More Detail. The NetView management console creates a view showing the routers, bridges, and hubs of the IP subnetwork, and the links that connect them. This is similar to the view shown in Figure 44.

This partial view shows some of the objects that make up the AIXI8911 network. The view shows five subnetworks (internet subnet aggregates) that are connected by routers, hubs, and bridges. It also shows an internet location cluster object, which is described in Location Views on page 70. This view also shows links between the various subnetworks and the routers, hubs, and bridges. These links show the status of the router, hub, or bridge interface that is a part of that.
subnetwork. For example, the link (connection) between router aixsvt03 and subnetwork 9.67.160 in Figure 42 on page 68 reflects the status of the aixsvt03 interface on subnetwork 9.67.160. Similarly, the link (connection) between router aixsvt03 and subnetwork 192.5.0.64 reflects the status of the aixsvt03 interface on subnetwork 192.5.0.64.

**IP Subnetwork Views**
Each IP subnetwork consists of a number of segments and connecting links, routers, bridges, and hubs. To see the components of an IP subnetwork, select a subnetwork such as 9.67.128 in Figure 44 on page 69, and select More Detail. Figure 45 shows a NetView management console More Detail view of subnetwork 9.67.128.

Figure 45 also shows that the selected sub-network is made up of two segments (9.67.128.Segment1 and 9.67.128.Segment2) and five routers (ndr2216, artgw0mn0, elitertr, ewcenx, and gandalfatm). This view also shows link aggregates between Segment1 and the routers. These links represent the status of the router interfaces on Segment1. For more detail, select the link (connection) between a router and Segment1 and then select More Detail. You can then display resource information, that is displayed in the resulting view, on the interface object of this router.

**Location Views**
Each location object represents a number of user-defined IP managed objects that have been defined under one NetView management console view object. To see the components of a location select a location object, like those shown in Figure 44 on page 69, and select Resource->More detail.

**Segment Views**
Each IP segment consists of a number of IP routers, bridges, hubs, and hosts. To see a more detailed view of 9.67.128.Segment1 on Figure 45, select it and select More Detail. Figure 46 on page 71 shows a more detailed NetView management console view of 9.67.128.Segment1.
Router, Bridge, Hub, Host, and Link Views
Each router, bridge, hub, host, and link aggregate object consists of one or more interfaces. To see the interfaces, select the object and request a more detailed view. For example, to see the interfaces associated with a link, select a link and select More Detail. Figure 47 shows a more detailed view of link webaix shown in Figure 46.

Managing TN3270 Resources
The view in Figure 44 on page 69 shows an IP network. This network includes an Internet Host. Selecting Configuration->Child on the Internet Host object would open a view like that shown in Figure 48 on page 72. Selecting Configuration->Child on the IP Host object yields IP Stack and IP Interface objects, and beneath the IP Stack object, a TN3270 server and TN3270 Clients...
object. Note that this CS/390 internet host is not defined in a workstation, but it is defined in a System 390.

Figure 48. Configuration Child on IP Host yields IP Stack and IP Interface, and beneath the IP Stack, a TN3270 server and TN3270 Clients

In the view shown in Figure 48, there is an IP Stack Aggregate object. Selecting More Detail on the IP Stack Aggregate object from this view opens a view like the one shown in Figure 49 on page 73.
The view shown in Figure 49 contains the TN3270 servers and clients defined as critical resources. The TN3270 Management function enables monitoring of Internet Hosts, TN3270 servers, and critical TN3270 clients. To define the clients that are critical to your business, use the `IPSTACK` parameter in the FLCS3270 file. For more information on this parameter, see "(MSM)COMMON.FLC_TN3270_FILE" on page 118.

Figure 49. IP Stack aggregate object containing TN3270 server and clients

The view shown in Figure 49 contains the TN3270 servers and clients defined as critical resources. The TN3270 Management function enables monitoring of Internet Hosts, TN3270 servers, and critical TN3270 clients. To define the clients that are critical to your business, use the `IPSTACK` parameter in the FLCS3270 file. For more information on this parameter, see "(MSM)COMMON.FLC_TN3270_FILE" on page 118.
Chapter 5. LNM Network Operation

This chapter explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage your LNM networks.

MultiSystem Manager Views

After you initialize network topology and status in RODM, the NetView management console provides views of your LNM networks. Using the NetView management console pull-down menus, you can navigate among the views to isolate failures and send commands to resolve problems.

Note: In this chapter, the term hub is used interchangeably with the terms controlled access unit and CAU.

LNM View Objects

The following LNM objects are displayed in various LNM views:

- Aggregate objects that represent:
  - A group of LAN networks
  - A single LAN network
  - Components of a segment
  - Components of a hub

- Real objects that represent:
  - OS/2 LAN Network Managers (LNMs)
  - Segments
  - Station adapters
  - Controlled Access Units (CAUs)
  - Controlled Access Unit (CAU) adapters
  - Bridge adapters
  - Parallel Bridge
  - Remote Bridge
  - Local Bridges
  - Bridge links

Figure 50 on page 76 shows the various types of objects that can be displayed in your LNM network views.
Finding Resources
To find resources, access the Locate Resource window. Using the Locate Resource window, you can locate an adapter by its MAC address, a bridge by the bridge name, a hub by the hub name, and a segment by the segment name. You can also locate a resource by its display name.

Navigating Network Views
To navigate through the views, begin by selecting the default MultiSystem Manager network view, MultiSysView, from the NetView management console business view shown in Figure 51 on page 77.

Figure 50. Types of LNM View Objects

Figure 51 on page 77 also shows the NetView management console network view named MultiSysView.
MultiSysView consists of a cluster object (a star within a circle) that represents all of the LNM networks defined in the initialization file. This aggregate object, LNM_Networks, has a resource type of *LAN network*. From MultiSysView, you can select more detailed views that display your LNM networks.

**Viewing LNM Networks**

LNM views consist of the following levels:

**LNM LAN Networks View**
A single view showing all your LNM networks

**LNM LAN Views**
Views showing the segments, bridges, and connecting bridge links that make up a single LNM network

**Segment Views**
Views showing the possible components of a segment:
- Segment real object
- Bridge adapters
- Station adapters
- Hubs
- Bridge adapters
- Bridge application program

**Hub Views**
Views showing the components of a controlled access unit (CAU):
- Internal CAU adapters
- CAU microcode

**LNM LAN Networks View**
MultiSystem Manager displays a separate LNM network for each LNM defined in your initialization file. To view your LNM networks, select the LNM_Networks object, as displayed in Figure 51 and select More Detail. The resulting view, as Figure 52 on page 78 shows in our example, one LNM network comprises the aggregate object LNM_Networks.
MultiSystem Manager represents each LNM network by two connected symbols:

- A node aggregate object called a Local area network, for example, B2288P50_LNM_Network.
- An application real object called a LAN network manager, for example, B2288P50_TRLAN_Mgr.

The local area network represents all the network resources monitored by LNM. The LAN network manager, B2288P50_TRLAN_Mgr or NTB7P106_TRLAN_Mgr, represents the topology agent residing in the LNM program. The name of the agent is comprised of the SNA service point address and the type of network the agent is managing (token-ring local area network).

To view additional information on the LNM network, select the LNM object and select Resource Properties. Figure 53 on page 73 shows the resulting NetView management console Resource Properties window for the selected LNM.
LNM Views
If you want to see the segments and bridges managed by an LNM, select the aggregate network node and select More Detail. Figure 54 on page 79 shows the view of the LNM network B2288P50_LNM_Network shown in Figure 52 on page 78.

![Figure 53. NetView management console LNM Resource Properties Window Example](image)

LNM Views
If you want to see the segments and bridges managed by an LNM, select the aggregate network node and select More Detail. Figure 54 on page 79 shows the view of the LNM network B2288P50_LNM_Network shown in Figure 52 on page 78.

![Figure 54. NetView management console LNM LAN View Example](image)

Segment Views
You can display the components of a segment, such as SEG0C05, as shown in Figure 55 on page 80. NetView management console shows the adapters on the segment as shown in the example in Figure 55 on page 80.
Figure 55 shows a token ring segment that consists of one hub, 5A980C5D. Figure 55 also shows a node, SEG0C05_Media, that is not attached to the token ring. This object, a token ring segment, represents the status of the token ring that can not be attributed to a specific adapter on the ring. For example, if a beaconing condition is detected on the ring, it might not be possible to determine which adapter initiated the beaconing. In this case the token ring segment, SEG0C05_Media, would change status to indicate a problem exists on the ring.

Segment Views on Networks Other Than Token Ring
MultiSystem Manager provides limited support for networks other than IBM token ring networks. To determine the type of segment in your view, select the segment object, then select Resource-Information and check the Other data field.

Hub Views
From the view in Figure 55, you can view the components of the hub. To view the components of the hub, select it and select More Detail. The NetView management console builds a view of the hub as shown in Figure 56 on page 81.
Figure 56. NetView management console Hub View (5A980C5D) Example

The view in Figure 56 shows that this hub is comprised of a controlled access unit (CAU), 5A980C5D_Microcode and three CAU adapters. These adapters represent the primary input, primary output, and secondary adapters. If the hub goes into wrap status, the status of the controlled access unit is set to intermediate.
Chapter 6. NetFinity Operation

This chapter explains how to initialize MultiSystem Manager, start and stop processing, and use the views and command support to manage your NetFinity networks.

MultiSystem Manager Views

After you initialize network topology and status in RODM, the NetView management console provides views of your NetFinity networks. Using the NetView management console pull-down menus, you can navigate among the views to isolate failures and send commands to resolve problems.

NetFinity View Objects

In NetFinity views, the following objects are common:

- Aggregate objects that represent:
  - A collection of NetFinity groups
  - A group of NetFinity systems
  - NetFinity systems managed by MultiSystem Manager (workstation aggregate)

- Real objects that represent:
  - Programs (displayed through the NetFinity Process Manager object)
  - Systems performance (displayed through the NetFinity Monitor Manager object)
  - The NetFinity Security Monitor

Figure 57 shows the various objects that are displayed in NetFinity network views.

![Figure 57. Types of NetFinity Objects]
Finding Resources

To display resources, access the Locate Resource window. Using the Locate Resource window, you can locate a NetFinity system and other NetFinity resources by their names.

Navigating Network Views

This section illustrates what you see when you navigate the views of a NetFinity network. The following GETTOPO command, sent to a topology agent with a host name of GULLIVER, was used to build the NetView management console views:

```
GETTOPO NFRES, SP=&GULLIVER
```

To navigate the views, begin by selecting the default MultiSystem Manager network view, **MultiSysView**, from the NetView management console business view, as shown in Figure 58.

Figure 58 also shows the network view, MultiSysView.

![Figure 58. NetView management console Default Network View (MultiSysView) Example](image)

MultiSysView consists of an object (a NetView management console cluster aggregate object) that represents all the NetFinity networks defined in the initialization file. This aggregate object is named **NetFinity_Networks** and has a resource type of **Group**. From MultiSysView, you can display more detailed views of your NetFinity networks.

Viewing NetFinity Networks

NetFinity views consist of the following levels:

- **NetFinity Networks View**
  - Single view that shows all your NetFinity groups

- **NetFinity Network Views**
  - Views showing the NetFinity groups within a network

- **NetFinity group Views**
  - Views showing the NetFinity systems that comprise a particular group
**NetFinity system Views**

Views showing the operating system of each system, Security Monitor, Monitor Manager, and Process Manager

**NetFinity Networks View**

To create the NetFinity networks view, select the NetFinity_Networks object displayed in [Figure 58 on page 84](#) and select More Detail. The resulting view, [Figure 59](#), shows the NetFinity networks that comprise the aggregate object NetFinity_Networks.

![Figure 59. NetView management console NetFinity Networks View Example](image)

A separate NetFinity network for each GETTOPO statement defined in your initialization file is displayed.

Each NetFinity network is represented by two connected symbols: a node aggregate object (NetFinity_network aggregate) and an application real object (NetFinity manager). In [Figure 59](#), the aggregate object (GULLIVER_NetFinity_Network) represents all the NetFinity resources in the network. The real object (GULLIVER_NetFinity_Mgr) represents the part of the MultiSystem Manager topology agent that resides on the service point.

The agent name is comprised of the service point address and the type of network (NetFinity) the agent is managing. To find additional information on the topology agent, select the NetFinity manager and select Resource Properties. The window shown in [Figure 60 on page 84](#) is the result.
NetFinity Network Views

Opening the network object associated with the service point from the NetFinity Networks view displays a view containing aggregate objects. Each NetFinity network object represents a NetFinity group (a collection of NetFinity systems) that was defined by using the Remote System Manager function of NetFinity on that service point. To see the groups contained in the aggregate object, for example, select the aggregate object shown in Figure 59 on page 85 and select More Detail. Figure 61 displays a view of the NetFinity groups.

Note: Each NetFinity manager can define different and unique groupings, however, the MultiSystem Manager agent will gather only group information defined on the service point machine.
**NetFinity Group Views**

Each NetFinity group consists of a number of NetFinity systems. To see the systems that make up a group, do one of the following:

- select a group in like those shown in Figure 61 on page 86, and select More Detail
- double-click the left mouse button on the group

Figure 62 shows a group in More Detail.

![Figure 62. NetView management console Group View Example](image)

These views contain a collection of aggregate object icons representing workstations.

**NetFinity System Views**

Opening a workstation object displays a view which can contain a variety of objects, depending on the parameters that were specified on the GETTOPO commands. Figure 63 on page 88 shows the NetView management console view containing a real object.
All, some, or none of the following objects may be present on the view of the workstation:

**Process Manager**
This object represents the applications being monitored. The presence of this icon indicates that Process Manager alerts are present in this view. The Process Manager object may be accompanied by attached objects. These objects are the individual applications being monitored. If the application objects are:
- Green, they are not in alert condition.
- Red, they are in an alert condition.

**Monitor Manager**
This object represents the overall status of the system. The presence of this icon indicates that Monitor Manager alerts are present in the view.

The Monitor Manager object may be accompanied by attached objects. These objects represent outstanding system alerts against this resource.

**Security Monitor**
This object represents the Security Monitor of NetFinity. The presence of this icon indicates that security alerts are being generated by NetFinity and forwarded to this icon.

This view displays an icon representing the operating system on which the workstation is running. One of the following objects will be present in this view.

**Windows**
The workstation this view represents is running the Windows operating system.

**Windows NT**
The workstation this view represents is running the Windows NT operating system.

**Windows 95**
The workstation this view represents is running the Windows 95 operating system.
OS/2 The workstation this view represents is running the OS/2 operating system.

NetWare The workstation this view represents is running the NetWare operating system.

Note: The display name for each of these objects will be the NetFinity name for this resource.

These objects receive alerts from other sources and are not representative of the overall status of the workstation. For example, these objects may receive alerts from IBM LAN Server via the FFST™. These objects, therefore, may not generate resolves properly and operator intervention may be required to return the icon to a satisfactory status.
Chapter 7. Open Topology Interface Network Operation

You can use the MultiSystem Manager Open topology feature to create your own topology agents and to manage diverse network resources, those resources not managed by other MultiSystem Manager topology features, for example.

You may also obtain applications from Tivoli Partner organizations that use the Open Topology Interface. Partner applications, that extend Tivoli NetView for z/OS management to dozens of network and systems resources that are not managed by Tivoli features, are available.

This chapter explains initializing MultiSystem Manager, starting and stopping MultiSystem Manager processing, and using the views and command support to manage open networks. See "Creating Applications to Manage New Topologies" on page 11 for more information on creating Open topology agents.

MultiSystem Manager Views

After you initialize network topology and status in RODM, the NetView management console provides views of your Open networks. Using the NetView management console pull-down menus, you can navigate among the views to isolate failures and send commands to resolve problems.

Open Topology Interface View Objects

The Open Topology Interface can be used to display views containing any object in the Tivoli NetView for z/OS Data Model Reference. The following Open objects are displayed in various Open views:

- A group of Open networks
- A single Open network
- A collection of resources grouped by location, business organization, or network systems topology
- Aggregate nodes
- Real nodes
- Aggregate links
- Real links

Figure 64 on page 92 shows the various types of objects that are displayed when using the Open Data Model.
Finding Resources

To find resources, access the Locate Resource window. From the NetView management console Tasks menu, select Locate Resource... to open the Locate Resource window. From the Locate Resource window, you can locate a resource by its display name.

Navigating Network Views

How you navigate through an Open network depends upon the views provided by your application. Open views may consist of different levels, as defined by the application used. By default, MultiSystem Manager creates 3 high-level objects for every Open Topology Interface application and displays them in two views. This topic illustrates the default set of views displayed when you navigate an example Open network. In the example shown in this chapter, the application is named 'Open.' The following GETTOPO command, sent to a topology agent with an LU name of NTB6I133, was used to build this set of views:
GETTOPO OPENRES, SP=NTB6I133, APPL=OPEN

To navigate the views, begin by selecting the default MultiSystem Manager network view, MultiSysView, from the NetView management console business tree as shown in Figure 65 on page 93.

Figure 65 on page 93 shows the NetView management console network view named MultiSysView.
MultiSysView consists of a cluster object (a star within a circle) that represents all the Open networks defined in the initialization file used to initialize MultiSystem Manager. This aggregate object is named Open_Networks and its resource type is Group. From MultiSysView, you can select more detailed views that display your Open networks.

**Viewing Open Networks**

The networks object is displayed in its own view, as shown in [Figure 65 on page 92](#). One or more pairs of Network and Manager objects are created as children of the Networks object. A pair of these objects are shown in [Figure 66 on page 93](#).

**Open Networks View**

MultiSystem Manager displays a separate Open network for each one defined in your initialization file. To view your Open networks on the NetView management console, select the Open_Networks object as shown in [Figure 65](#) and select **More Detail**. The resulting NetView management console view, as shown in [Figure 66 on page 93](#) in our example, shows one Open networks object containing one Open network object.
MultiSystem Manager represents each Open network by two connected symbols:

- A node aggregate object called a **network aggregate**, for example, `NTB6I133_OPEN_Network`.
- An application real object called a **manager**, for example, `NTB6I133_OPEN_Mgr`.

The network aggregate represents all the resources monitored by the Open application. The manager, `NTB6I133_OPEN_Mgr`, represents the topology agent residing in the Open application. The name of the agent is comprised of the SNA service point address and the type of network the agent is managing (Open network).

To view additional information on the Open network on NetView management console, select the Open manager and select **Resource Properties**. Figure 67 shows the resulting NetView management console Resource Properties window for the selected Open object.

![NetView management console Open Networks View (Open_Networks) Example](image1)

**Figure 66. NetView management console Open Networks View (Open_Networks) Example**

![NetView management console Open Topology Interface Resource Properties Window Example](image2)

**Figure 67. NetView management console Open Topology Interface Resource Properties Window Example**

94 Tivoli NetView for z/OS MultiSystem Manager User’s Guide
An Open Topology Interface application will consist of one or more additional levels of views. These views may consist of real objects and aggregate objects with any of these objects shown linked together. Refer to the information with each of your Open Topology Interface applications for a description of navigating the views contained in a network aggregate object.
Chapter 8. Tivoli Management Region Operation

This chapter explains how to initialize MultiSystem Manager, start and stop MultiSystem Manager processing, and use the views and command support to manage your Tivoli management region networks.

MultiSystem Manager Views

After you initialize network topology and status in RODM, the NetView management console provides views of your Tivoli management region networks. Using the NetView management console pull-down menus, you can navigate among the views to isolate failures and send commands to resolve problems.

Tivoli Management Region View Objects

The following Tivoli management region objects are displayed in various Tivoli management region views:

- Aggregate objects that represent:
  - A group of Tivoli management region networks
  - A single Tivoli management region network
  - Managed regions
  - User-defined policy regions
  - Managed nodes (IP Systems)
  - Gateways
  - Endpoints

- Real objects that represent:
  - Topology agents
  - Monitors

Figure 68 shows the various types of objects that you can see in your MultiSystem Manager views of Tivoli management region.

![Figure 68. Types of Tivoli management region View Objects](image)

Finding Resources

To find resources, access the Locate Resource window. From the NetView management console Tasks menu, select Locate Resource... to open the Locate Resource window. From this window, you can locate the Tivoli management region device by its names. You can also locate a resource by its display name.
Navigating Network Views

This topic illustrates the views that are displayed when you navigate a Tivoli management region network. The following GETTOPO command, sent to a topology agent with a host name of MSMTIV2, was used to build this set of NetView management console views:

```
GETTOPO TMERES, SP=MSMTIV2
```

To navigate the views, begin by selecting the default MultiSystem Manager network view, **MultiSysView**, from the NetView management console business tree as shown in Figure 69 on page 98.

Figure 69 shows the NetView management console network view named MultiSysView.

MultiSysView consists of an object (a NetView management console cluster aggregate object) that represents all the Tivoli management region networks defined in the initialization file used to initialize MultiSystem Manager. This aggregate object is named TME10_Networks and its resource type is Group. From MultiSysView, you can get more detailed views that display your Tivoli management region networks.

Viewing Tivoli Management Region Networks

Tivoli management region views consist of the following levels:

**Tivoli Networks View**
A single view that shows all your Tivoli management region networks

**Tivoli Network Views**
Views showing the Tivoli management region and policy regions of a Tivoli management region network

**TMR View**
A view showing the managed nodes in a Tivoli management region in their repeater configuration
TME® Policy Region
A view showing the managed nodes and sub-policy regions in a policy region

TME Managed Node View
A view showing the TME 10 Distributed Monitoring monitors associated with a managed node or endpoint

Tivoli Networks View
To create the Tivoli management region networks view for NetView management console, select the TME10_Networks object displayed in Figure 69 on page 98 and select More Detail. The resulting view, Figure 70, shows the Tivoli management region networks that comprise the aggregate object Tivoli_Networks.

This view shows one Tivoli management region network. MultiSystem Manager displays a separate Tivoli management region network for each Tivoli management region agent defined in your initialization file. You will see a separate Tivoli management region network for each GETTOPO statement you have in your initialization file.

Each Tivoli management region network is represented by two connected symbols:
• a node aggregate object (MSMTIV2_Tivoli_Network)
• an application real object (MSMTIV2_Tivoli_Mgr)

In Figure 70, the aggregate object represents all the managed Tivoli management region resources in the network. The real object represents the MultiSystem Manager topology agent. The name of the agent is comprised of the TCP/IP host name and the type of network the agent is managing, Tivoli management region.

To find additional information on the topology agent for NetView management console, select the Tivoli management region manager and select Resource Properties. Figure 71 on page 100 shows the resulting Resource Properties window for the selected Tivoli management region manager on the NetView management console.
Tivoli Network Views

To view the managed Tivoli management region resources in the network, select one of the managers, examples of which are shown in Figure 70 on page 99, then select More Detail. Your NetView management console creates a view showing the Managed Region aggregate and Policy Region aggregates which are related to the manager in Figure 72 and Figure 73 on page 101.
Tivoli Management Region Views
Each Tivoli management region Managed Region Aggregate consists of managed nodes. To view the IP Systems Aggregates, select the Managed Region aggregate object and then select More Detail. Figure 73 displays the software distribution configuration for the MSMTIV2 Tivoli management region on NetView management console.

Figure 73. NetView management console Tivoli Policy Region Example

Figure 74. NetView management console IP System aggregate View Example
Tivoli Management Region Lightweight Client Framework
Gateways and End Points

The lightweight client framework (Tivoli Management Architecture) extends the number of nodes a Tivoli management region can manage with a gateway (Tivoli Management Gateway), which is a role taken by one or more managed nodes in a Tivoli management region.

The gateway acts as a surrogate Tivoli management region server for the resources, end points (Tivoli Management Agent), logged into it. The endpoints do not have an oserv running on them, but they can execute methods sent to them by the gateway, where they can then cache locally. Endpoints function like managed nodes, and the topology agent gathers the same information from the endpoints that it does from the managed nodes in a Tivoli management region:

- Tivoli management region number
- IP address
- Host name
- Status
- OS type
- OS release
- OS version

Any managed node that functions as a gateway becomes a repeater, which works out conveniently for the Tivoli management region (physical) view of the topology agent at the Tivoli NetView for z/OS host. The gateways are automatically displayed in the repeater configuration and physical view of the MultiSystem Manager agent. Figure 75 shows an example view of gateways and endpoints.

Figure 75. NetView management console Gateways and Endpoints View Example

Figure 75 shows two endpoints (WNBA411_Endpoint and hooks_Endpoint) that are connected to a gateway (msgmtiv2_Gateway). The gateway also serves as a Tivoli management region server in this view.
Tivoli Management Region Managed Node Views

Each IP System Aggregate consists of one or more monitors. Monitors can be viewed by selecting the IP System aggregate and then selecting More Detail. Figure 76 shows a detailed view of a monitor shown in Figure 74 on page 101.

Figure 76. NetView management console Monitor View Example
Part 3. Appendixes
Appendix A. Initialization Statements

This appendix contains the syntax descriptions for the MultiSystem Manager initialization statements. There are two categories of initialization statements: general and feature specific.

General statements define various aspects of your system and network environment to MultiSystem Manager. They define system defaults, information about the Resource Object Data Manager (RODM) used, and information about your networking environment. General statements are specified in CNMSTYLE.

GETTOPO RES and ONLY statements can be feature specific. They are used to define service points to MultiSystem Manager during initialization. GETTOPO statements are mentioned in this appendix, but are not thoroughly documented in this book. Refer to Tivoli NetView for z/OS Command Reference, or online help, for a description of the GETTOPO command. GETTOPO statements can be specified in the MultiSystem Manager initialization file if you want to retrieve topology and status of your managed network during MultiSystem Manager initialization.

The following initialization statements are required and must be specified in CNMSTYLE:

1. (MSM)function.autotask.MSMdefault
2. COMMON.FLC_RODMNAME
3. COMMON.FLC_RODMAPPL

If you do not modify any other CNMSTYLE statements related to MultiSystem Manager, default values are used.

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

MultiSystem Manager provides a sample initialization file in the DSIPARM data set. The sample initialization file, FLCAINP, contains commented %INCLUDE statements for additional sample files that contain examples of GETTOPO statements.

[Table 11] shows the MultiSystem Manager feature and its sample initialization file.

<table>
<thead>
<tr>
<th>MultiSystem Manager Feature</th>
<th>Sample Initialization File</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Protocol</td>
<td>FLCSIIP</td>
</tr>
<tr>
<td>Internet Protocol TN3270</td>
<td>FLCSITN</td>
</tr>
<tr>
<td>LNM</td>
<td>FLCSILNM</td>
</tr>
<tr>
<td>NetFinity</td>
<td>FLCSINF</td>
</tr>
<tr>
<td>Open Topology Interface</td>
<td>FLCSIOPN</td>
</tr>
<tr>
<td>TMR</td>
<td>FLCSITME</td>
</tr>
</tbody>
</table>
You can use the DISPTOPO command to display some of the information defined by the initialization statements. If you change the initialization statements in CNMSTYLE, you must issue the RESTYLE COMMON command followed by the INITTOPO command to inform MultiSystem Manager of the changes.

Syntax Rules for GETTOPO Statements in the Initialization File

These statements consist of keyword parameters (KEYWORD=variable). The syntax rules for coding these statements are as follows:

• Start each statement on a separate line of the file.
• Separate keyword parameters by blanks or commas.
• Statements containing more than one keyword parameter can span multiple lines. If a statement spans more than one line:
  – Do not split entries in the middle of a keyword parameter.
  – End each continued line with a comma.
  – Do not end the last line with a comma.
• You can code the %INCLUDE statement. Refer to the Tivoli NetView for z/OS Administration Reference for information about the %INCLUDE statement.
• Start comment lines with an asterisk (*) in column 1.
COMMON.FLC_RODMAPPL

Syntax

```plaintext
COMMON.FLC_RODMAPPL=rodm_appl
```

**Purpose of Command**
This statement specifies the RODM application ID used by FLCARODM. The default value is the NetView domain name concatenated with the string MSM.

**Operand Descriptions**

- `rodm_appl`
  Specifies the RODM application ID used by FLCARODM.
COMMON.FLC_RODMNAME

Syntax

```plaintext
COMMON.FLC_RODMNAME=rodm_name
```

Purpose of Command
This statement specifies the name of the RODM on the host that MultiSystem Manager should use to store topology and status information.

Operand Descriptions

**rodm_name**

The name of the RODM on the host that MultiSystem Manager should use to store topology and status information. This is the same name you used to identify RODM to GMFHS.

The name can be from 1–8 alphanumeric characters in length.
**GETTOPO RES and ONLY**

**Purpose of Command**

This statement describes specific information about a MultiSystem Manager feature that is managing a portion of your enterprise. Code one GETTOPO RES or GETTOPO ONLY statement for each MultiSystem Manager feature that has an associated service point. GETTOPO statements are specified in the MultiSystem Manager initialization file.

Table 12 lists the MultiSystem Manager features along with its GETTOPO RES and ONLY statements for each.

**Table 12. MultiSystem Manager Agent and GETTOPO Statements**

<table>
<thead>
<tr>
<th>MultiSystem Manager Feature</th>
<th>GETTOPO RES and ONLY Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Protocol</td>
<td>GETTOPO IPRES, GETTOPO IPONLY</td>
</tr>
<tr>
<td>LNM</td>
<td>GETTOPO LNMRES, GETTOPO LNMONLY</td>
</tr>
<tr>
<td>NetFinity</td>
<td>GETTOPO NFRES, GETTOPO NFONLY</td>
</tr>
<tr>
<td>Open Topology Interface</td>
<td>GETTOPO OPENRES</td>
</tr>
<tr>
<td>Tivoli management region (TMR)</td>
<td>GETTOPO TMERES, GETTOPO TMEONLY</td>
</tr>
</tbody>
</table>

For examples using GETTOPO RES and ONLY, see "Network Configuration Diagrams" on page 37. For more information regarding GETTOPO statements, refer to the *Tivoli NetView for z/OS Command Reference* or online help.
(MSM)COMMON.FLC_DEF_NETW_VIEW

Syntax

(\texttt{(MSM)COMMON.FLC\_DEF\_NETW\_VIEW=MultiSysView/MultiSysView LAN Networks})

\texttt{(MSM)COMMON.FLC\_DEF\_NETW\_VIEW=network\_view\_name/network\_view\_annotation})

Purpose of Command

This statement specifies the default name and description of the network level view in which you want the object representing the network resource to appear.

This default name and description is used if you do not specify the \texttt{NETWORK\_VIEW} keyword on a \texttt{GETTOPO} command.

This statement is optional. If it is coded, you must specify both \texttt{network\_view\_name} and \texttt{network\_view\_annotation} and you must separate them with a right slash (\texttt{/}). See "Defining NetView Management Console Views" on page 28 for additional details on the usage of this statement.

Operand Descriptions

\texttt{network\_view\_name}

The name for your default network view. This is the name that is displayed in the network view list in the NetView management console window.

This name can be 1–32 alphanumeric characters in length. You can use these special characters only: \# \@ \$ \( \) \: \? \' \_ \& \+ \< \>. The first character must be alphabetic or numeric.

\texttt{network\_view\_annotation}

The description of your default network view. It is displayed in the \texttt{Description} field of the NetView management console window.

The description can be 1–32 alphanumeric characters in length. You can use all special characters, except comma (,) and equal sign (=). You can also use embedded blanks.
**Purpose of Command**

This statement specifies the file that contains the information necessary for Multisystem Manager to process exception views. The (MSM)COMMON.FLC_EXCEPTION_VIEW_FILE statement is optional. However, it must be specified if you want Multisystem Manager to perform exception view processing.

Refer to the prologue of the sample member FLCSEXV in the CNMSAMP data set for more detail on the content of the Multisystem Manager exception view file.

**Operand Descriptions**

*exception_view_file_name*

The name of the file that contains the exception view statements for Multisystem Manager. This file must reside in a DSIPARM data set.

The exception view file name must be 1–8 alphanumeric characters in length. There is no default value for *exception_view_file_name*.
(MSM)COMMON.FLC_RODMINT

Syntax

\[(MSM)\text{COMMON.FLC\_RODMINT} = \text{rodm\_retry\_interval}\]

Purpose of Command

This statement specifies the amount of time, in seconds, between retries of a RODM request that has failed because RODM is checkpointing. When RODM is checkpointing to disk, it cannot process certain transactions, so MultiSystem Manager must wait until the checkpoint process is finished. If MultiSystem Manager tries but cannot access the information because RODM is checkpointing, it waits the number of seconds you have specified and tries again.

Operand Descriptions

rodm_retry_interval

The amount of time, in seconds, between RODM retries.

Value: An integer in the range of 0–15

Default: 5
(MSM)COMMON.FLC_RODMRETRY

Syntax

\[(MSM)\text{COMMON.FLC\_RODMRETRY}=3\]

\[(MSM)\text{COMMON.FLC\_RODMRETRY}=\text{rodm\_retry\_count}\]

Purpose of Command
This statement specifies the number of times to retry a RODM request that has failed because RODM is checkpointing. When RODM is checkpointing to disk, it cannot process certain transactions, so MultiSystem Manager must wait until the checkpoint process is finished. If MultiSystem Manager tries but cannot access the information because RODM is checkpointing, it waits and tries again. It repeats this cycle until successful or until the number of retries is exhausted.

Operand Descriptions
\text{rodm\_retry\_count}

The number of times to retry a request after a RODM request failure.

Value: An integer in the range of 0–5

Default: 3
(MSM)COMMON.FLC_RUNCMDRETRY

Syntax

```c
(MSM)COMMON.FLC_RUNCMDRETRY=runcmd_retry_count
```

**Purpose of Command**
This statement specifies the number of times to retry a RUNCMD after an initial failure with a sense code of 0851 (session busy).

**Operand Descriptions**

*runcmd_retry_count*

The number of times to reissue the RUNCMD after the initial failure.

**Value:**
An integer from 0–10

**Default:**
3
(MSM)COMMON.FLC_TCPNAME

Syntax

\[(MSM)\text{COMMON.FLC_TCPNAME}=TCP/IP \text{ stack name}\]

Purpose of Command
This statement specifies the job identifier for the TCP/IP stack that will be used for TCP/IP communication with service points. This statement should be specified if you are using TCP/IP communication over a connection that is not secure. This statement is optional. If this statement is omitted, then the TCP/IP communication routine(s) will use a default TCP/IP stack job identifier of TCPIP.

Operand Descriptions

\(TCP/IP \text{ stack name}\)

The name of the TCP/IP stack.
(MSM)COMMON.FLC_TN3270_FILE

Syntax

(MSM)COMMON.FLC_TN3270_FILE=tn3270_file_name

Purpose of Command
This statement specifies the file that contains the configuration information for TN3270 Management. This information is necessary for MultiSystem Manager to enable the TN3270 Management feature.

Operand Descriptions

 tn3270_file_name
The name of the file that contains configuration information for the TN3270 Management feature. This file must reside in a DSIPARM data set.

The name of the exception view file must be 1 - 8 alphanumeric characters in length. The sample configuration file for the TN3270 Management feature is FLCS3270. There is no default value for tn3270_file_name.
(MSM) function.autotask.MSMdefault

Syntax

```plaintext
(MSM) function.autotask.MSMdefault=default_autotask_name
```

**Purpose of Command**

This statement specifies the default NetView autotask that MultiSystem Manager should use during GETTOPO command processing. The default autotask name is used if the AUTOTASK keyword is not specified on the GETTOPO command and the related service point object is not in RODM.

**Operand Descriptions**

`default_autotask_name`

The name of the default NetView autotask.

MultiSystem Manager provides a predefined autotask that you can specify instead of creating your own. If you want to use the MultiSystem Manager autotask, specify AUTOMSMD for this parameter.

MultiSystem Manager expects the autotask name to be 1–8 alphanumeric characters in length. It accepts only these special characters: #, @, and $. It expects the name to start with an alphabetic character or #, @, or $.
Appendix B. Resource Information Window—Other Data Field

This appendix lists the MultiSystem Manager resource types for each MultiSystem Manager feature, and the contents of the Other data field for each resource type.

The following tables contain the contents of the Other data field in the NetView management console Resource Information window for each of the MultiSystem Manager resource types. This data is stored in the DisplayResourceOtherData field in the Resource Object Data Manager (RODM) for each object.

If more than one field is listed, the fields are listed in the order they are displayed.

### Internet Protocol

*Table 13. Other Data Fields on Resource Information Windows*

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet networks</td>
<td>None</td>
</tr>
<tr>
<td>Internet network aggregate</td>
<td>None</td>
</tr>
<tr>
<td>Internet manager</td>
<td>IP Agent Level=</td>
</tr>
<tr>
<td></td>
<td>Host=</td>
</tr>
<tr>
<td></td>
<td>Port=</td>
</tr>
<tr>
<td></td>
<td>Topology Level=</td>
</tr>
<tr>
<td></td>
<td>Driver Level=</td>
</tr>
<tr>
<td></td>
<td>Map Name=</td>
</tr>
<tr>
<td></td>
<td>Application Name=</td>
</tr>
<tr>
<td>Internet subnet aggregate</td>
<td>Network Name=</td>
</tr>
<tr>
<td>Internet location aggregate</td>
<td>Location Name=</td>
</tr>
<tr>
<td>Internet segment aggregate</td>
<td>Segment Name=</td>
</tr>
<tr>
<td></td>
<td>Segment Type=</td>
</tr>
<tr>
<td>Internet router aggregate</td>
<td>Router Name=</td>
</tr>
<tr>
<td></td>
<td>System Description=</td>
</tr>
<tr>
<td>Internet bridge aggregate</td>
<td>Bridge Name=</td>
</tr>
<tr>
<td></td>
<td>System Description=</td>
</tr>
<tr>
<td>Internet hub aggregate</td>
<td>Hub Name=</td>
</tr>
<tr>
<td></td>
<td>System Description=</td>
</tr>
<tr>
<td>Internet host aggregate</td>
<td>Host Name=</td>
</tr>
<tr>
<td></td>
<td>System Description=</td>
</tr>
<tr>
<td>Internet link aggregate</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>MAC Address=</td>
</tr>
<tr>
<td>Internet interface</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>MAC Address=</td>
</tr>
<tr>
<td>IP Stack aggregate</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>IPhost=</td>
</tr>
<tr>
<td></td>
<td>sysDescr=</td>
</tr>
<tr>
<td></td>
<td>sysOID=</td>
</tr>
<tr>
<td></td>
<td>sysLoc=</td>
</tr>
<tr>
<td></td>
<td>sysContact=</td>
</tr>
</tbody>
</table>
### Table 13. Other Data Fields on Resource Information Windows (continued)

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP system aggregate</td>
<td>IP Address=6 7</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>LAN workstation aggregate</td>
<td>MAC Address=5 7</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>Open system aggregate</td>
<td>MAC Address=6 7</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>TN3270 Client</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>connectionID=</td>
</tr>
<tr>
<td></td>
<td>Domain=</td>
</tr>
<tr>
<td>TN3270 Server aggregate</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>IPHost=</td>
</tr>
</tbody>
</table>

### LAN Network Manager

### Table 14. Other Data Fields on Resource Information Windows

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAN networks</td>
<td>None</td>
</tr>
<tr>
<td>Local area network</td>
<td>None</td>
</tr>
<tr>
<td>LAN network manager</td>
<td>LNM Level=</td>
</tr>
<tr>
<td></td>
<td>Tier4=1</td>
</tr>
<tr>
<td></td>
<td>Reporting Link2=CONTROLLING</td>
</tr>
<tr>
<td></td>
<td>OBSERVING 1</td>
</tr>
<tr>
<td></td>
<td>OBSERVING 2</td>
</tr>
<tr>
<td></td>
<td>OBSERVING 3</td>
</tr>
<tr>
<td></td>
<td>DEFAULT</td>
</tr>
<tr>
<td></td>
<td>Default Reporting Link4=CONTROLLING</td>
</tr>
<tr>
<td></td>
<td>OBSERVING 1</td>
</tr>
<tr>
<td></td>
<td>OBSERVING 2</td>
</tr>
<tr>
<td></td>
<td>OBSERVING 3</td>
</tr>
<tr>
<td></td>
<td>DEFAULT</td>
</tr>
<tr>
<td></td>
<td>Automatic Bridge Link=ACTIVE</td>
</tr>
<tr>
<td></td>
<td>Access Control4=ACTIVE</td>
</tr>
<tr>
<td></td>
<td>Adapter Monitoring=ACTIVE</td>
</tr>
<tr>
<td></td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>NetView for AIX Connection4=DISABLE</td>
</tr>
<tr>
<td></td>
<td>ENABLE</td>
</tr>
<tr>
<td></td>
<td>SNA Address=</td>
</tr>
<tr>
<td>Token-ring segment aggregate</td>
<td>Type=</td>
</tr>
<tr>
<td>LAN segment aggregate</td>
<td>Same as LAN segment</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Other Data Field Contents</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Bridge</td>
<td>AutoLink=YES</td>
</tr>
<tr>
<td>Bridge aggregate²</td>
<td>Same as Bridge</td>
</tr>
<tr>
<td>Bridge link⁴</td>
<td>Bridge= Segment= MAC Address=</td>
</tr>
<tr>
<td>LAN segment</td>
<td>Type=ETHERNET/IEEE 802.3</td>
</tr>
<tr>
<td>Token-ring segment</td>
<td>Type=</td>
</tr>
<tr>
<td>LAN workstation aggregate</td>
<td>MAC Address=⁵⁷</td>
</tr>
<tr>
<td>IP system aggregate</td>
<td>IP Address=⁶⁷</td>
</tr>
<tr>
<td>LAN station adapter</td>
<td>MAC Address= Segment= Universal Address= CAU id=¹ Attachment Module=¹ Lobe=¹ Monitor=YES</td>
</tr>
<tr>
<td>LAN bridge adapter</td>
<td>MAC Address= Segment= Bridge=⁴ Universal Address= CAU id=¹ Attachment Module=¹ Lobe=¹ Monitor=YES</td>
</tr>
<tr>
<td>LAN concentrator aggregate</td>
<td>Segment= Registered=YES</td>
</tr>
</tbody>
</table>
Table 14. Other Data Fields on Resource Information Windows (continued)

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAU adapter</td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>Segment=</td>
</tr>
<tr>
<td></td>
<td>Universal Address=</td>
</tr>
<tr>
<td></td>
<td>CAU id=</td>
</tr>
<tr>
<td></td>
<td>Primary In Adapter</td>
</tr>
<tr>
<td></td>
<td>Primary Out Adapter</td>
</tr>
<tr>
<td></td>
<td>Secondary Adapter</td>
</tr>
<tr>
<td></td>
<td>Monitor²=YES</td>
</tr>
<tr>
<td>Controlled access unit</td>
<td>Segment=</td>
</tr>
<tr>
<td></td>
<td>Registered=YES</td>
</tr>
<tr>
<td>Open system aggregate</td>
<td>MAC Address=³⁶⁷</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
</tbody>
</table>

| NetFinity |

Table 15. Other Data Fields on Resource Information Windows

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group</td>
<td>None</td>
</tr>
<tr>
<td>Network aggregate</td>
<td>Agent Application=</td>
</tr>
<tr>
<td></td>
<td>Agent Element=</td>
</tr>
<tr>
<td></td>
<td>Agent Level=</td>
</tr>
<tr>
<td>Manager</td>
<td>SNA Address=</td>
</tr>
<tr>
<td></td>
<td>Agent Application=</td>
</tr>
<tr>
<td></td>
<td>Agent Element=</td>
</tr>
<tr>
<td></td>
<td>Agent Level=</td>
</tr>
<tr>
<td>NetFinity Group aggregate</td>
<td>None</td>
</tr>
<tr>
<td>LAN Workstation aggregate</td>
<td>Mac address=³⁷</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Version=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>IP System aggregate</td>
<td>IP Address=⁶⁷</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Version=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>Application</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Windows System</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>OS/2 System</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Windows NT System</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Windows 95 System</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>NetWare System</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
</tbody>
</table>
Table 15. Other Data Fields on Resource Information Windows (continued)

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitor Manager</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Program Process</td>
<td>Service Point=</td>
</tr>
<tr>
<td></td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Program</td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Monitor</td>
<td>Mac Address=</td>
</tr>
<tr>
<td>Security Monitor</td>
<td>None</td>
</tr>
<tr>
<td>Open system aggregate</td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
</tbody>
</table>

Open Topology Interface

Table 16. Open Topology Interface Other Data Fields on Resource Information Windows

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group</td>
<td>None</td>
</tr>
<tr>
<td>Network aggregate</td>
<td>Agent Application=</td>
</tr>
<tr>
<td></td>
<td>Agent Level=</td>
</tr>
<tr>
<td>Manager</td>
<td>SNA Address=</td>
</tr>
<tr>
<td></td>
<td>Agent Application=</td>
</tr>
<tr>
<td></td>
<td>Agent Level=</td>
</tr>
<tr>
<td>IP system aggregate</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>LAN workstation aggregate</td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>Open system aggregate</td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
</tbody>
</table>

Tivoli Management Region

Table 17. Other Data Fields on Resource Information Windows

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>TME 10 Networks - Group</td>
<td>None</td>
</tr>
<tr>
<td>TME 10 Network - Network aggregate</td>
<td>None</td>
</tr>
</tbody>
</table>
### Table 17. Other Data Fields on Resource Information Windows (continued)

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Other Data Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>TME 10 Mgr - Manager</td>
<td>Hostname=</td>
</tr>
<tr>
<td></td>
<td>Port=</td>
</tr>
<tr>
<td></td>
<td>TME Agent Level=</td>
</tr>
<tr>
<td></td>
<td>Operating System=</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OSERV process is Up/Down</td>
</tr>
<tr>
<td></td>
<td>Tivoli enterprise console process is UP</td>
</tr>
<tr>
<td>Policy Region Aggregate</td>
<td>Policy Region=</td>
</tr>
<tr>
<td>Managed Region Aggregate</td>
<td>TMR Number=</td>
</tr>
<tr>
<td>IP system aggregate</td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>LAN workstation aggregate</td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
<tr>
<td>Monitor</td>
<td>Hostname=</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td>Open system aggregate</td>
<td>MAC Address=</td>
</tr>
<tr>
<td></td>
<td>IP Address=</td>
</tr>
<tr>
<td></td>
<td>OS=</td>
</tr>
<tr>
<td></td>
<td>OS Ver=</td>
</tr>
<tr>
<td></td>
<td>OS Release=</td>
</tr>
</tbody>
</table>

**Notes:**

1. This information is provided only for adapters contained in or controlled by a CAU.
2. This information is provided only for resources managed by LNM 1.1 or LNME.
3. This information is provided only if the CAU adapter is on a segment that is monitored by LNM or LNME.
4. This information is provided only for resources managed by LNM 2.0 or later.
5. When using the topology correlation function, a typical value example is as follows:
   
   ```
   MAC address=4000A17D006,OS=NT,OS Ver=4.00
   ```
6. When using the topology correlation function, a typical example value is as follows:

   ```
   IPAddress=9.37.36.7,OS=NT,OS Ver=4.00
   ```
7. When using the topology correlation function, the maximum example value is as follows:

   ```
   MAC address=40000A17D006,IP address=9.37.36.7,OS=NT,OS VER=4.00, Segment No.=SEG100B,IPX address=00004444.40000A17D006, SNA Node=PU4657FA,IP HostName=GSMEYERS,Location=E214/503
   ```
Appendix C. Global Variables

Table 18 lists the global variables used by the MultiSystem Manager topology manager to store information about MultiSystem Manager. See “Appendix A Initialization Statements” on page 107 for an explanation of the referenced initialization statements.

Table 18. MultiSystem Manager Global Variables

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLC_CR_TN_*</td>
<td>TN3270 critical resource identifier</td>
</tr>
<tr>
<td></td>
<td>The unique name for each critical client resource identified to be queried.</td>
</tr>
<tr>
<td></td>
<td>Assigned by the IP Stack agent, it is comprised of the IP address and name</td>
</tr>
<tr>
<td></td>
<td>of the client to be queried.</td>
</tr>
<tr>
<td>FLC_DEF_AUTOTASK</td>
<td>Default Autotask Name</td>
</tr>
<tr>
<td></td>
<td>The name of the default autotask that MultiSystem Manager is using to</td>
</tr>
<tr>
<td></td>
<td>process topology and status requests. This name is specified on the</td>
</tr>
<tr>
<td></td>
<td>(MSM)function.autotask.MSMDefault statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_DEF_NETWORK_VIEW_DESC</td>
<td>Default Network View Description</td>
</tr>
<tr>
<td></td>
<td>The description for the default network view. This description is specified</td>
</tr>
<tr>
<td></td>
<td>on the network_view_annotation portion of the (MSM)COMMON.FLC_DEF_NETW_VIEW</td>
</tr>
<tr>
<td></td>
<td>statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_DEF_NETWORK_VIEW_NAME</td>
<td>Default Network View Name</td>
</tr>
<tr>
<td></td>
<td>The name of the MultiSystem Manager default network-level view. This name</td>
</tr>
<tr>
<td></td>
<td>is specified on the network_view_name portion of the (MSM)COMMON.FLC_DEF_</td>
</tr>
<tr>
<td></td>
<td>NETW_VIEW statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_EXCEPTION_VIEW_FILE</td>
<td>Exception View File Name</td>
</tr>
<tr>
<td></td>
<td>The name of the exception view file that MultiSystem Manager is using for</td>
</tr>
<tr>
<td></td>
<td>exception view processing. The name of this file is specified on the (MSM)</td>
</tr>
<tr>
<td></td>
<td>COMMON.FLC_EXCEPTION_VIEW_FILE statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_INFILE_NAME</td>
<td>Initialization Member Name</td>
</tr>
<tr>
<td></td>
<td>The name of the current MultiSystem Manager initialization file. This is the</td>
</tr>
<tr>
<td></td>
<td>name of the initialization member specified on the INITTOPO command. If the</td>
</tr>
<tr>
<td></td>
<td>initialization member was not specified on the INITTOPO command, the default</td>
</tr>
<tr>
<td></td>
<td>initialization member FLCAINP was used. See the online help and the Tivoli</td>
</tr>
<tr>
<td></td>
<td>NetView for z/OS Command Reference for information about the INITTOPO</td>
</tr>
<tr>
<td></td>
<td>command.</td>
</tr>
</tbody>
</table>
### Table 18. MultiSystem Manager Global Variables (continued)

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLC_IPSTACK_RUN</td>
<td>TN3270 Management processing.</td>
</tr>
<tr>
<td></td>
<td>If the value is set to YES, yes, Y or y, then TN3270 Management is set to</td>
</tr>
<tr>
<td></td>
<td>run after the GETTOPO IPRES command has successfully completed.</td>
</tr>
<tr>
<td>FLC_IPSTACK_SC=1</td>
<td>TN3270 query server identifier</td>
</tr>
<tr>
<td></td>
<td>If the value is set to 1, only TN3270 server resources are being monitored.</td>
</tr>
<tr>
<td></td>
<td>No additional queries need to be issued.</td>
</tr>
<tr>
<td>FLC_IPSTACK_SC=2</td>
<td>TN3270 query for other resources identifier</td>
</tr>
<tr>
<td></td>
<td>If you set the value to 2, in addition to the initial query to identify</td>
</tr>
<tr>
<td></td>
<td>TN3270 servers, an additional query must be issued to identify any critical</td>
</tr>
<tr>
<td></td>
<td>clients (as identified by the FLC_CR_TN_* global variable) or wildcards</td>
</tr>
<tr>
<td></td>
<td>(as identified by the FLC_WILDCARD global variable).</td>
</tr>
<tr>
<td>FLC_RODMAPPL</td>
<td>RODM Application ID</td>
</tr>
<tr>
<td></td>
<td>The user application ID used by MultiSystem Manager to access RODM. This ID</td>
</tr>
<tr>
<td></td>
<td>is specified on the COMMON.FLC_RODMAPPL statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_RODMINT</td>
<td>RODM Interval</td>
</tr>
<tr>
<td></td>
<td>The amount of time, in seconds, between retries of a RODM request that</td>
</tr>
<tr>
<td></td>
<td>has failed because RODM is checkpointing. When RODM is checkpointing to</td>
</tr>
<tr>
<td></td>
<td>disk, it cannot process certain transactions, so MultiSystem Manager</td>
</tr>
<tr>
<td></td>
<td>must wait until the checkpoint process is finished. If MultiSystem Manager</td>
</tr>
<tr>
<td></td>
<td>tries but cannot access the information because RODM is checkpointing, it</td>
</tr>
<tr>
<td></td>
<td>waits and tries again. This value is specified on the (MSM)COMMON.FLC_RODM</td>
</tr>
<tr>
<td></td>
<td>INT statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_RODNAME</td>
<td>RODM Name</td>
</tr>
<tr>
<td></td>
<td>The name of the RODM that MultiSystem Manager is using to store topology</td>
</tr>
<tr>
<td></td>
<td>and status information. This name is specified on the (MSM)COMMON.FLC_RODM</td>
</tr>
<tr>
<td></td>
<td>NAME statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_RODMRETRY</td>
<td>RODM Retry Count</td>
</tr>
<tr>
<td></td>
<td>The number of times MultiSystem Manager retries a RODM request that has</td>
</tr>
<tr>
<td></td>
<td>failed because RODM is checkpointing. When RODM is checkpointing to disk,</td>
</tr>
<tr>
<td></td>
<td>it cannot process certain transactions, so MultiSystem Manager must wait</td>
</tr>
<tr>
<td></td>
<td>until the checkpoint process is finished. If MultiSystem Manager tries</td>
</tr>
<tr>
<td></td>
<td>but cannot access the information because RODM is checkpointing, it waits</td>
</tr>
<tr>
<td></td>
<td>and tries again. This number is specified on the (MSM)COMMON.FLC_RODMRETRY</td>
</tr>
<tr>
<td></td>
<td>statement in CNMSTYLE.</td>
</tr>
<tr>
<td>Variable Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FLC_RUNCMDRETRY</td>
<td>RUNCMD Retry Count</td>
</tr>
<tr>
<td></td>
<td>The number of times a RUNCMD is retried after an initial failure that has a sense code of 0851 (session busy). This value is specified on the (MSM)COMMON.FLC_RUNCMDRETRY statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_TN3270_FILE</td>
<td>The TN3270 configuration file that MultiSystem Manager uses for the TN3270 Management feature. This file name is specified on the (MSM)COMMON.FLC_TN3270_FILE statement in CNMSTYLE.</td>
</tr>
<tr>
<td>FLC_TOPOMGR_STATUS</td>
<td>Current Status</td>
</tr>
<tr>
<td></td>
<td>The current status of MultiSystem Manager. See &quot;Issuing the INITTOPO Command&quot; on page 4 for information about initializing MultiSystem Manager.</td>
</tr>
<tr>
<td></td>
<td>ENABLED</td>
</tr>
<tr>
<td></td>
<td>MultiSystem Manager is able to process GETTOPO commands.</td>
</tr>
<tr>
<td></td>
<td>INITIALIZATION_FAILED</td>
</tr>
<tr>
<td></td>
<td>MultiSystem Manager is not able to process GETTOPO commands. An INITTOPO command was issued, but initialization failed. The failure was due to a syntax error or incorrect information in the initialization file.</td>
</tr>
<tr>
<td></td>
<td>INITIALIZING</td>
</tr>
<tr>
<td></td>
<td>An INITTOPO command was issued and MultiSystem Manager is processing the initialization file. Once the initialization file is processed, the status is changed to INITIALIZATION_FAILED or ENABLED.</td>
</tr>
<tr>
<td></td>
<td>NEVER_INITIALIZED</td>
</tr>
<tr>
<td></td>
<td>MultiSystem Manager has not been initialized and is not able to process GETTOPO commands.</td>
</tr>
<tr>
<td></td>
<td>SUSPENDED</td>
</tr>
<tr>
<td></td>
<td>A SUSPTOPO command was issued and the processing of GETTOPO commands is suspended.</td>
</tr>
<tr>
<td>FLC_WILDCARD</td>
<td>TN3270 critical client range identifier</td>
</tr>
<tr>
<td></td>
<td>Identifies a range of critical TN3270 clients to be queried.</td>
</tr>
</tbody>
</table>
Appendix D. MultiSystem Manager Traps and Alerts

The purpose of this appendix is to provide a numeric listing of the MultiSystem Manager alerts that are sent to NetView by the MultiSystem Manager agents.

Tivoli NetView Topology Manager Traps

The following table lists the Tivoli NetView for UNIX and Tivoli NetView for NT traps that are sent to MultiSystem Manager. Tivoli NetView for UNIX and Tivoli NetView for NT send these alerts to indicate problems or status changes on the network.

The MultiSystem Manager automation table sample, FLCAIAUT, receives the alerts and takes the specified action. If you want different actions to be taken, you can modify the sample FLCAIAUT.

For additional information about alerts and traps, refer to the NetView for UNIX documentation.

Table 19. Tivoli NetView for UNIX and Tivoli NetView for NT Traps

<table>
<thead>
<tr>
<th>UNIX Trap ID number</th>
<th>Windows NT Trap ID number</th>
<th>Meaning</th>
<th>Action taken by the automation table sample FLCAIUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>565504401</td>
<td>565504401</td>
<td>Agent started and waiting</td>
<td>Changes the status of the agent to intermediate.</td>
</tr>
<tr>
<td>565504402</td>
<td>565504402</td>
<td>Agent up and ready</td>
<td>Changes the status of the agent to satisfactory and issues a GETTOPO IPRES command to get topology and status of all resources managed by the agent.</td>
</tr>
<tr>
<td>565504403</td>
<td>565504403</td>
<td>Agent down</td>
<td>Changes the status of the agent to unsatisfactory.</td>
</tr>
<tr>
<td>50790441</td>
<td>50790441</td>
<td>Interface managed</td>
<td>Checks to determine if the interface exists in RODM. If the interface exists, the interface up or down traps update the status. If the interface does not exist in RODM, MultiSystem Manager issues a GETTOPO IPDETAIL to obtain status and topology.</td>
</tr>
<tr>
<td>50790442</td>
<td>50790442</td>
<td>Interface unmanaged</td>
<td>Changes the status of the interface to unknown.</td>
</tr>
<tr>
<td>58785794</td>
<td>58785794</td>
<td>Node added</td>
<td>Issues a GETTOPO IPDETAIL command to get node status and topology.</td>
</tr>
<tr>
<td>58785795</td>
<td>58785795</td>
<td>Node deleted</td>
<td>Purges the node from RODM.</td>
</tr>
<tr>
<td>58916866</td>
<td>58916866</td>
<td>Interface up</td>
<td>Changes the status of the interface to satisfactory.</td>
</tr>
<tr>
<td>58916867</td>
<td>58916867</td>
<td>Interface down</td>
<td>Changes the status of the interface to unsatisfactory.</td>
</tr>
</tbody>
</table>
Table 19. Tivoli NetView for UNIX and Tivoli NetView for NT Traps (continued)

<table>
<thead>
<tr>
<th>UNIX Trap ID number</th>
<th>Windows NT Trap ID number</th>
<th>Meaning</th>
<th>Action taken by the automation table sample FLCAIUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>58916968</td>
<td>58916883</td>
<td>Network Unreachable</td>
<td>Updates the User Status field and causes a flag to be set in the upper left corner of the subnet (internet) object on the NetView management console display.</td>
</tr>
<tr>
<td>58916969</td>
<td>58916884</td>
<td>Network Reachable</td>
<td>Updates the User Status field (RODM for internet object) and removes the flag from the upper left corner of the subnet (internet) objects.</td>
</tr>
<tr>
<td>58916970</td>
<td>58916885</td>
<td>Router Unreachable</td>
<td>Updates the User Status field and causes a flag to be set in the upper left corner of the internetRouter object on the NetView management console display.</td>
</tr>
<tr>
<td>58916971</td>
<td>58916886</td>
<td>Router Down</td>
<td>Updates the User Status field and causes a flag to be set in the upper left corner of the internetRouter object on the NetView management console display.</td>
</tr>
<tr>
<td>58916972</td>
<td>58916887</td>
<td>Router Interface Unreachable</td>
<td>Sets the Display Status field to intermediate (131/White).</td>
</tr>
<tr>
<td>58916973</td>
<td>58916888</td>
<td>Router Up</td>
<td>Updates the User Status field and removes the flag from the upper left corner of the internetRouter object on the NetView management console display.</td>
</tr>
</tbody>
</table>

Tivoli Management Region Alerts

Current status can be maintained for the TME 10 Distributed Monitoring monitors displayed by NetView management console for managed nodes. The Tivoli administrator must select the event forwarding option when selecting a monitor for distribution to a managed node. This causes a particular activation of the monitor to generate a Tivoli enterprise console event for that monitor. MultiSystem Manager receives Distributed Monitoring events forwarded to NetView as alerts by the Event Automation Services.

The status of the monitor resources in NetView will be determined by the value of the severity keyword in the event or alert. Tivoli enterprise console severities are:

Table 20. Tivoli enterprise console Severity

<table>
<thead>
<tr>
<th>Tivoli enterprise console Severity</th>
<th>NetView graphical display</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRITICAL</td>
<td>unsatisfactory/red</td>
</tr>
<tr>
<td>FATAL</td>
<td>unsatisfactory/red</td>
</tr>
<tr>
<td>HARMLESS</td>
<td>satisfactory/green</td>
</tr>
<tr>
<td>MINOR</td>
<td>intermediate/white</td>
</tr>
<tr>
<td>WARNING</td>
<td>intermediate/white</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>unknown/grey</td>
</tr>
<tr>
<td>Tivoli enterprise console Severity</td>
<td>NetViw graphical display</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>CLOSED - status only</td>
<td>satisfactory/green</td>
</tr>
</tbody>
</table>
Index

Special Characters

(\texttt{MSM}}\texttt{COMMON.FLC\_DEF\_NETW\_VIEW} 112
(\texttt{MSM}}\texttt{COMMON.FLC\_EXCEPTION\_VIEW\_FILE} 113
(\texttt{MSM}}\texttt{COMMON.FLC\_RODMINT} 20, 114
(\texttt{MSM}}\texttt{COMMON.FLC\_RODMRETRY} 20, 115
(\texttt{MSM}}\texttt{COMMON.FLC\_RUNCMDRETRY} 116
(\texttt{MSM}}\texttt{COMMON.FLC\_TCPNAME} 117
(\texttt{MSM}}\texttt{COMMON.FLC\_TN3270\_FILE} 118
(\texttt{MSM}}\texttt{function.autotask.MSMdefault} 20, 119

A

accessibility information ix
ADAPTERS parameter 26
aggregation priorities, setting 59
alert history 58
alert history file 8
alerts 131
Tivoli management region 132
APPL parameter 20
automatic topology and status 9
automation
AON-based 11
NetView-based 11
RODM-based 11
AUTOMSMD 20

B

benefits, MultiSystem Manager 3
BLDVIEWS 10, 57
books
feedback viii
online viii
ordering viii
bridge views
IP 67
bridge views, IP 71

C

CNMSTYLE
initialization statements 19
CNMSTYLE initialization statements customizing 19
command interface 9
command menu
NetView management console 10
COMMON.FLC\_RODMAPPL 109
COMMON.FLC\_RODMNAME 19, 110
communication protocol 3, 21
Configuration->Parent view
NetView management console 52
configuration diagrams (continued) individual network 42
naming individual networks 43
network aggregate 39
new view, network aggregate 40
correlated aggregate objects 51
correlated resources, viewing 51
correlation 20
networks, different types 50
viewing correlated resources 51
correlator field setting 55, 56
creating additional network views NetView management console 31
creating views 10
creating your own network views 30
Customer Support ix
customizing initialization statements in CNMSTYLE 19

default autotask
(\texttt{MSM}}\texttt{function.autotask.MSMdefault} 20
default network object example NetView management console 33
default network objects, using IP 32
default objects, using Internet Protocol 33
LN M 34
NetFinity 35
Open Topology Interface 35
Tivoli management region 36
default views, changing names
NetView management console 30
disability information ix
displaying individual feature networks 37
DISPTOPO 48
DISPTOPO command 63
DISPTOPO status window 48
DMCS command 64
documentation and tools, external 12
dynamic topology discovery 8
e-mail contact ix
ELEMENT parameter 27
ENDPOINT parameter 28
endpoints 102
event viewer 8
EXCEPTION\_VIEW\_FILE (\texttt{MSM}}\texttt{COMMON.FLC\_EXCEPTION\_VIEW\_FILE} exception views defining in CNMSTYLE 20

F

failing object, finding 58
feedback about publications ix
finding a failing object 58
finding object status 58
finding resources
Tivoli management region 97
FLBSYSD 51
FLC\_CR\_TN\_* 127
FLC\_DEF\_AUTOTASK 127
FLC\_DEF\_NETWORK\_VIEW\_DESC 127
FLC\_DEF\_NETWORK\_VIEW\_NAME 127
FLC\_EXCEPTION\_VIEW\_FILE 127
FLC\_INFILE\_NAME 127
FLC\_IPSTACK\_RUN 127
FLC\_IPSTACK\_SC=1 127
FLC\_IPSTACK\_SC=2 127
FLC\_RODMAPPL 127
FLC\_RODMINT 127
FLC\_RODMNAME 127
FLC\_RODMRETRY 127
FLC\_RUNCMDRETRY 127
FLC\_TN3270\_FILE 127
FLC\_TOPOMGR\_STATUS 127
FLCAINP 19
FLCSDM8 51
FLCSIIP 19, 107
FLCSILNM 19, 107
FLCSINF 19, 107
FLCSIOPN 19, 107
FLCSTIME 19, 107
FLCSTIMR 19
FLCSITN 19, 107
focal point commands, issuing 46
FTP site 12

gateways 102
GETTOPO command 47, 64
ADAPTERS parameter 26
additional parameters
IP 24
LN M 26
NetFinity 26
Open Topology Interface 27
Tivoli management region 27
HEARTBEAT parameter 24
HIDDEN parameter 25
HOSTONLY parameter 27, 64
HOSTS parameter 25
IPONLY 24
IPRES 24
LNMONLY 26
LNMRSS 26
LPMON 26
NFRES 26
OPENRES parameter 27, 64
optional parameters for NetFinity 27
sp parameter 20

G
Index 137

NetFinity (continued)
- operating system resources 50
- operation 83
- optional GETTOPO parameters 27
- other data field 124
- process manager 88
- security monitor 88
- system views 87
- using the default objects 35
- view objects 83
- viewing networks 84

NetView command tree facility 57, 60
NetView Help, using 63
NetView management console
- 1st_Shift_View 31
- changing the name of the default view 30
- command menu 10
- Configuration->Child view 49
- Configuration->Parent view 49, 52
- creating additional network views 31
- default network object example 33
- defining views 26
- MultiSysView 29
- navigating views 49
- network views 29
- Off_Shift_View 31, 32
- Resource->More Detail view 49
- using default network objects
  - IP 32
- using the default network views 29
- network aggregate, changing names 37
- network object
  - Internet Protocol 33, 35
  - LNM 34
  - NetFinity 35
  - Tivoli management region 36
- network objects
  - Internet Protocol 33
  - LNM 34
  - NetFinity 35
  - Open Topology Interface 35
  - Tivoli management region 36
- network objects, creating 36
- network operation
  - getting started 45
  - Internet Protocol 65
  - LAN Network Manager (LNM) 75
  - NetFinity 83
  - Open Topology Interface 91
- overview 45
- Tivoli management region 97
- network problems, resolving 11, 58
- network topology, initializing 45
- network views
  - IP 69
  - NetView management console 29
  - Tivoli management region 100
- network views, navigating
  - Internet Protocol 66
  - LAN Network Manager (LNM) 76
  - NetFinity 84
  - Tivoli management region 98
- networks
  - creating 30
  - using 37
  - networks aggregate, changing names 36

networks aggregate object
- Internet Protocol 33
- LNM 34
- NetFinity 35
- Open Topology Interface 35
- Tivoli management region 36

networks view
- Internet Protocol 68
- NetFinity 85
- Tivoli management region 99
- NFONLY parameter 26
- NFPASS parameter 27
- NFRVICE parameter 26
- NFSSYMONITOR parameter 27
- NMC
  - service views 9

O
- object status, finding 58
- Off_Shift_View 31
- NetView management console
  - 32
  - online help 13, 62
  - online publications viii
- Open Topology Interface
  - additional GETTOPO parameters 27
  - ELEMENT parameter 27
  - HOSTONLY parameter 27
  - network object 35
  - network objects 35
  - networks aggregate object 35
  - node and node aggregate resources 50
  - OPENRES parameter 27
  - operation 91
  - other data field 125
  - using default objects 35
  - OPENRES parameter 27
  - optional GETTOPO parameters
    - LNM 26
    - NetFinity 27
    - Tivoli management region 28
  - ordering publications viii
  - other data field
    - IP 121
    - LNM 122
    - NetFinity 124
    - Open 125
    - Tivoli management region 125

P
- POLICY parameter 28
  - preventing removal, objects 61
  - process manager, NetFinity 88
  - PU name 21
  - publications
    - feedback viii
    - online viii
    - ordering viii

R
- REMOTE parameter 20
  - REMOTE parameter, coding 22

removing resources
- aggregate objects 61
- objects from views 60
- objects meeting criteria 60
- real objects 61
- REMVOBJS 60, 61
- REMVOBJS command 64
- resolving network problems 58
- resource information window
  - other data field 121
  - IP 121
  - LNM 122
  - NetFinity 124
  - Open 125
  - Tivoli management region 125

Resource Information window
- aggregation priority (real objects only) 49
- customer data 49
- degraded (aggregate objects only) 49
- managed by 49
- operator status 49
- other data 49
- resource name 49
- severely degraded (aggregate objects only) 49
- system status 49
- total resources (aggregate objects only) 49
- type 49
- unsatisfactory (aggregate objects only) 50
- unsatisfactory/exception resources (aggregate objects only) 49

RESTOPO command 47, 64

RODM
- checkpointing 20
- correlated aggregate objects 51
- defining 19
- defining, GMFHS 19

RODMINT
  - (MSM)COMMON.FLC_RODMINT 20, 114

RODMNAME
  - COMMON.FLC_RODMNAME 19, 110

RODMRETRY
  - (MSM)COMMON.FLC_RODMRETRY 20, 115

router views, IP 67, 71
- rules, syntax 108

RUNCMD retry count
  - (MSM)COMMON.FLC_RUNCMDRETRY 20, 116

S
- security monitor, NetFinity 88
- segment views
  - Internet Protocol 70
  - IP 67
- segment views, LNM 79, 80
- service points, defining 20
- service views, LNM
  - NetView management console 9