TME 10 Global Enterprise Manager
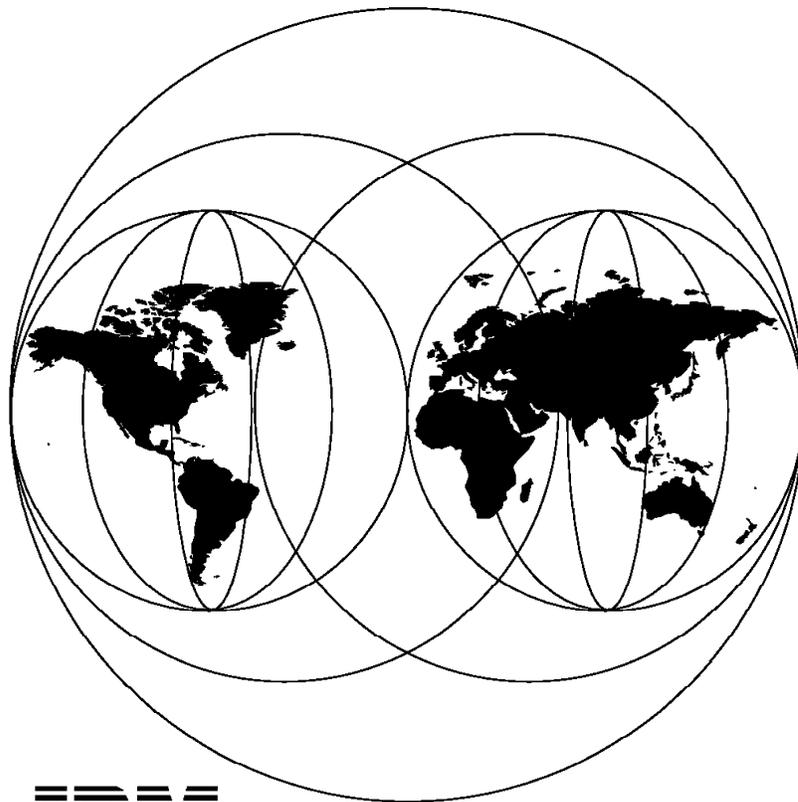# Topology Service and NetView 3270 Java Client

December 1997

International Technical Support Organization

Raleigh Center

IBM

International Technical Support Organization

**TME 10 Global Enterprise Manager
Topology Service and NetView 3270 Java Client**

December 1997

> **Take Note!**
>
> Before using this information and the product it supports, be sure to read the general information in Appendix C, "Special Notices" on page 195.

First Edition (December 1997)

This edition applies to Version 1 Release 1 of TME 10 Global Enterprise Manager, Program Number 5697-B83 for use with MVS/ESA or OS/390. The NetView Java Client described in this book is to be used with TME 10 NetView for OS/390 Version 1 Release 1, Program Number 5697-B82.

> **Note**
>
> This book is based on a pre-GA version of a product and may not apply when the product becomes generally available. We recommend that you consult the product documentation or follow-on versions of this redbook for more current information.

Contents

Figures

Tables

xi

Preface

A combined approach for systems management, including all platforms, has the highest priority in most IT organizations.

The Tivoli Management Environment 10 (TME 10) Global Enterprise Manager (GEM) is a new product that combines most of the needed systems management functions for various platforms into a single system approach.

This redbook helps you to install and customize the TME 10 Global Enterprise Manager and use its Topology, Command and Event/Automation services. A step-by-step approach is used first to get the basic functions running and then extends these to a more sophisticated usage.

The book uses examples and scenarios to show you how to define TME 10 Distributed Monitoring monitors in the distributed environment and then manage your TME environment from TME 10 NetView for OS/390. It provides scenarios showing the flow of events and alerts between TME 10 NetView for OS/390 and T/EC. Used in combination with the new command service, these functions allow you to build customized automation solutions.

This redbook also covers how to install and customize the newly announced TME 10 NetView for OS/390 Java Client.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the Systems Management and Networking ITSO Center, Raleigh.

Arne Olsson is from the Systems Management and Networking ITSO Center, Raleigh. His responsibilities at the ITSO include TME 10 NetView for OS/390 and TME 10 Global Enterprise Manager. Before joining the ITSO he worked as a Systems Engineer in Sweden.

Anders Johansson is from IBM Sweden. Anders has six years of experience in the operation/help desk area, nine years of experience in the system programmer area, two years of experience in the Manager of Network and Operations department and eight years of experience as a consultant in the Network and System Management field.

George Steinborn is from IBM Germany. George is a senior SE in the software support group for Germany and CER. He has many years of experience in the large system environment in different areas. He has been on several international assignments in the U.S. and Far East supporting large systems and SNA networking, NetView/390, RODM and system automation. He is responsible for second level support for TME 10 host products and TME 10 Global Enterprise Manager.

Roberto Castanedo is from IBM Spain. Roberto joined IBM in 1989, working in the Barcelona Lab in charge of the MVS system and communications. In 1994 he moved to RDS to provide systems management solutions to banking customers. Since 1997 he works in a team that implements solutions based on TME 10 products.

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in "ITSO Redbook Evaluation" on page 207 to the fax number shown on the form.

- Use the electronic evaluation form found on the Redbooks Web sites:

   For Internet users   `http://www.redbooks.ibm.com`
   For IBM Intranet users   `http://w3.itso.ibm.com`

- Send us a note at the following address:

      `redbook@vnet.ibm.com`

Chapter 1. Global Enterprise Manager Overview

This chapter briefly describes the components of the Global Enterprise Manager and how they work together. The redbook *TME 10 Global Enterprise Manager, Event/Automation and User Administration*, SG24-4921 focused on Event Automation and User Administration and is the base for this book. Here we present the Topology, Command Integration Services and the new TME 10 NetView for OS/390 Java Client functions. A more detailed discussion of the whole set of Global Enterprise Manager functions is described in *TME10 Global Enterprise Manager, Installation and User's Guide*, available with the product.

1.1 Introduction

The Global Enterprise Manager works as a bridge between the OS/390 and distributed environments. It is designed to integrate system management functions for cross-platform business applications. You have to have prerequisite products that run on the OS/390 platform and the distributed platform. The main functions you need are:

- TME 10 NetView for OS/390 Version 1 Release 1

- The MultiSystem Manager component of TME 10 NetView for OS/390

    This is part of TME 10 NetView for OS/390 if you have ordered the Graphical Enterprise Option.

- The Global Enterprise Manager Event/Automation Service

- The Global Enterprise Manager Command Service

- The Global Enterprise Manager Topology Service which provides the MSM Agent

- A TMR server running the Tivoli framework

- Tivoli Enterprise Console

- The TME 10 Distributed Monitoring 3.5.

- The TME 10 NetView for OS/390 Java Client

1.1.1 Topology Integration

The systems management area has become a more heterogeneous environment. The MultiSystem Manager (MSM) provides an integrated and centralized network management function to manage your network from an NGMF workstation.

The Topology Integration Service of Global Enterprise Manager provides an extended view into the Tivoli Managed Region (TMR) from an OS/390 platform. Using OS/390 as your manager platform, you are able to gain additional information about your TMRs. TME 10 Distributed Monitoring monitors can be displayed in a graphical way using NGMF.

MultiSystem Manager uses a manager-agent relationship to manage its resources. MultiSystem Manager provides a topology manager that runs on OS/390 under NetView. Global Enterprise Manager provides an application that runs on a TMR server and acts as an agent (MSMAgent) to the topology manager on the S/390 platform.

The connection between the manager and agent is established via a TCP/IP network.

With the TME 10 Topology Service you can:

- Request the topology of a TMR using the GETTOPO command from MultiSystem Manager and store the topology objects in the Resource Object Data Manager (RODM).

Once you have the toplogy in RODM the status of the resources will be updated dynamically:

- The TME 10 Distributed Monitoring monitor will send the results of the monitoring to T/EC.
- T/EC will forward the event to the event adapter on the OS/390 host.
- The event adapter will convert the T/EC event into an alert that will be picked up by NetView for OS/390 to report the status change of the object in RODM.
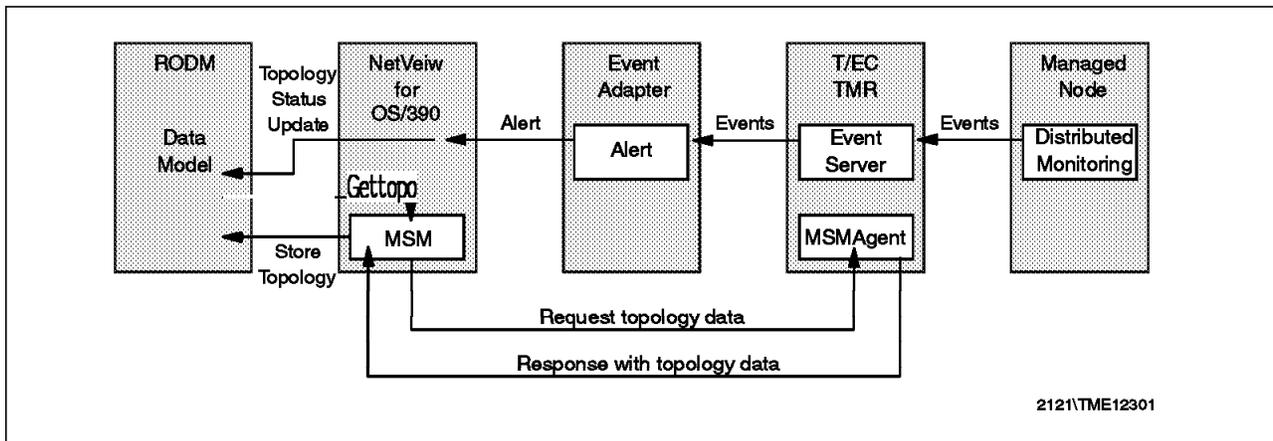
Figure 1. Topology Data Flow

1.1.2 Command Integration

The Global Enterprise Manager command integration service is the vehicle to send commands from a workstation to NetView for OS/390 and vice versa. Used in combination with the Event/Automation Service it provides a set of functions that allows you to take management action on any system/network component from a single workstation. These two services are the base components that allow you to build automation solutions where traditional host-based automation needs to be extended to include a distributed TME environment.

NetView
for OS/390

T/EC
TMR
APM Server
APM Client

MSM → TCP/IP → MSMAgent

OST ← LU6.2 ← CMD - Service

TME123\TME12303
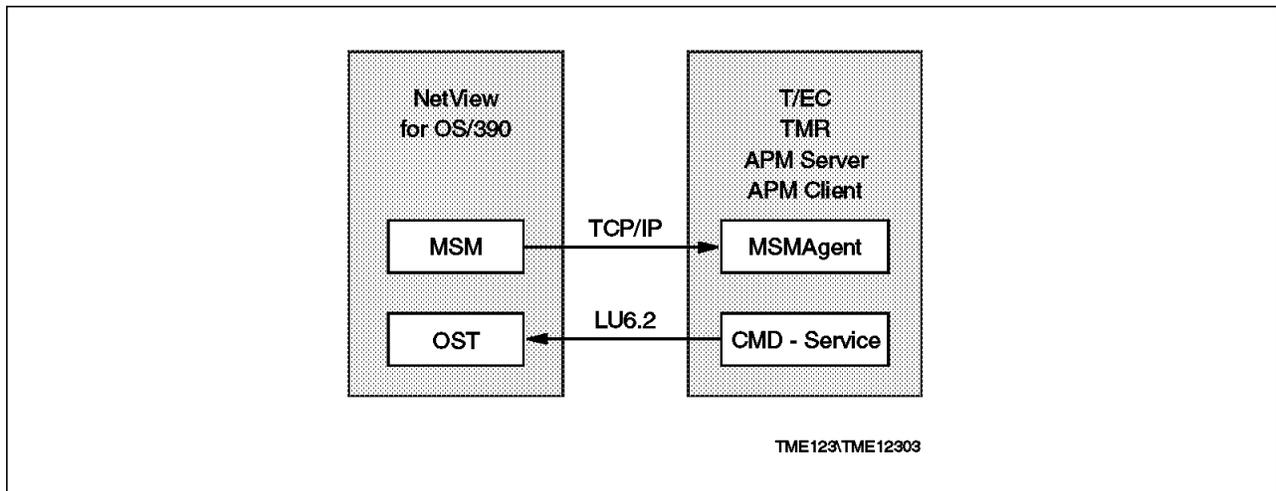
Figure 2. Command Service Data Flow

The Command Integration Service is used in two ways:

1. Using commands from the TME 10 NetView for OS/390 side.

   From your NGMF workstation choose a TME 10 object, open the command tree, and issue resource-specific commands for this specific resource. The command will be sent down to the MSMAgent which will send the command to the appropriate TME 10 resource.

   There is also a TME 10 NetView for OS/390 command line interface that allows you to send commands to your TME resources.

   This command support is provided by TME 10 NetView for OS/390 and the Global Enterprise Manager Topology service.

2. Using the command interface at the Application Policy Manager server.

   This support requires the Global Enterprise Manager Application Policy Manager and allows you to send commands to TME 10 NetView for OS/390 from the Application Policy Manager client. The hostcmd command can be used to send commands to TME 10 NetView from the command line. The Application Policy Manager server needs to be installed on this machine.

   Communication from the APM server to TME 10 NetView for OS/390 is based on an LU 6.2 or a TCP/IP connection. You would need to log on to TME 10 NetView for OS/390 and issue a NETCONV command to the machine where the Application Policy Manager server is running. Then you can send any command up to the host NetView. The command runs under the specific OST task you logged on to.

1.1.3 Application Policy Manager

The TME 10 GEM Application Policy Manager (APM) is a new way to present and manage your business systems. A business system includes all the components that together perform a business function. With Application Policy Manager you are able to graphically monitor and control the applications in your business system. The various components in your business system may reside in either the TME 10 workstations and/or at the OS/390 host. Using Application Policy Manager you can also view the relationship/connections between the components of your business systems. You can also issue commands to the various components in your business system. The actual support provided in your business systems is dependent on the instrumentation of the various

components. You can instrument your applications using the Tivoli Application Management Specification (AMS). During the residency we did not work with Application Policy Manager as such; we just installed it as a prerequisite of the Command Integration Service.

1.1.4 TME 10 NetView for OS/390 Java Client

To communicate with NetView for OS/390 you have to use a 3270 type screen or an emulation program that provides this kind of connection. Since in a heterogeneous world independence of operating systems is a very important issue, Global Enterprise Manager and TME 10 NetView for OS/390 will move in the direction of using Java applications. Examples of such applications are the TME 10 NetView for OS/390 Java Client and Application Policy Manager client.
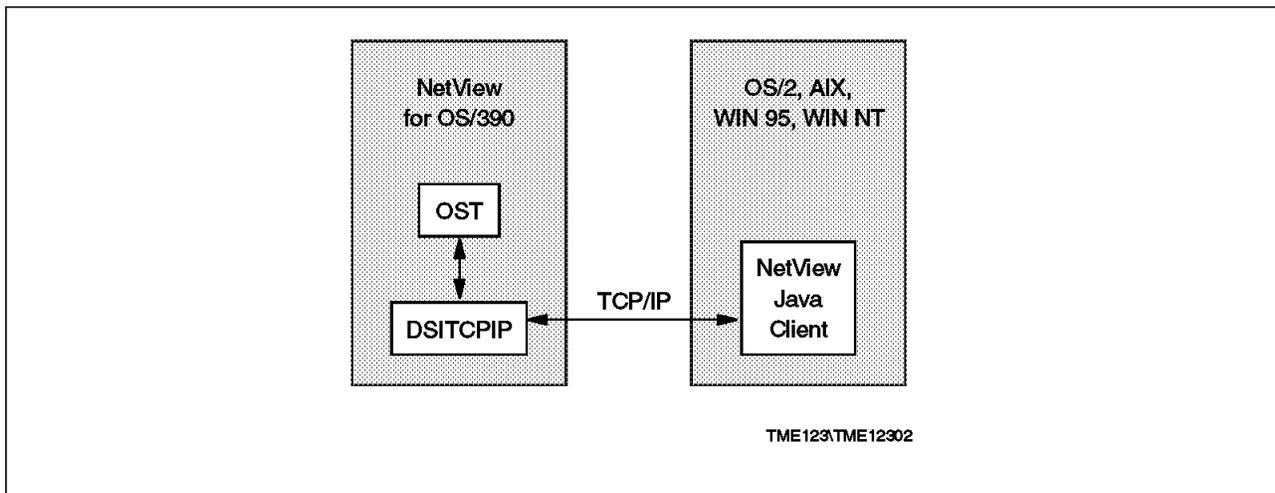
Figure 3. NetView Java Client Data Flow

The TME 10 NetView for OS/390 Java Client is a workstation program that talks to TME 10 NetView for OS/390 via a TCP/IP connection and will run on different workstation platforms using Java technology. The TME 10 NetView for OS/390 Java Client is available for OS/2, AIX, Windows 95 and Windows NT. Using the client, you can access both the command facility and full-screen applications in TME 10 NetView for OS/390.

1.2 The Environment

During our residency we used the ITSO installed test environment shown in Figure 4.
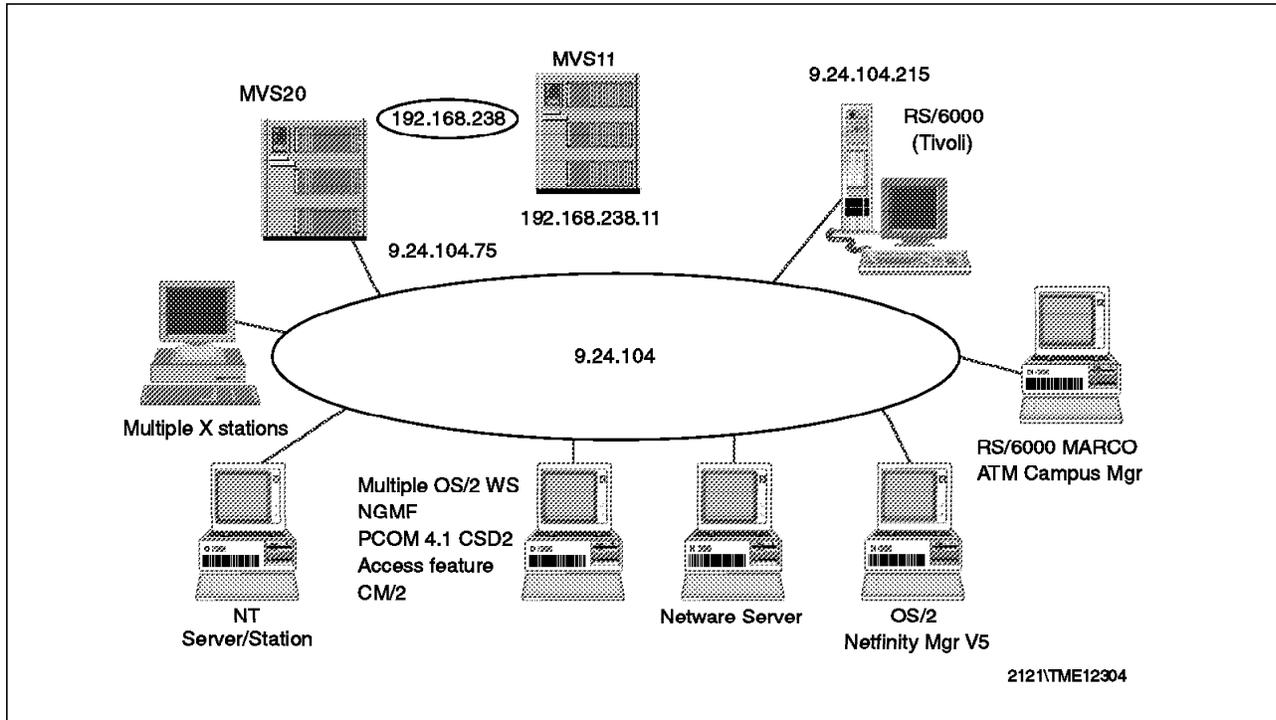
Figure 4. ITSO Configuration

We used MVS11 as our OS/390 server which IP-wise has been defined as a host behind MVS20. We had the latest level of TME 10 NetView for OS/390 installed and used PCOMM and the OS/2 Access Feature for NGMF.

The Tivoli code was initially installed on an RS/6000 called RS/600015. The MSMAgent was installed on the same machine for the duration of the residency. This RS600015 had a two way connection to another TMR called foxtrot. Foxtrot is an NT 4.0 machine. In each TMR we had one NT managed node and their names were wtr05193 and romeo. The configuration is shown in the following picture:

MVS 11
TME 10 NetView for OS/390

NGMF

rs600015

MSMAgent

Interconnected TMRs
Two Way Connection

foxtrot

wtr05193

romeo

2121\2121TOP1
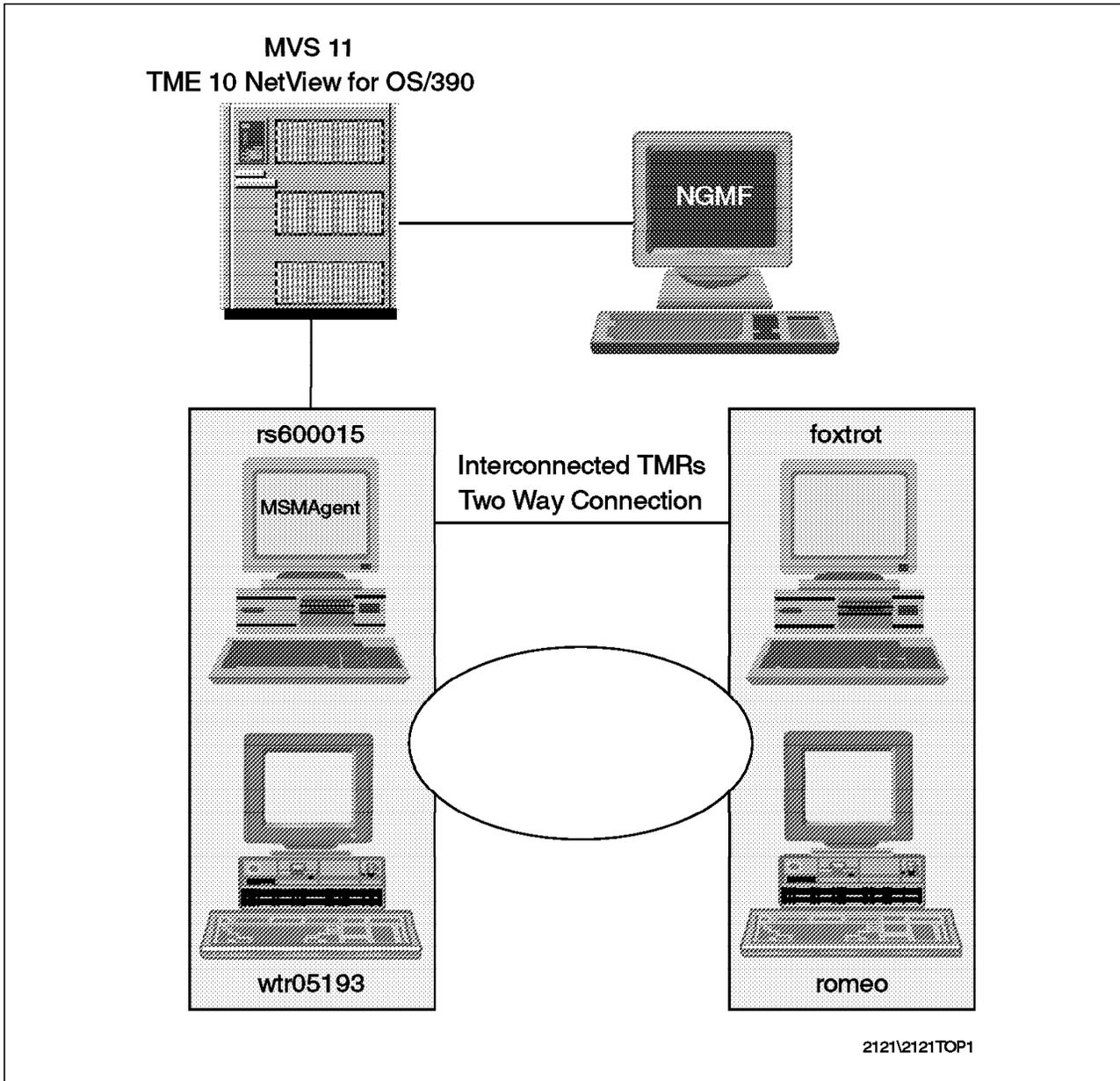
Figure 5. Residency Environment

It should be noted that this configuration was changed later as indicated in 4.1.4.1, "Automation Scenario" on page 92.

Chapter 2. Base Product Installation

Before you start to install and customize the new integration services of TME 10 Global Enterprise Manager you must install some base products on the S/390 side and on your workstations.

2.1 TME 10 NetView for OS/390 Installation

A prior residency had installed the host code for TME 10 NetView for OS/390 so we don't repeat the installation steps here again. Please use the TME 10 NetView for OS/390 installation books in addition to the redbook *An Introduction to TME 10 NetView for OS/390*, SG24-4922.

2.1.1 Global Enterprise Manager Integration Services Event/Automation

The TME 10 Global Enterprise Manager, Event/Automation, installation had been done in an earlier residency. Please refer to the *TME 10 Global Enterprise Manager Installation and User's Guide*, available with the product, in addition to the redbook *TME 10 Global Enterprise Manager, Event/Automation and User Administration*, SG24-4921 for details of how this should be done.

2.2 Installation of the TME 10 Products in the Distributed Environment

We assume that TME 10 Framework and its patches, Service Pack 1 and the Lightweight Client Framework (LCF) patch required for TME 10 Distributed Monitoring are installed and operational. The LCF patch function is available in Framework 3.2 which is required for the general availability version of Distributed Monitoring 3.5. We also assume that the prerequisite database for Tivoli Enterprise Console is installed. The purpose of this section is to provide a guide to what we installed in our environment. As always, please refer to the official documentation and PSP buckets for the latest and most accurate information.

We installed all the products from the desktop, logged on as root. But you can log in as any other TME administrator with the super or Install_product role.

After successful installation of the framework and Service Pack 1 you should have a desktop as shown in Figure 6 on page 8.

If you have installed the Lightweight Client Framework (LCF) patch or Framework 3.2, you will have an additional icon called EndpointManager on your desktop. The LCF patch function is available in Framework 3.2.
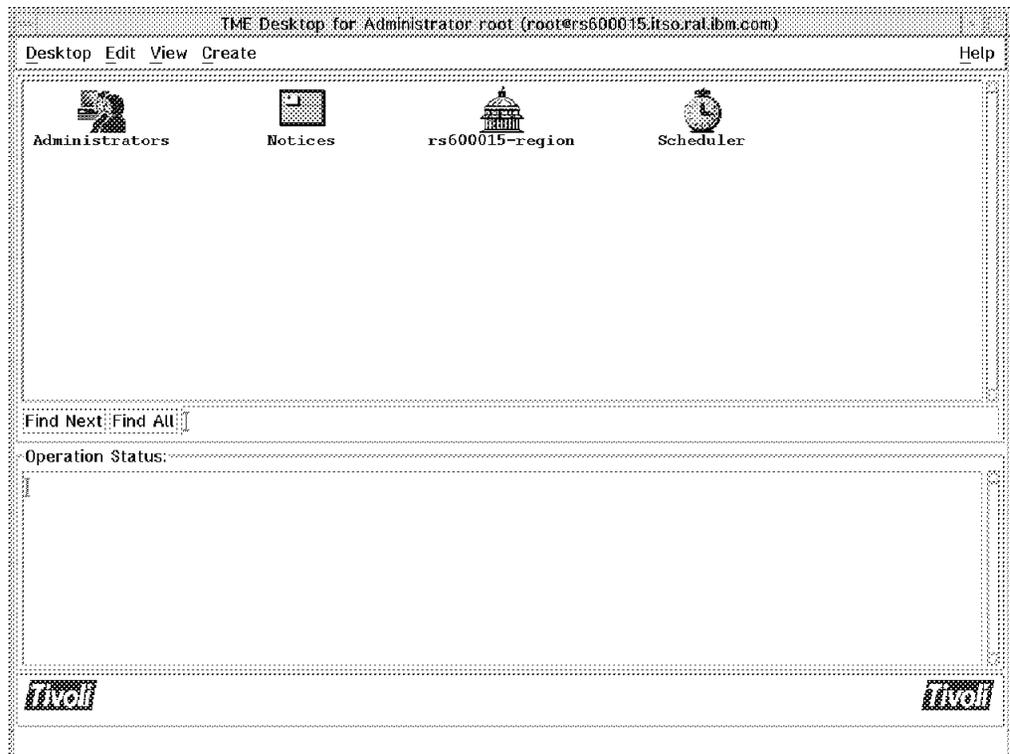
Figure 6. Desktop after Framework Installation

2.2.1 Installation of Tivoli Products

We installed the prerequisite Tivoli products and Global Enterprise Manager Integration Services from the Tivoli desktop with the following:

1. Select **Desktop** from the TME Desktop menu bar.

2. Select **Install = = > Install Product** from the pull-down menu. It will display the Install Product dialog as shown in Figure 7 on page 9.
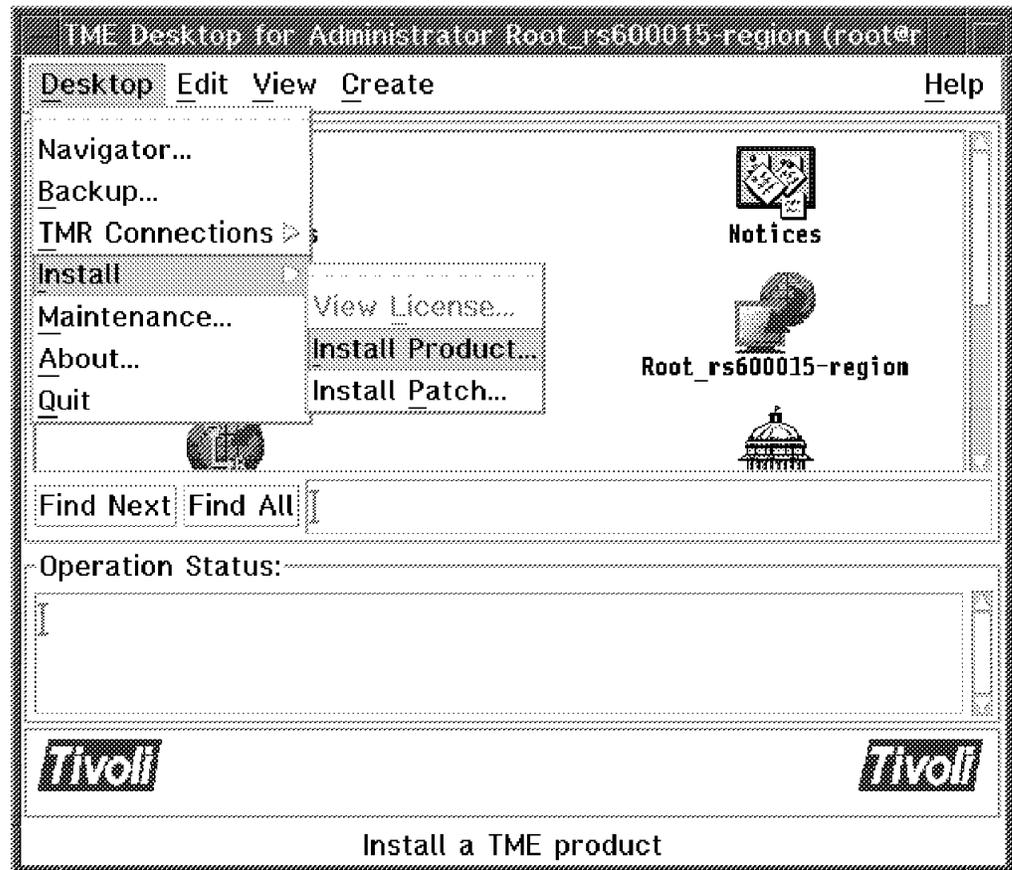
Figure 7. Install Product Dialog

3. If the Select Product to Install does not have the product you want to install or it is empty, then click on **Select Media**. The File Browser dialog will let you change the path to the installation media. Fill in the directory containing the product and finally click on **Set Media & Close**. In our scenario it was located in the /cdrom directory.

4. Highlight the product by clicking on it in the Select Product to Install list.

5. Fill in the Clients to install on list by selecting clients from the Available Clients list.

6. Click on the **Install & Close** or **Install** button. A product install window appears, where you see that checking product dependencies are correct. It asks you to accept the information.

7. Select the **Continue Install** button to begin the installation process. Wait until Finished installation appears on the screen, and then check if there are any errors. Click on the **Cancel** button to exit.

These steps were used for all the products that were used in our residency. The following sections only show you the Select Product to install screens to show what's available on the product CDs and what we installed.

2.2.2 TME 10 Distributed Monitoring Installation

TME 10 Distributed Monitoring 3.5 is a prerequisite product that is required for Global Enterprise Manager Topology Service. It allows you to define the various monitors that can be viewed from NGMF.
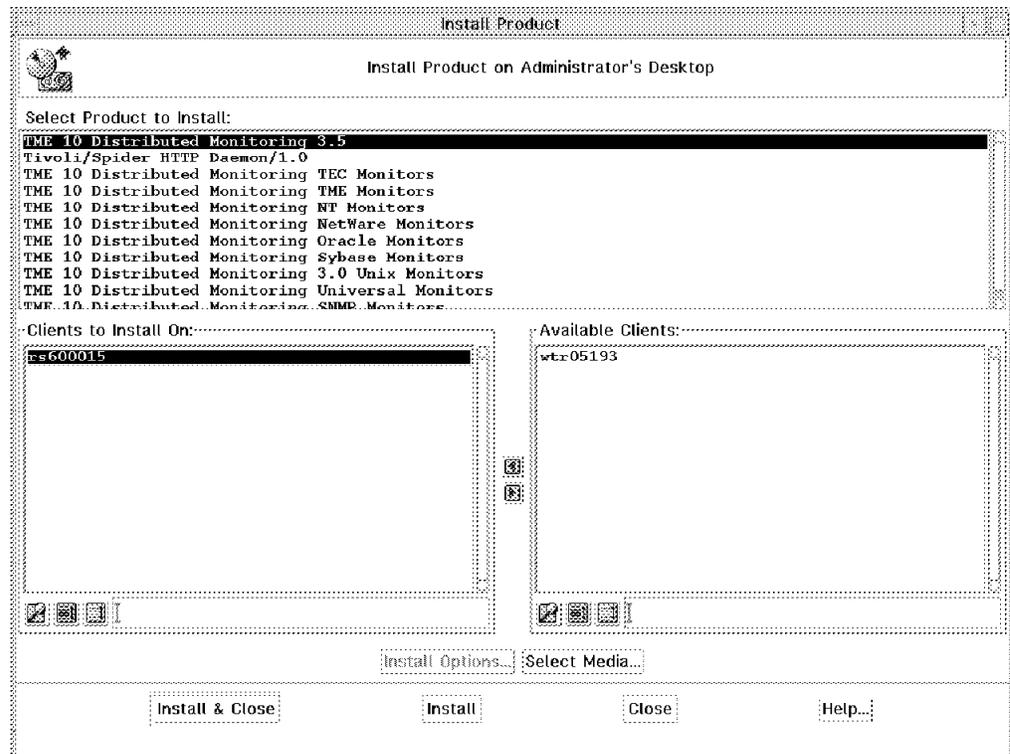
Figure 8. Installation Screen for TME 10 Distributed Monitoring

Distributed Monitoring 3.5 provides various monitoring packages as part of the product. In addition to the base TME 10 Distributed Monitoring 3.5 product, we installed NT, Universal and UNIX monitor packages. Install those that apply to your environment.

2.2.3 TME 10 Enterprise Console 3.1 Server

TME 10 Enterprise Console 3.1 is a prerequisite product for both the Event/Automation and the Topology services. It consists of a server and a console component. The server component must be installed first and requires some customization during the installation.
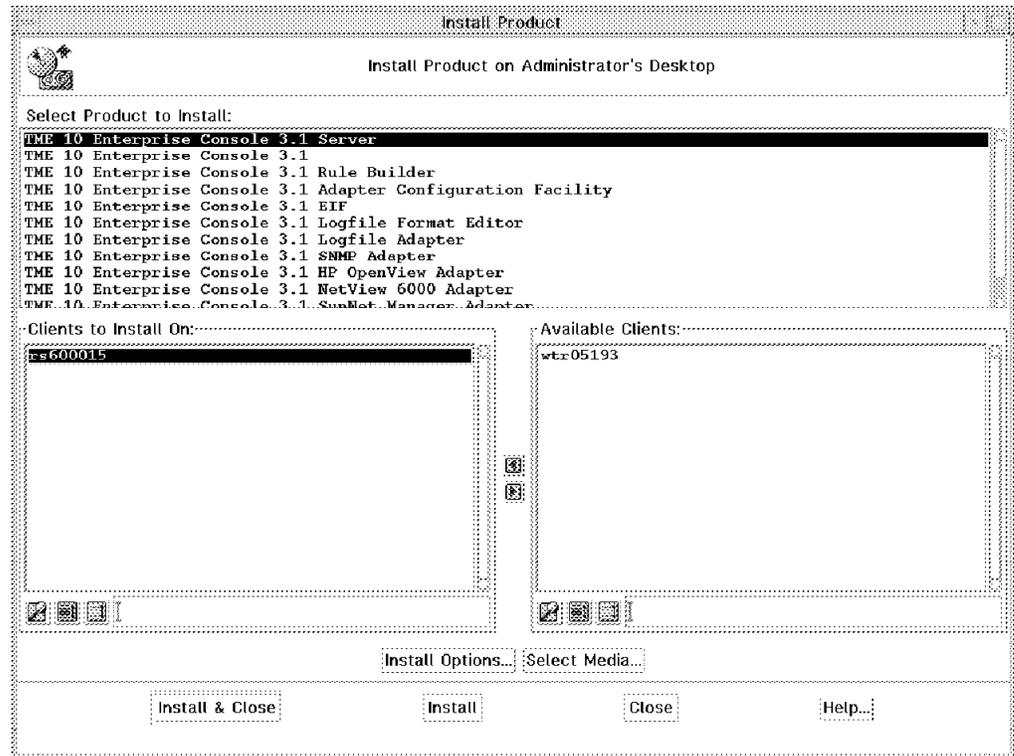
Figure 9. Installation Screen for T/EC Server

The Install Options dialog is displayed, as shown in Figure 10 on page 12. Refer to the *TME 10 Enterprise Console User's Guide* for an explanation of the different options. In our environment we selected:

1. Database Vendor: Sybase

2. Database Home: /usr/local/sybase

3. Database ID: tec

4. Database Server ID: Leave blank if the database server runs in the same system as the T/EC Server.

5. Check the option **Start the Event Server at oserv() start time** if you wish to start the event server when the object dispatcher is started.

6. Click on the **Set** button.
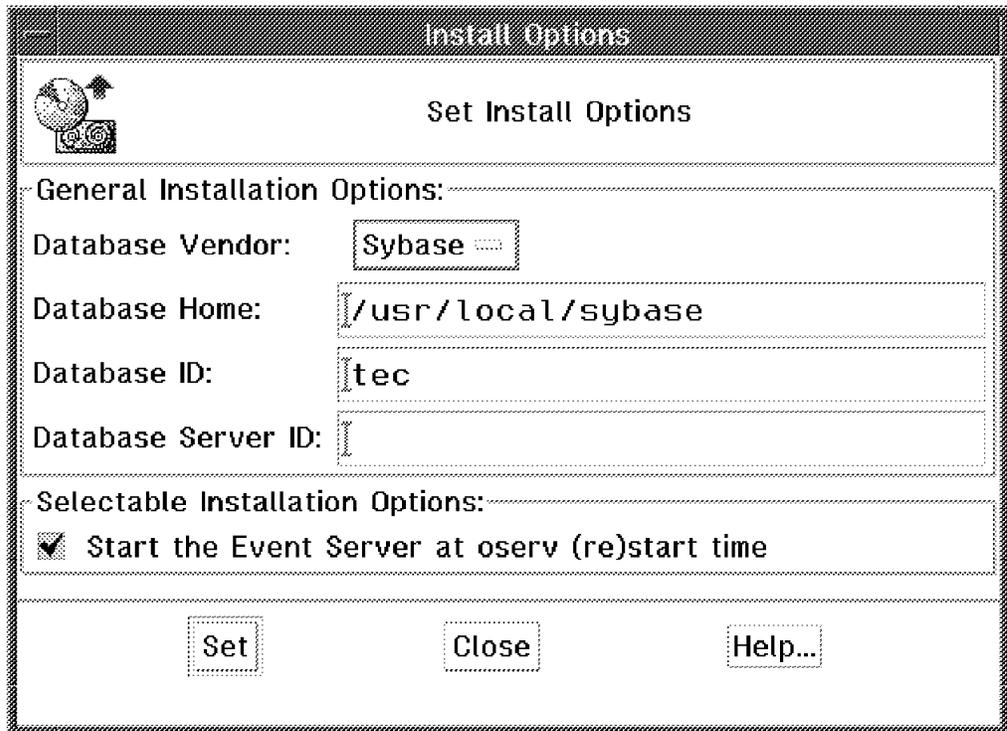
Figure 10. Installation Options for the T/EC Database

Now we are ready to configure the database for T/EC:

1. First, verify that the RIM object is properly installed. Enter the command wgetrim tec. Check that the values in the command response are correct. In our environment the command response was as shown in Figure 11.


```
root@rs600015[/tmp]# wgetrim tec
RIM Host:        rs600015
RDBMS User:      tec
RDBMS Vendor:    Sybase
Database ID:     tec
Database Home:   /usr/local/sybase
Server ID:
root@rs600015[/tmp]#
```
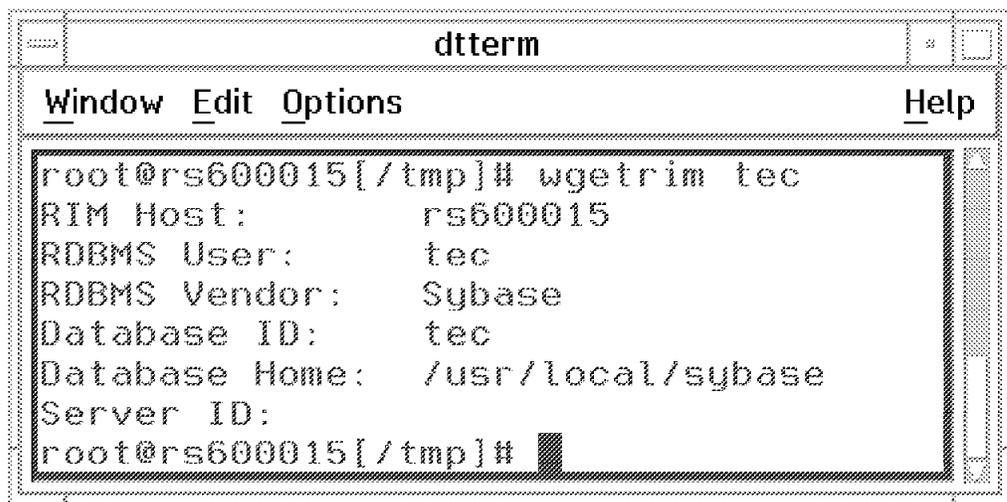
Figure 11. wgetrim Command Output

2. Go to the directory $BINDIR/TME/TEC/sql, and run the database configuration script ./cr_tec_db.sh. You are prompted for the device to contain the tables, the initial database size and the DBA password.

2.2.4 TME 10 Enterprise Console

The TME 10 Enterprise Console must be installed after the server component has been successfully installed. The only other function we installed was the Rule Builder since we wanted to create our own rules. This is optional and we actually created the rules without the graphical user interface. As you see there are many alert adapters on the CD that can be used if appropriate in your environment.
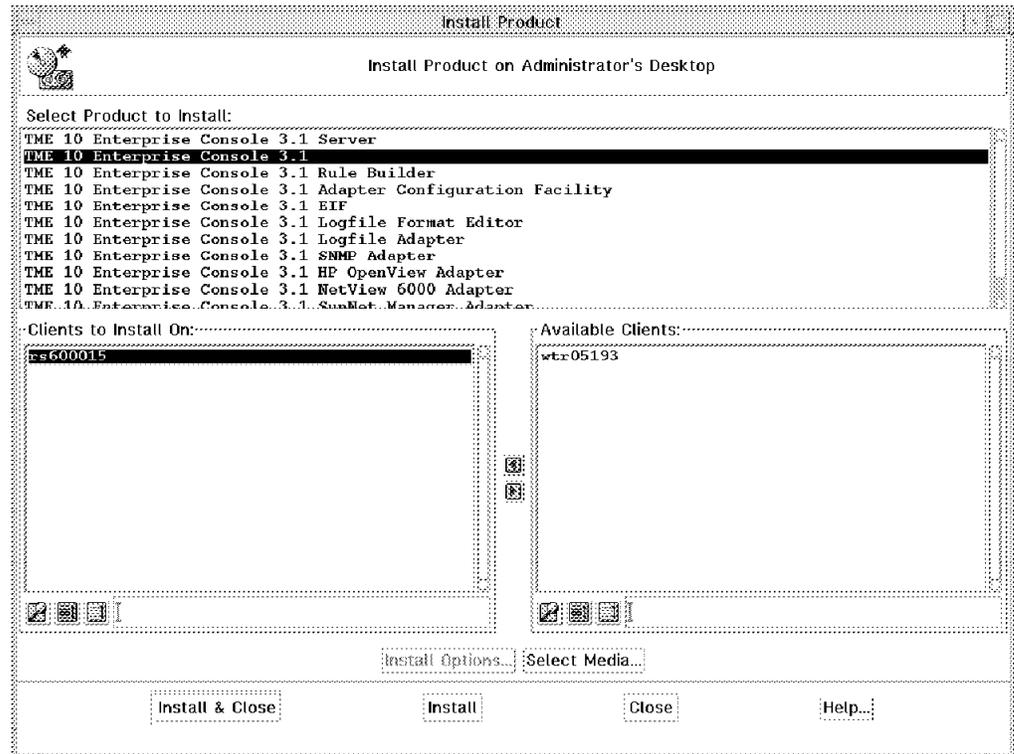
Figure 12. Installation of T/EC

2.2.5 TME 10 Global Enterprise Manager Installation

From the Global Enterprise Manager CD we installed the necessary Global Enterprise Manager Integration Services as well as the Application Policy Manager Server and Client.

2.2.6 TME 10 Event/Automation Service

The first product is the Event/Automation Service that provides the necessary .baroc and .rls files for the the event receiver, message adapter and alert adapter.
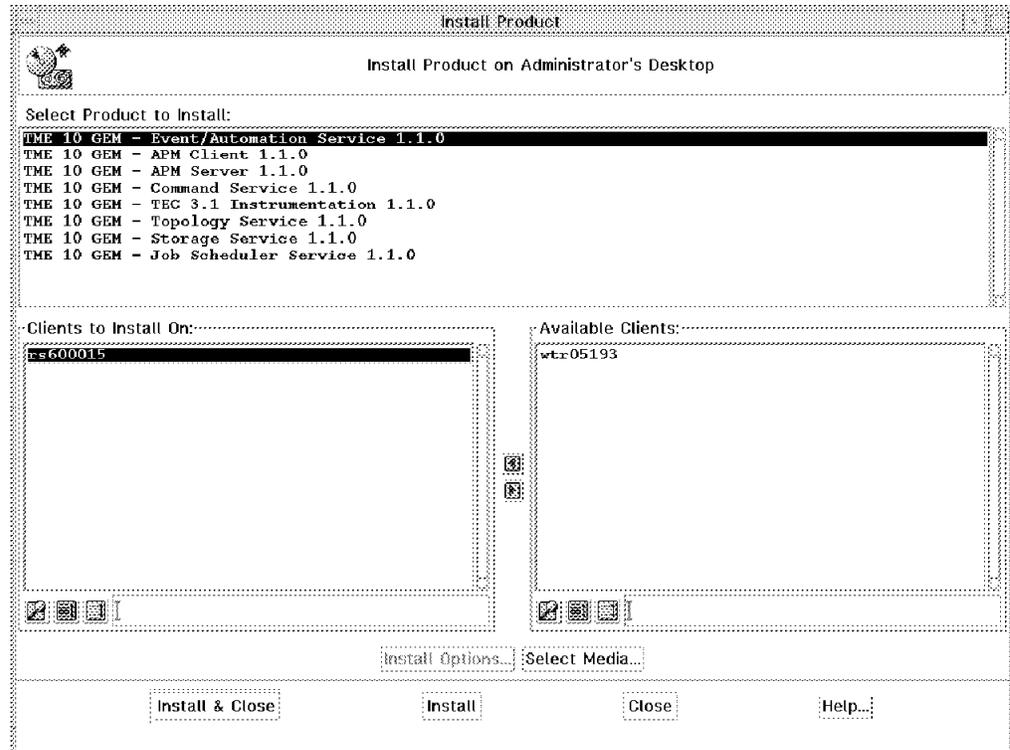
Figure 13. Installation of Global Enterprise Manager Services

2.2.7 TME 10 Topology Service

The next step was to install the Topology Service that provides the MSMAgent code. The Topology Service installs the necessary programs for the MSMAgent to be able to discover and forward topology information to TME 10 NetView for OS/390.

2.2.8 TME 10 APM Server

We then installed the APM server and client, which is a new function that allows you to manage business systems. Since our residency didn't cover Application Policy Manager we installed it as a prerequisite for the command service.

2.2.9 TME 10 Command Service

We installed the Command Service which provides the code required to send commands from the TME environment to TME 10 NetView for OS/390.

2.2.10 TME 10 APM Client

Last we installed the Application Policy Manager Client which provides the new Java interface that is used to manage AMS instrumented applications.

Chapter 3. Installation and Customization of the Topology Service

The Global Enterprise Manager has a new integrated feature, the Topology Service. Using this feature customers are now able to monitor their distributed TME 10 environment from TME 10 NetView for OS/390. In this chapter we show you how to install and customize this feature.

3.1 Introduction to MultiSystem Manager

Prior to TME 10 NetView for OS/390, graphical monitoring of non-SNA resources was provided by a separate product called MultiSystem Manager (MSM). This has now been enhanced and integrated into TME 10 NetView for OS/390 and is called the MultiSystem Manager component of TME 10 NetView for OS/390.

The MultiSystem Manager communicates with agent code that provides information on resources on various platforms to the graphical NetView. These resources can then be monitored by NGMF. Commands can be issued from the NGMF screen and the output can be seen in the NGMF Response window. Alerts and resolve alerts can be received by the NetView host and displayed on NGMF.

MultiSystem Manager works with the following products to gather topology data for the Graphical Management feature:

- NetView for AIX monitoring IP networks

- LNM on OS/2 monitoring LAN networks

- LMU/2 on OS/2 monitoring OS/2 systems and NetWare resources

- NetWare for SAA and NetWare server monitoring a NetWare network

- Open Topology Agents running on OS/2 or AIX allow monitoring on any resource

With the integration of MultiSystem Manager into TME 10 NetView for OS/390, two new monitoring agents have been added to this list:

- NetFinity Manager running on OS/2 monitoring client systems

- There is also a new NetFinity agent running on Windows NT for monitoring of NetFinity clients. This agent is using TCP/IP for communication to TME 10 NetView for OS/390.

- Nways Campus Manager running on NV6000 (AIX) monitoring ATM networks

For more information on the MultiSystem Manager features that were available prior to the integration with TME 10 NetView for OS/390, please review the TME 10 NetView for OS/390 product documentation and the following ITSO redbooks:

- *Managing NetWare Environments from MVS Using NPM, MSM-NetWare,* SG24-4527

- *Managing IP Networks Using NetView MultiSystem Manager R2,* GG24-4337

- *Centralized Management of LNM and NetWare Networks Using NetView MultiSystem Manager MVS/ESA* GG24-4181

With the Global Enterprise Manager, MSMAgent running on a TMR server monitoring TME 10 environments was added.

This agent code acts as the topology agent for all of the TME 10 resources. The communication between the TME 10 NetView for OS/390 as the manager and the MSMAgent is via TCP/IP.

The agent collects resource information and status from the distributed Tivoli environment in logical and physical configurations and forwards these to MultiSystem Manager.

The physical configurations of the TMRs show the distribution hierarchy and status of managed nodes in the distributed network (repeater network).

The logical configuration shows the same resources as in the physical configuration, but organized by policy regions of the TMRs. The managed nodes and all TME 10 Distributed Monitoring monitors that have been distributed to these managed nodes will be included.

3.2 Prerequisites

Since MultiSystem Manager is a component of the TME 10 NetView for OS/390 V1R1, product number 5697-B82, all the prerequisites for the TME 10 NetView for OS/390 also apply for the MultiSystem Manager.

A complete list of the required prerequisites can be found in:

- *TME 10 NetView for OS/390 Installation and Administration Guide*, SC31-8236
- *TME 10 Global Enterprise Manager Installation and User's Guide*, available with the product
- *TME 10 Global Enterprise Manager Topology Service User's Guide*, available with the product
- *TME 10 Global Enterprise Manager Release Notes*

In our environment we installed the following software as prerequisites for the Global Enterprise Manager Topology Service:

- TME 10 Enterprise Console 3.1

  Sybase was used as the database product for T/EC.

  The TME 10 GEM Topology Service can be used with T/EC 2.6.
- TME 10 Global Enterprise Manager Event/Automation Service
- TME 10 Global Enterprise Manager Command Service
- TME 10 Distributed Monitoring 3.5
- TME 10 Distributed Monitoring Monitors
- TME 10 Rule Builder (optional)

3.2.1 MultiSystem Manager Customization

Under the assumption that the base code of TME 10 NetView for OS/390 including the MultiSystem Manager component of TME 10 NetView for OS/390 is installed and customized we have to do the following tasks to get the TME 10 feature of MultiSystem Manager up and running:

- Customize the automation table for the TME 10 MultiSystem Manager feature
- Set up the initialization member for the TME 10 MultiSystem Manager feature

- Define the exception views for the TME 10 MultiSystem Manager feature:
  - Define or extend the FLCEXV member
  - Define and load the appropriate part of the data model for RODM
  - Customize the DUIFSSMT (exception view mapping table) and assemble and link it
- Download the NGMF workstation code and add the MultiSystem Manager command set

3.2.1.1 Automation Table Customization

If you are using DSITBL01 as your automation table, uncomment the FLCSTBLT include statement:

```
*************************************************************************
* Uncomment the following statement if you are running the TME        *
* feature of MultiSystem Manager.                                     *
*************************************************************************
%INCLUDE FLCSTBLT
*
*
*************************************************************************
*                                                                     *
*    UNCOMMENT THE NEXT AUTOMATION STATEMENT TO AUTOMATE ALERTS AND    *
*    RESOLUTIONS FOR GMFHS.                                           *
*                                                                     *
*************************************************************************
IF (MSUSEG(0000) ¬= '' | MSUSEG(0002) ¬= '') & HIER ¬= ''
  THEN EXEC(CMD('DUIFECMV') ROUTE(ONE DUIFEAUT)) CONTINUE(Y);
*                                                                     *
*                                                                     *
```

The included FLCSTBLT member shown below contains the necessary automation table entries.

```
**********************************************************************
* Act upon the alerts/resolves from TME-10 T/EC Event Server that   *
* are from Sentry.  The T/EC Event Server alerts are demarcated by  *
* the last bit in the first byte of subvector 92.  For Sentry       *
* events, HIER(1) will be set to SENTRY.                            *
**********************************************************************

IF (MSUSEG(0000.92 3 8) = '1' | MSUSEG(0002.92 3 8) = '1') &
   HIER(1) = 'SENTRY' .
THEN
  EXEC(CMD('FLCATAUT') ROUTE(ONE AUTOTMEA))
  EXEC(CMD('FLCATALH') ROUTE(ONE AUTOMSM))
  CONTINUE(N);
```

The FLCSTBLT member includes calls to two different CLISTs:

- FLCATAUT
- FLCATALH

The FLCATAUT is used to build the objects in RODM for the MultiSystem Manager Agent and Tivoli resources and takes care of the MultiSystem Manager

Agent status. The FLCATALH is used to change the status fields of the TME 10 objects in RODM based on the alerts coming in via the event adapter.

Keep the TME 10 automation entries above the entry for alerts going to GMFHS to avoid unnecessary calls. The alerts coming from the TME 10 resources are not handled by GMFHS.

3.2.1.2 MultiSystem Manager Initialization Members

There is a new MultiSystem Manager initialization member for the Global Enterprise Manager Topology Service that has to be customized. The member FLCSITME is located in the SFLCSAMP library and has to be copied to your DSIPARM data set. This member is used to define your TME service point where the MSM topology agent is installed. In addition there are some other keywords that have to be defined. For a more detailed explanation see the description in the sample file A.2.1, "FLCSITME Used for the TME Feature of MultiSystem Manager" on page 138 and *Global Enterprise Manager Topology Service User's Guide*, available with the product.

 We used the following customization for our setup:

```
*************************************************************************
GETTOPO TMERES,
SP=RS600015,
POLICY=YES,
MONITORS=YES,
PORT=3333,
NETWORK_AG_OBJECT=TME10-first-test,
NETWORK_NAME=RALEIGH_TMR_TEST,
NETWORK_VIEW=TME10_VIEW/TEST1
*************************************************************************
```

Figure 14. MultiSystem Manager Initialization Member FLCSITME

The FLCSITME member will be included, using the %include statement, in the main MultiSystem Manager initialization member FLCAINP. We used the following definitions in our FLCAINP setup:

```
DEF_AUTOTASK=AUTOMSMD
RODMNAME=RODM1
EXCEPTION_VIEW_FILE=FLCSEXV
*************************************************************************
* Include the initialization statements for TME service points.      *
*************************************************************************
%INCLUDE FLCSITME
```

Figure 15. MultiSystem Manager Initialization Member FLCAINP

For a more detailed explanation see the *Global Enterprise Manager Topology Service User's Guide*, available with the product.

Make sure you have uncommented the following statement in the CNMS1048 member of your DSIPARM data set:

    %INCLUDE FLCS1048

The member FLCS1048 is in data set SFLCSAMP and provides the mapping between MultiSystem Manager commands and the help panels.

3.2.1.3 MultiSystem Manager Exception Views

The exception view is a very useful way to recognize failing resources and put them in a separate view. This separate view has to be created in RODM. The FLCSDM6T member in data set SFLCSAMP is a sample on how to define the exception view object in RODM.

We used the following definitions in our FLCSDM6T:

```
CREATE INVOKER  ::= 0000001;
       OBJCLASS ::= Exception_View_Class;
       OBJINST  ::= MyName = (CHARVAR) 'TME10_Monitors';
       ATTRLIST
       Annotation ::= (CHARVAR) 'TME10 Sentry Monitors',
       ExceptionViewName ::= (CHARVAR) 'TME10MON';
END;
```

Figure 16. FLCSDM6T Exception View Object in RODM

The next step will be to tell MultiSystem Manager which objects of which classes are suitable for exception view processing. To do this you have to create a member in the DSIPARM data set with the appropriate definitions. The member name of this file has to map the value of the Exception_View_File= definition in your FLCACINP member. A sample member of FLCSEXV is provided in the SFLCSAMP data set.

For our tests we created a member FLCSEXV with the following definitions:

```
 * agent
 OBJECTCL=1.3.18.0.0.3315.0.3.1
      EXVWNAME=TME10_Monitors
 OBJECTCL=Monitor
    EXVWNAME=TME10_Monitors
```

Figure 17. MultiSystem Manager Member FLCSEXV for Exception Views

Objects that are stored under the class defined in OBJECTCL are eligible for exception view processing. If an exception occurs for an object, it will be connected to the object defined under EXVWNAME. This definition has to map the Myname field for the exception view defined in member FLCSDM6T.

Now that we have defined the exception view objects and which objects under which classes are eligible for this process we have to decide what is an exception condition.

The member DUIFSMT in the CNMSAMP data set defines the standard exception provided with TME 10 NetView for OS/390. If you would like to create your own exception views, you have to modify this member. MultiSystem Manager provides a sample member FLCSSMT in the SFLCSAMP data set.

The following definitions were used in our environment:

```
 *--    agent
           DUIFSMTE CLASS=1.3.18.0.0.3315.0.3.1,                              C
                   XCPT=(UNSAT)
 *--    monitors
           DUIFSMTE CLASS=Monitor,                                           C
                   XCPT=(UNSAT)
```

Figure 18. MultiSystem Manager Member FLCSSMT for Exception Views

Under the class statement you define the class of your eligible objects and under
XCPT you define the exception situation. Like in our case all objects with an
unsatisfactory status should be in the exception views. The class definition here
has to map your class definition done in the FLCSEXV member.

The above statements have to be included in the DUIFSMT member that then
has to be compiled and linked into your user method link library. We used the
JCL found in Figure 136 on page 137 to do this. Some points that you need to
be aware of about this JCL are:

- The job also includes an RODM loader step to activate the changes made to
  the status mapping definition.

- With OS/390 R3 the old assembler compiler and linkage editor are replaced
  by a new assembler compiler ASMA90 and a new linkage editor HEWL.
  Please refer to the TME 10 NetView for OS/390 PSP bucket for more
  information.

- Be aware that the SYSLIB DD statement in the LKED step should point to
  your LE/370 library if you have installed the LE/370 version of TME 10
  NetView for OS/390. We used the LE/370 version of TME 10 NetView for
  OS/390 but the product sample CNMSJH13 pointed to the C/370 library.

3.2.1.4 NGMF Workstation Code
There are no changes to the download process of the workstation code to
previous releases. You will find a detailed description of how to install the
MultiSystem Manager part of the workstation code in *Global Enterprise Manager
Topology Service User′s Guide*, available with the product, and a step-by-step
approach in the appendix ″OS/2 Access feature and Personal Communications″
of *An Introduction to TME 10 NetView for OS/390*, SG24-4922.

3.3 Customizing the Topology Services
After installing the base and product code you have to customize the different
components to fit your environment. The following paragraphs give you a
detailed description on how to do this task.

3.3.1 Customizing Enterprise Console
At this point we are ready to define consoles and load rules.

3.3.1.1 Components of the Enterprise Console

The Enterprise Console uses distributed event adapters to collect information, a central event server to process the information, and distributed event consoles to present the information.

The following is a high-level description of the various resources you will need to know when working with the Tivoli Enterprise Console:

- Events: Is the unit of information. An event means that a change in the state of a system resource has occurred.

- Event Adapter: Is a process that gathers information from a source (typically an application or a system resource), formats the information and forwards it to the event server.

- Event Server: Is the central database that handles all events from the distributed systems. The event server takes the event, creates an entry in the database for it, evaluates the event with a set of rules, and then sends the event to the event console.

- Event Console: Is a graphical user interface (GUI) that allows operators to manage the events.

- Baroc Files: Defines classes of events to the event server.

- Rules Files: Defines automated actions to be taken against an incoming event, or against a change of the status of an event. The rules are used for starting automated actions and correlation of events.

3.3.1.2 Loading a Rule Base

When TME 10 Enterprise Console in installed, we get a new icon on the Desktop, called EventServer (see Figure 19).
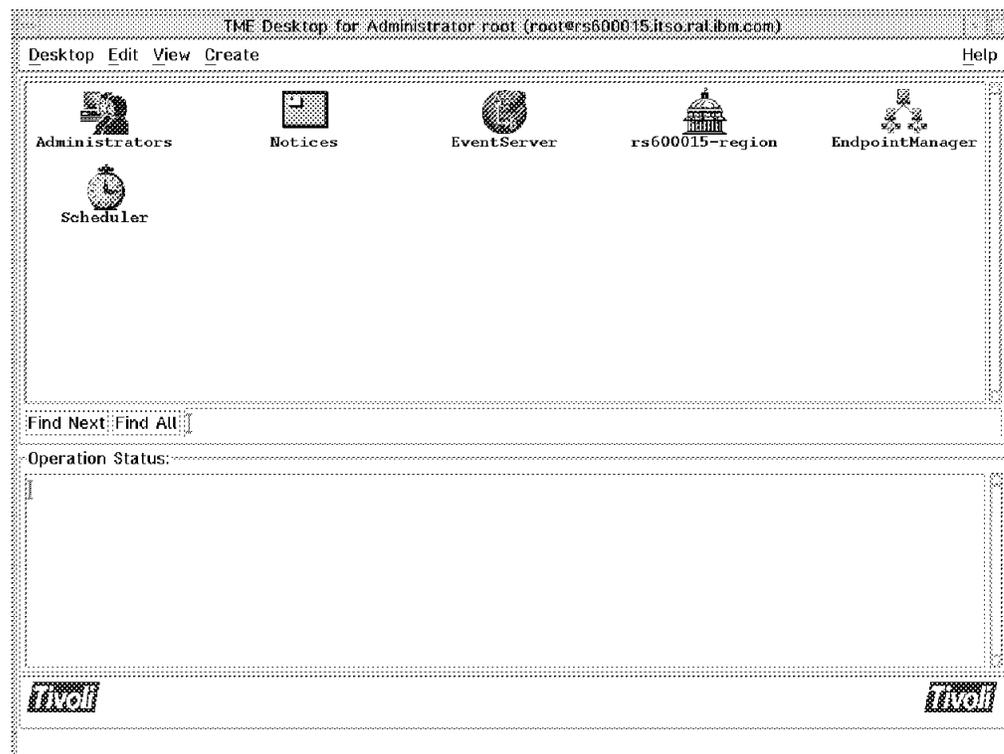
Figure 19. Event Server Icon on Desktop

There is a set of rules and baroc files that must be loaded in order to get events properly from the different sources.

The rules and baroc files for each product are located in several directories on the disk. TME 10 Global Enterprise Manager provides a sample shell script to load all those baroc and rules files into the EventServer. This script is named gemevent.sh. This sample shell script is located in $BINDIR/GEM/EventService.

This shell script installs object classes and rules for Event Service, Storage Service, Topology Service and Application Policy Manager. But in order to have Monitors for NT the NT baroc files must be added manually. The gemevnt.sh and a modified version of this script are provided in B.4, "Shell Scripts Used" on page 175.

The shell script also creates event groups and event sources for TME 10 Global Enterprise Manager, creates a UNIX user and the associated Tivoli administrator with the permissions to manage specific events groups for Global Enterprise Manager.

Edit the file and enter the appropriate information for the following variables:

1. OS390id = The IP name of the mainframe that will receive the events sent by T/EC
2. APMServer = The name of the APM server

In our environment we loaded the following baroc files:

1. tecad_nv390msg.baroc
2. tecad_snaevent.baroc
3. Sentry_baroc
4. tivoli_baroc
5. universal.baroc
6. MSMAgent.baroc
7. interapp.baroc
8. All the baroc files for NT monitors

We loaded the following rule files:

1. tecad_nv390msg.rls
2. tecad_snaevent.rls
3. tecad_nv390fwd.rls
4. forwardSentry.rls
5. interapp.rls

Note: The order in which the rule files are loaded is very important.

Run the script, and check for errors before continuing.

The script creates a rule base in the EventServer called GemEvent.

3.3.1.3 Creating a Console

If you have run gemevent.sh at this point, you have an event console defined for the GemAdmin user. In order to create a console for any other user, you must follow the next steps. To create a console, the user must have a senior role in the TMR Roles.

To give a user the senior role, the following steps can be performed:

1. Open the **Administrators** icon on your Desktop.

2. Select with the left mouse button the administrator for whom you want to add the senior role. Click on the right mouse button and select **Edit TMR Roles** from the menu.

Figure 20. Editing the TMR Roles

3. Fill in the Current Roles list by selecting the appropriate role from the Available Roles box.
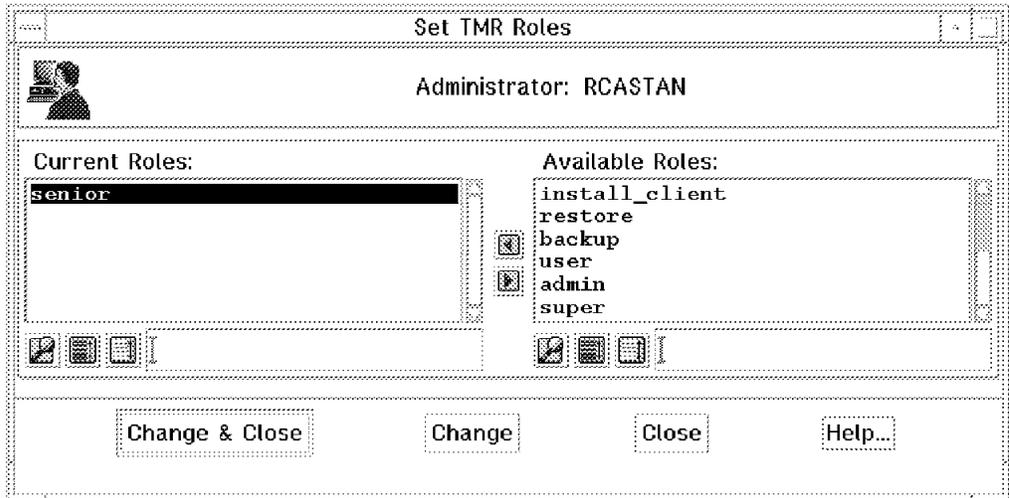
Figure 21. Setting the Senior Role

4. Click on **Change & Close** button, to set the role and exit.

The following steps can be used to create the event console for the user:

1. Select **Create** from the TME Desktop menu bar.
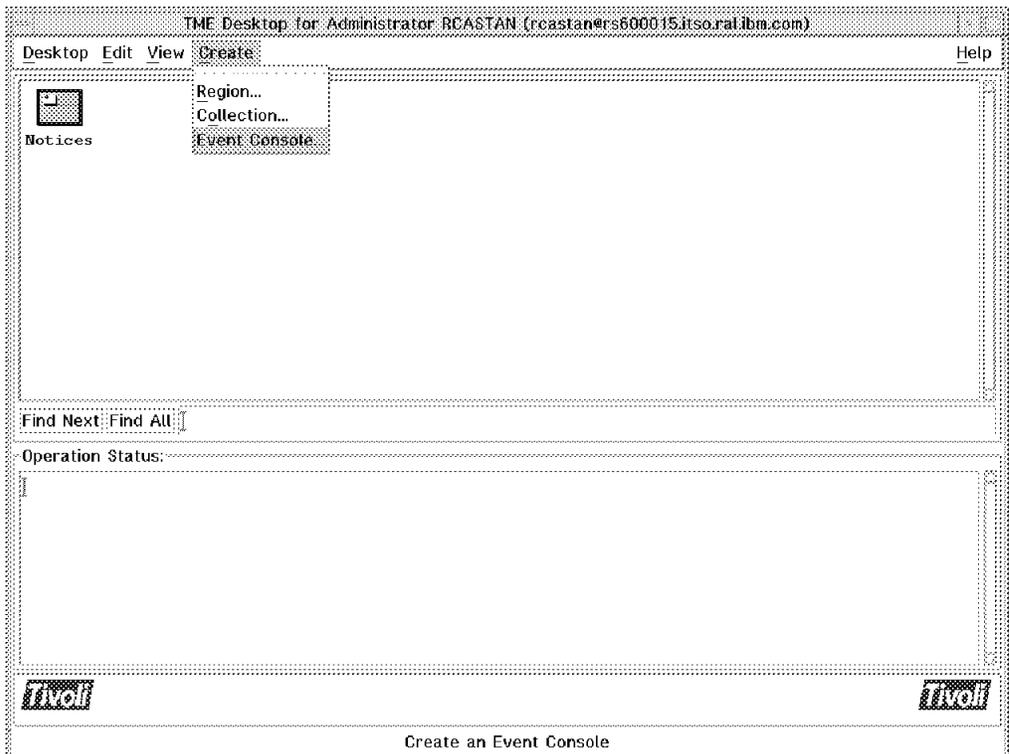
Figure 22. Creating an Event Console

2. Select **Event Console**. This will display the Create Event Console dialog.

```
┌─────────────────────────────────────────────────────────────────┐
│ ▦▦▦▦▦▦▦▦▦▦▦▦▦▦ Create Event Console ▦▦▦▦▦▦▦▦▦▦▦▦▦ │ · │▦│
├─────────────────────────────────────────────────────────────────┤
│  ⬤   Create Event Console for Administrator  RCASTAN            │
│                                                                   │
│                              Available Hosts                      │
│                        ┌──────────────────────────────┐ ▲        │
│                        │ foxtrot                      │ ░        │
│    Select Host where   │ ▓rs600015▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓│ ░        │
│    Console will execute:│ wtr05193                    │ ░        │
│                        │                              │ ░        │
│                        └──────────────────────────────┘ ▼        │
│                        ┌──┐┌──────────────────────────┐           │
│                        │▨ ││                          │           │
│                        └──┘└──────────────────────────┘           │
│                                                                   │
│       ┌────────┐        ┌────────┐        ┌────────┐             │
│       │ Create │        │ Cancel │        │ Help...│             │
│       └────────┘        └────────┘        └────────┘             │
└─────────────────────────────────────────────────────────────────┘
```
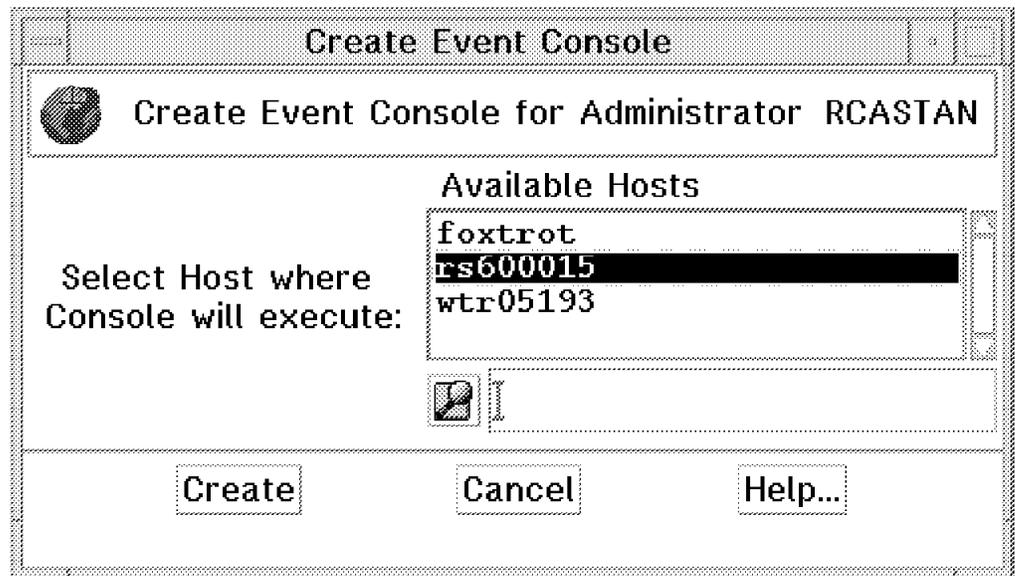
Figure 23. Event Console Dialog

3. Select from the list of hosts of the TMR the one where the console will run.

4. Click on the **Create** button to create the console and exit the dialog.

At this point we have a new icon for the console on the Desktop. The next step is to assign event groups to the event console. In order to do this, the user must have a super role.

1. Click with the right mouse button on the Event Console window and from the menu select **Assign Event Groups**.
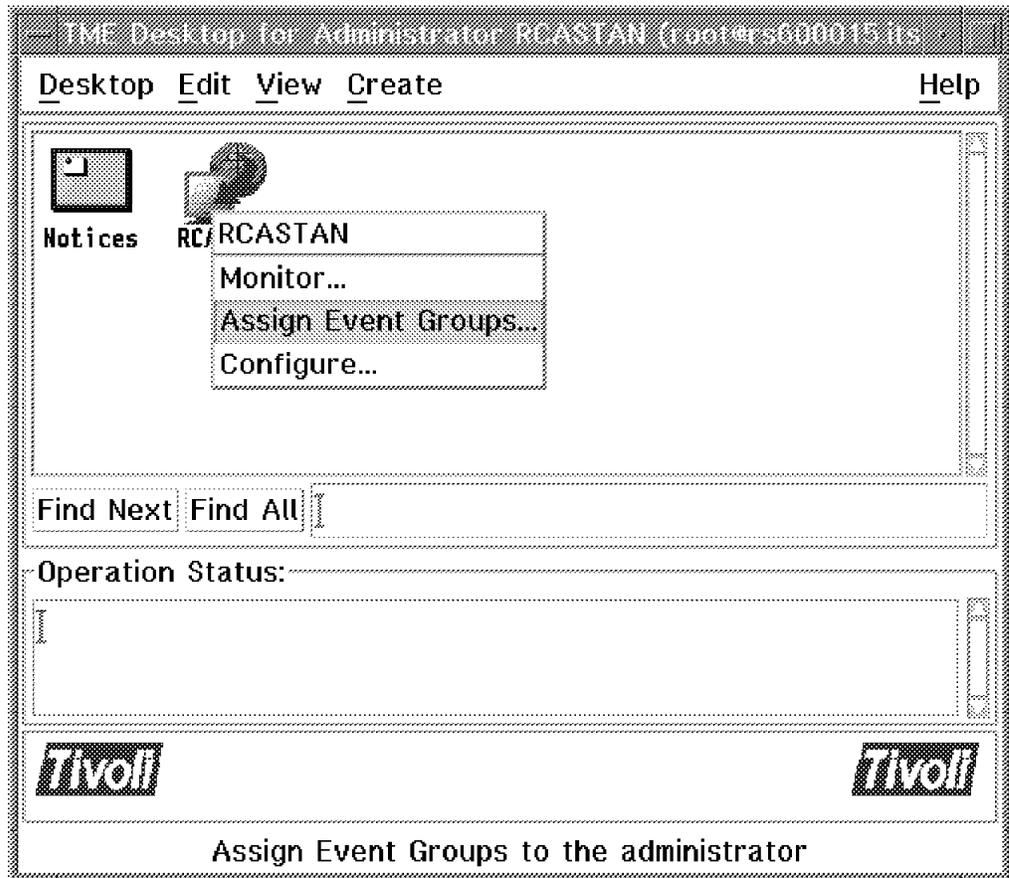
Figure 24. Assign Event Groups

2. In the Assign Event Groups dialog choose one or more event groups from those listed in Unassigned Event Groups, and move them with the right arrow to the Assigned Event Groups window.

3. Select an event group from the Assigned Event Groups, and assign the necessary roles.
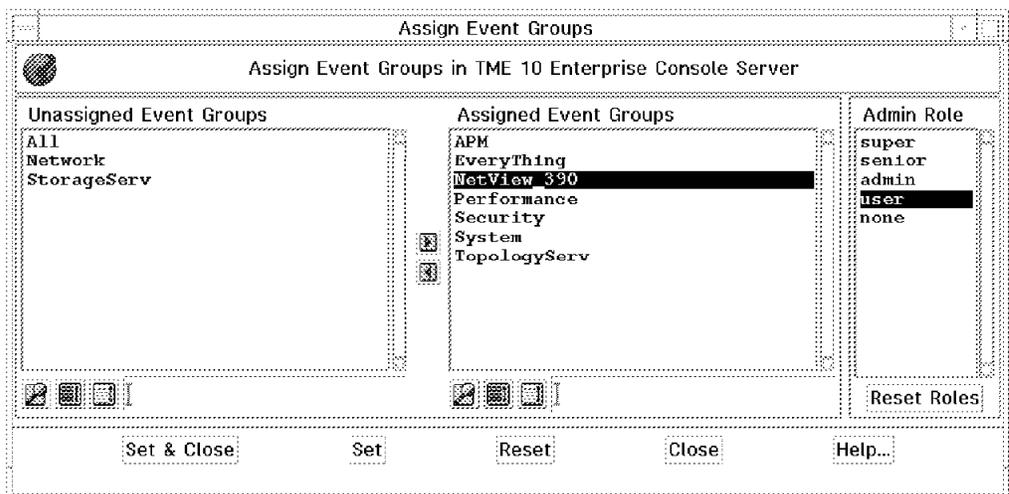
Figure 25. Assigning Event Groups to the Console

4. Click on the **Set** button to save all roles assignments for an event group. Repeat this operation for each event group.

5. Click on the **Set & Close** button to save and exit.

The last step is to select the event server location from the available TMRs:

1. Click with the right mouse button on the Event Console window and from the menu, select **Configure**.

2. In the Server Location dialog select the appropriate TMR in the scrolling list.

Figure 26. Server Location Dialog

3. Click on the **Set & Close** button to save and exit.

Now we are able to check that the console is loaded and operational:

1. Start the event server by selecting **Start-up** from the EventServer icon menu.

2. Check for the message `TME 10 Enterprise Console Server successfully started` in the Operation Status window.

3. Open the console by double-clicking on the icon. The TME 10 displays the Event Sources and the Event Groups dialogs:
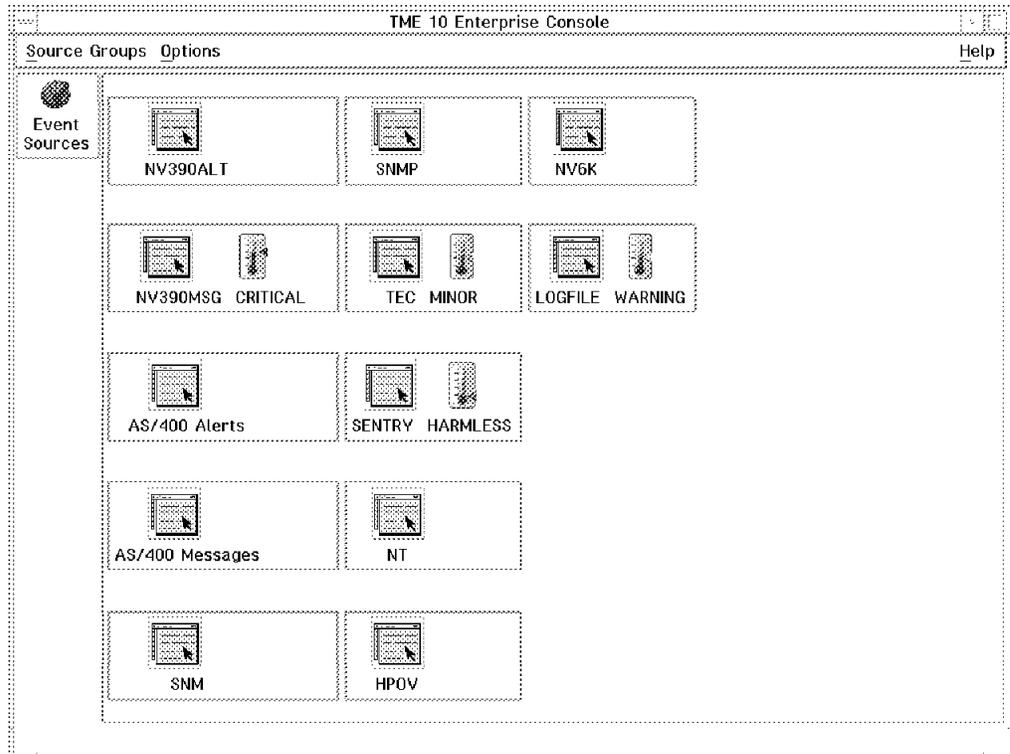
Figure 27. Event Console Source View
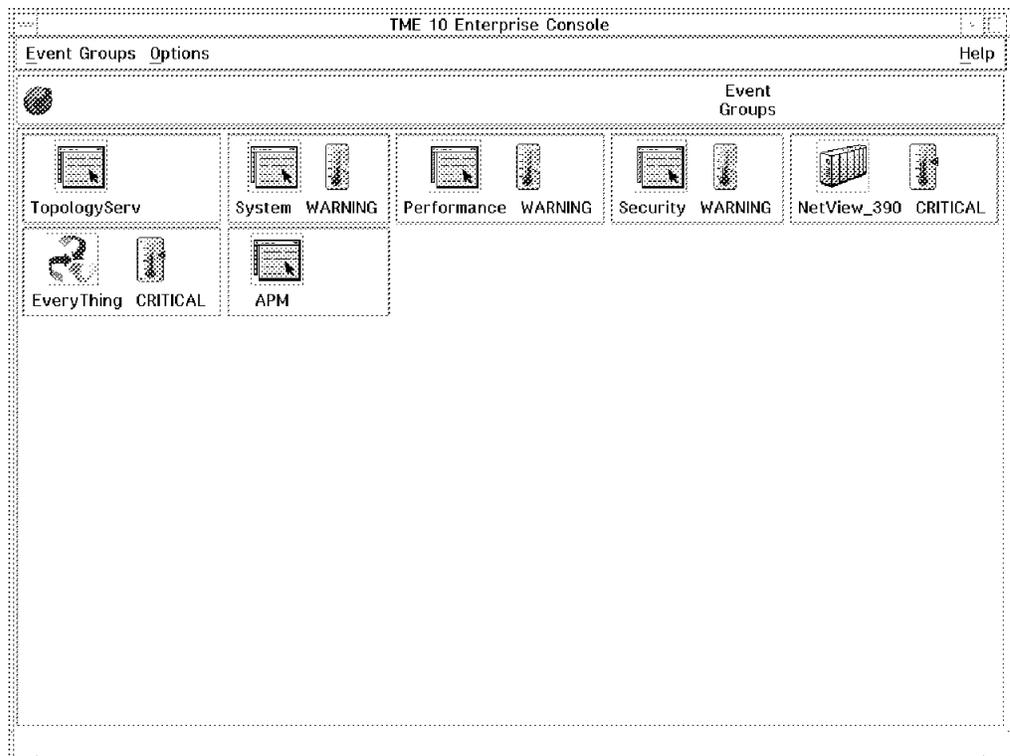
Figure 28. Event Console Groups View

3.3.2 Customizing TME 10 Distributed Monitoring

This customization can be divided in two parts:

1. Assigning Distributed Monitoring Managed Resources to a Policy Region

    • TME 10 Distributed Monitoring Managed Resources Customization

2. Creating an environment in which monitors can be created and distributed to defined subscribers:

    • Create Profile Manager

    • Create Profile

    • Define Subscribers to a Profile Manager

3.3.2.1 Assign Managed Resources to a Policy Region

On the desktop, double-click on the policy region icon or click with the right mouse button on the icon and select **Managed Resources** from the menu.

In our example we double-clicked on the policy region icon and then selected **Properties** on the menu bar. Then we selected **Managed Resources**.
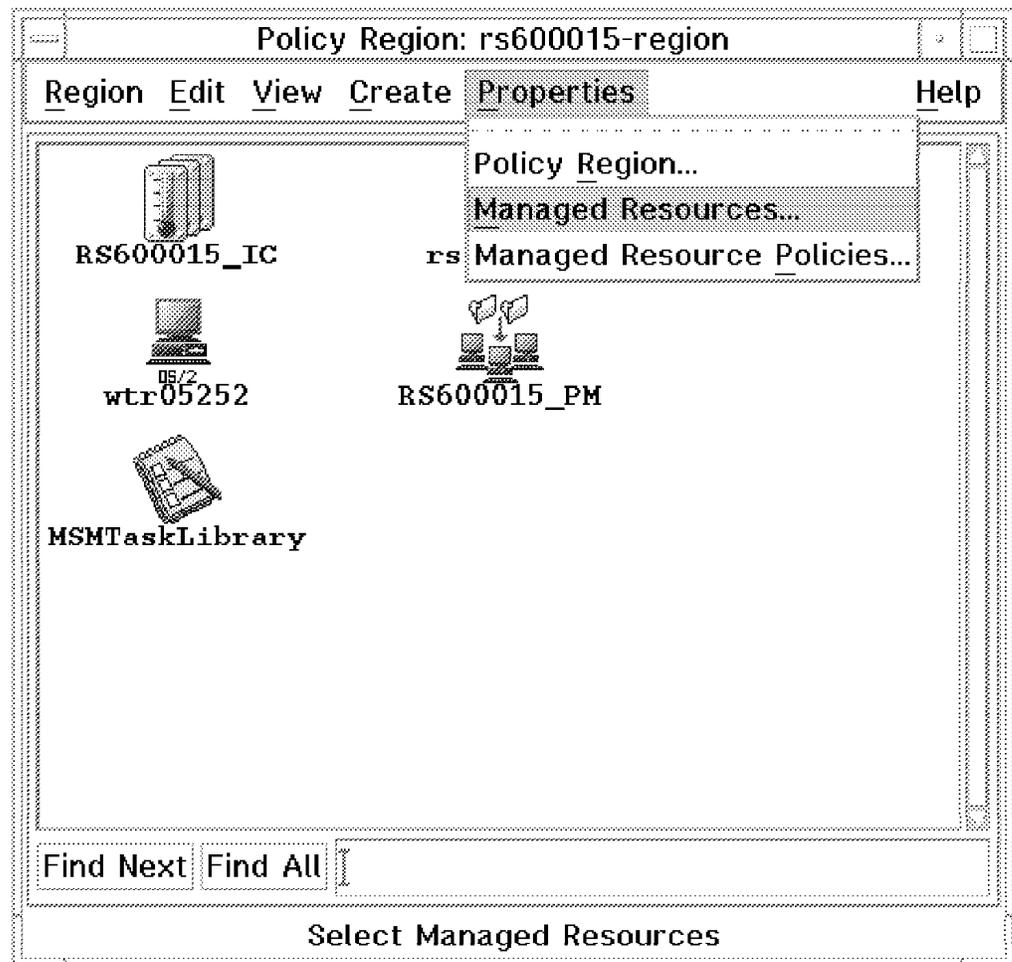
Figure 29. Managed Resources Selection

You should select the **IndicatorCollection, SentryProfile** and **SentryProxy** options from the Available Resources scrolling list and click the left-arrow button to move these managed resources to the Current Resources scrolling list.

This operation requires TME Admin role of senior.

Figure 30. Assign Distributed Monitoring Resources As Managed Resources

Click the **Set & Close** button to add these resources to a policy region and close the Set Managed Resources dialog.

For more information about adding managed resources to a policy region from the command line, refer to the wsetpr command in the *TME 10 Framework Reference Guide*.

3.3.2.2 Create Profile Manager

The second part of customizing TME 10 Distributed Monitoring is to create a profile manager in which you can store your TME 10 Distributed Monitoring profiles. Subscribers can then be defined to the profile manager allowing distribution of any profile defined in that profile manager to a managed node. The first step is to create the profile manager.

From the policy region menu, click on **Create**, and select **Profile Manager**.
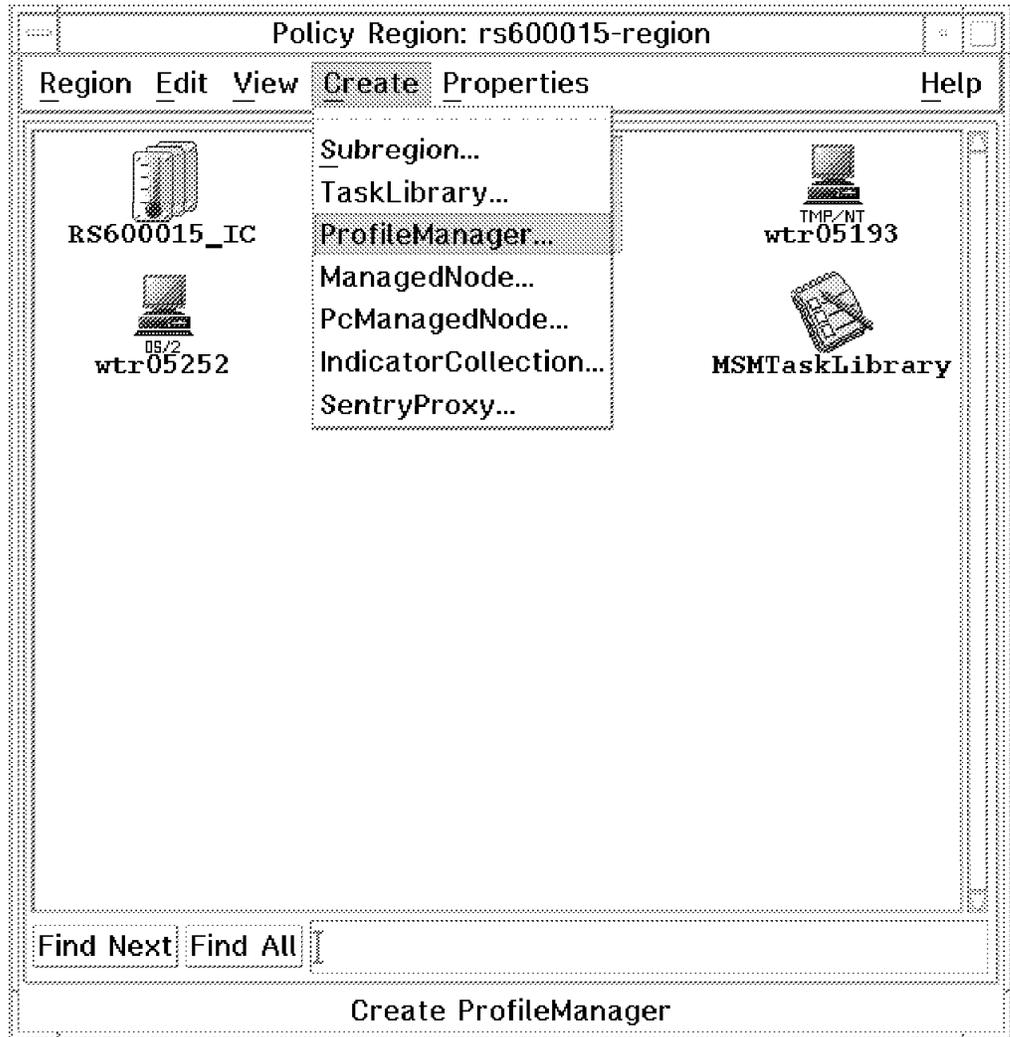
Figure 31. Create Profile Manager Selection

The Create Profile Manager window will appear (see Figure 32 on page 34).
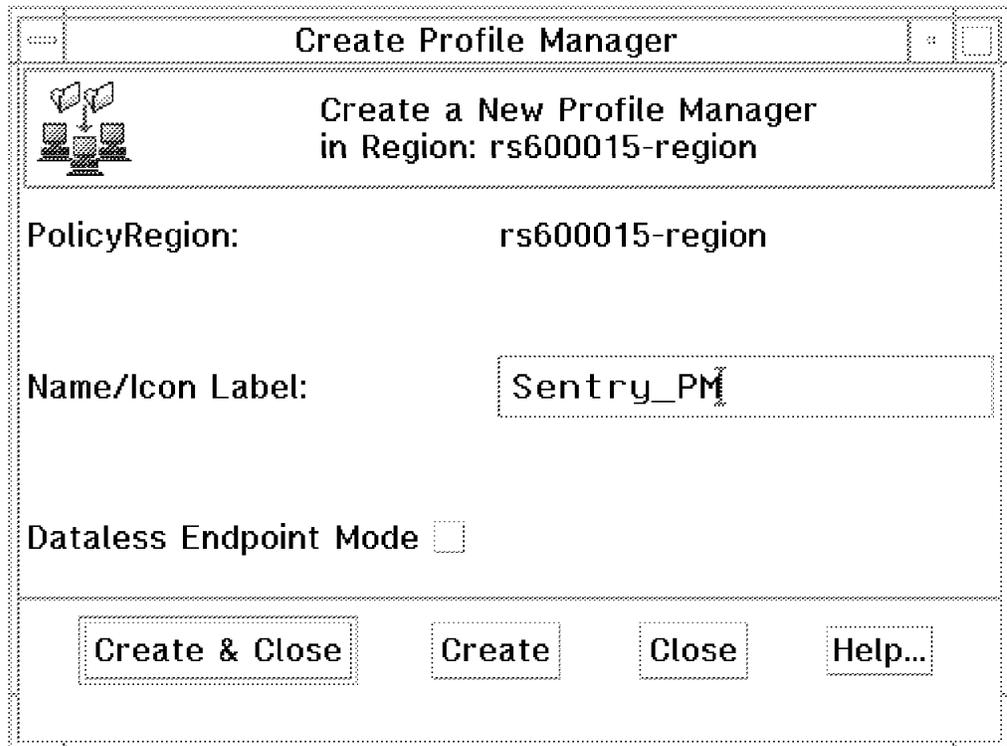
Figure 32. Create Profile Manager Dialog

Fill in the Name/Icon Label with the name of the Profile Manager and click on the **Create & Close** button.

3.3.2.3 Create Profile
With the profile manager created we now create the profile that will contain the TME 10 Distributed Monitoring monitors.

From the policy region, double-click on the profile manager icon. TME displays the Profile Manager window (see Figure 34 on page 36).
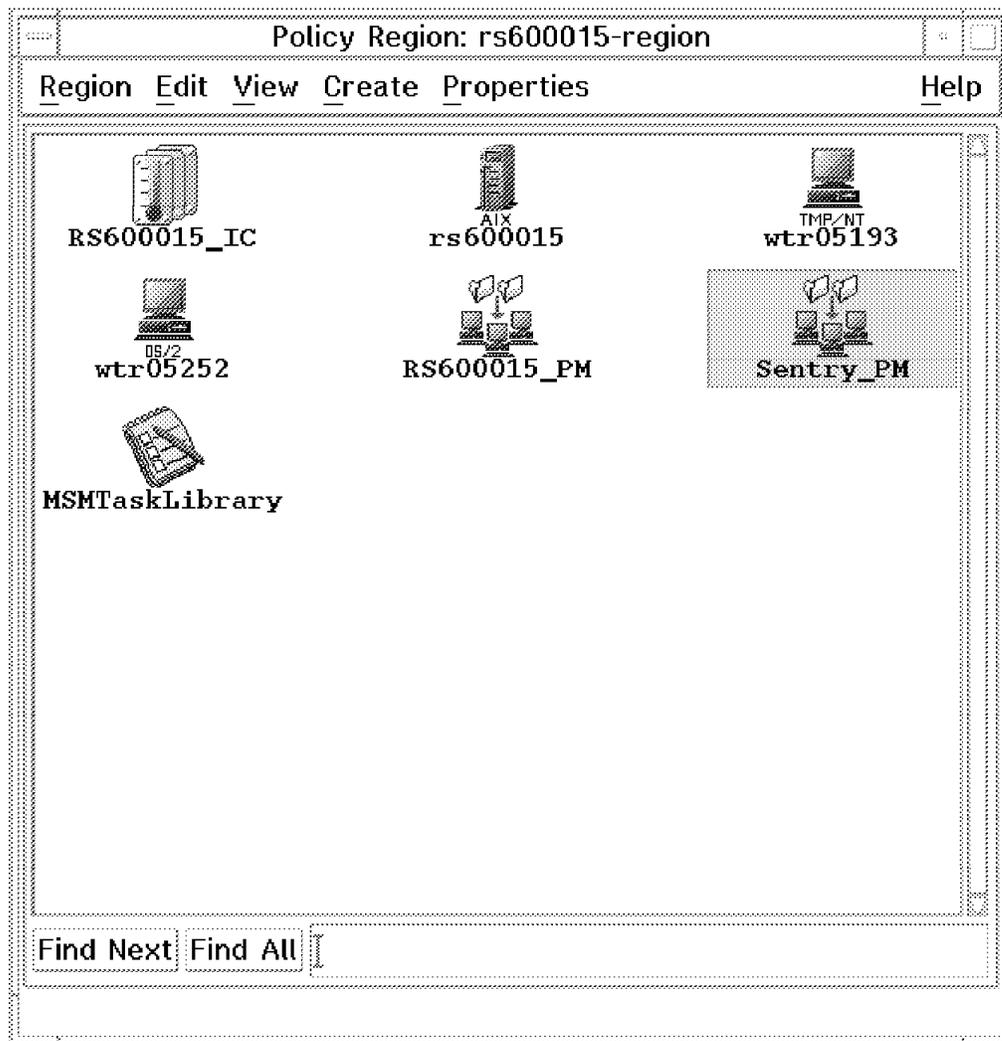
Figure 33. Profile Manager Selection

You can also click on the profile manager icon with the right mouse button and select **Open....** This will give the same result.

This operation and onwards requires TME Admin role of admin.
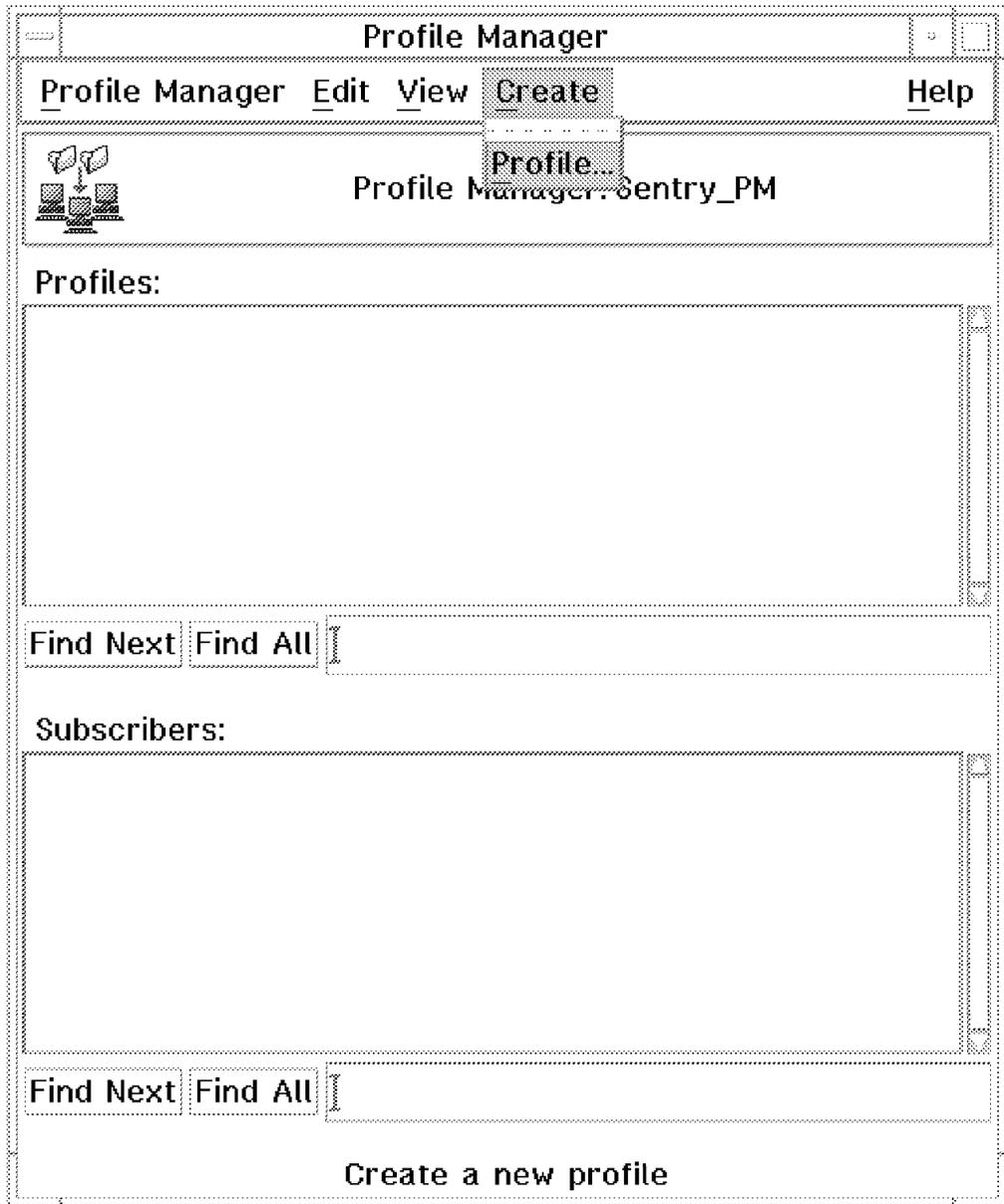
Figure 34. Create Profile Selection

Select **Profile...** from the Create menu to display the Create Profile dialog (see Figure 35 on page 37).

```
 ┌─────────────────────────────────────────────────────┐
 │ ──    │           Create Profile                       │
 ├─────────────────────────────────────────────────────┤
 │  ┌────┐      Create a New Profile                      │
 │  │icon│   in Profile Manager: Sentry_PM                │
 │  └────┘                                                │
 ├─────────────────────────────────────────────────────┤
 │  Name/Icon Label: │ SentryProf              │         │
 │                                                        │
 │  Type:                                                 │
 │  ┌─────────────────────────────────────────────────┐  │
 │  │           SentryProfile                         │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  │                                                 │  │
 │  └─────────────────────────────────────────────────┘  │
 │                                                        │
 │  ┌──────────────┐  ┌────────┐  ┌───────┐  ┌───────┐   │
 │  │ Create & Close│  │ Create │  │ Close │  │ Help… │   │
 │  └──────────────┘  └────────┘  └───────┘  └───────┘   │
 └─────────────────────────────────────────────────────┘
```
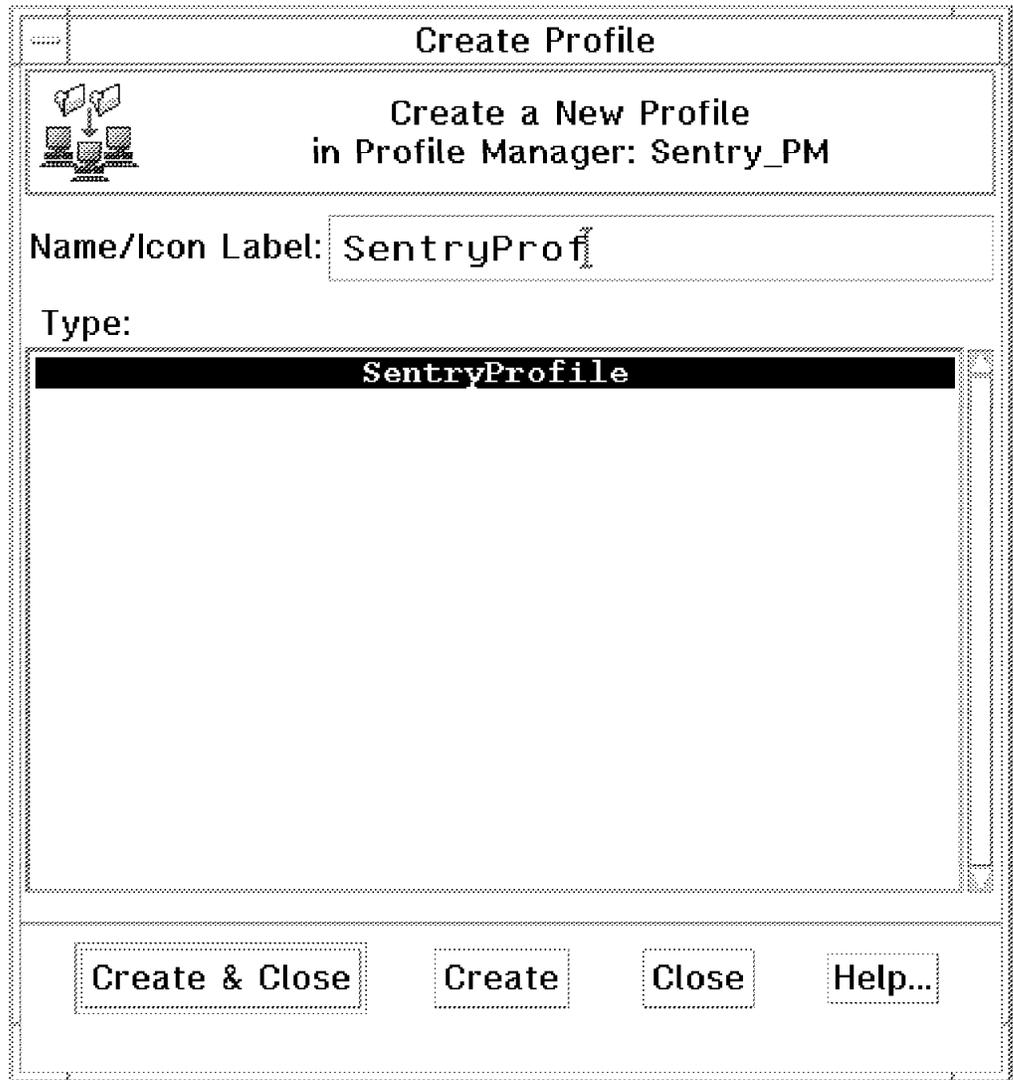
Figure 35. Create Profile Dialog

Enter a unique name for the profile in the Name/Icon Label field.

Select the **SentryProfile** option from the Type scrolling list.

Click the **Create & Close** button to create the profile and return to the Profile Manager window. An icon representing the newly created TME 10 Distributed Monitoring profile is displayed in the Profile Manager window (see Figure 36 on page 38).

Figure 36. Profile Manager Dialog

For more information about creating a TME 10 Distributed Monitoring profile in a profile manager from the command line, refer to the wcrtprf command in the *TME 10 Framework Reference Guide.*

3.3.2.4 Define Subscribers to a Profile Manager

Defining subscribers to a profile manager will allow you to distribute any of the profiles defined in the profile manager to the managed node.

To add one or more subscribers to a profile manager, from the Profile Manager window, select **Subscribers...** from the Profile Manager pull-down menu. This will display the Subscribers dialog (see Figure 38 on page 40).
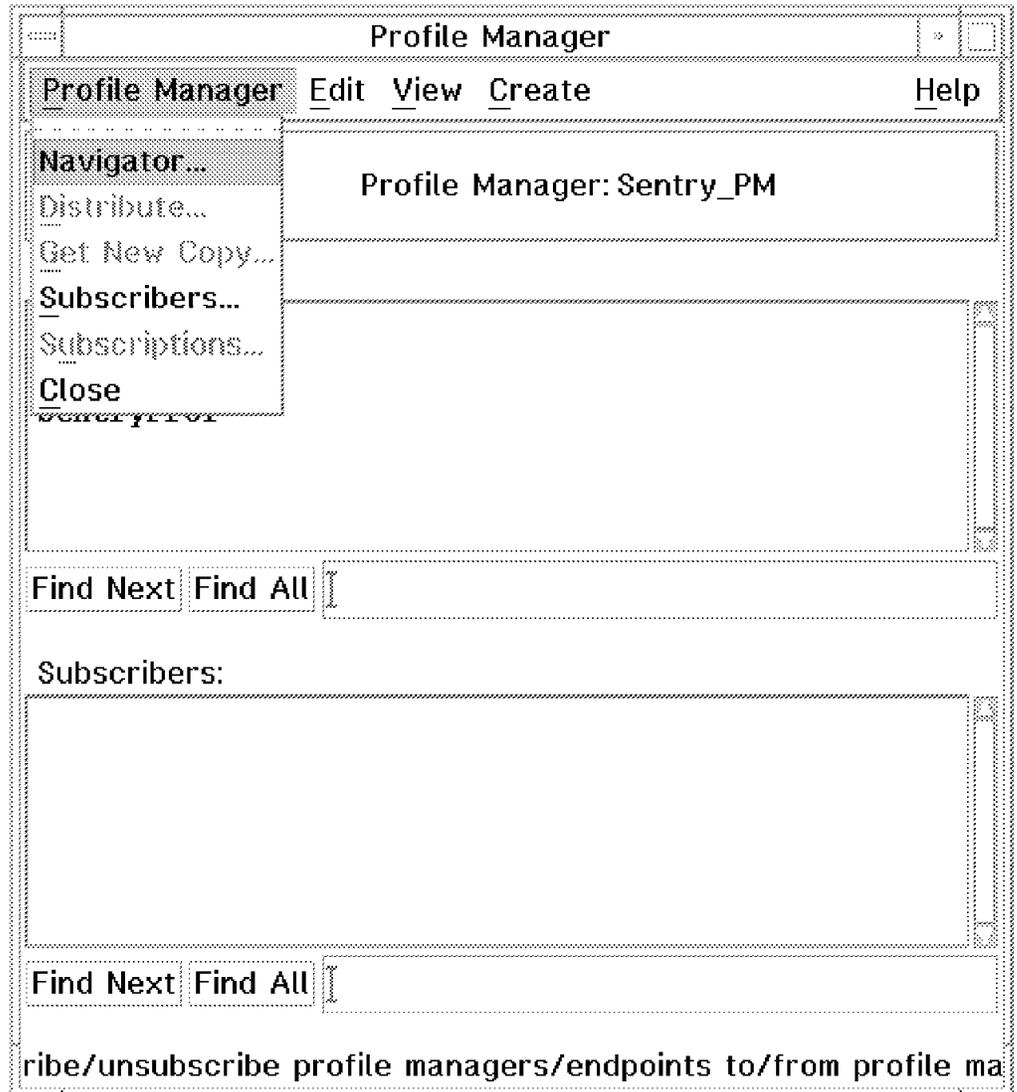
Figure 37. Profile Manager Subscribers Selection
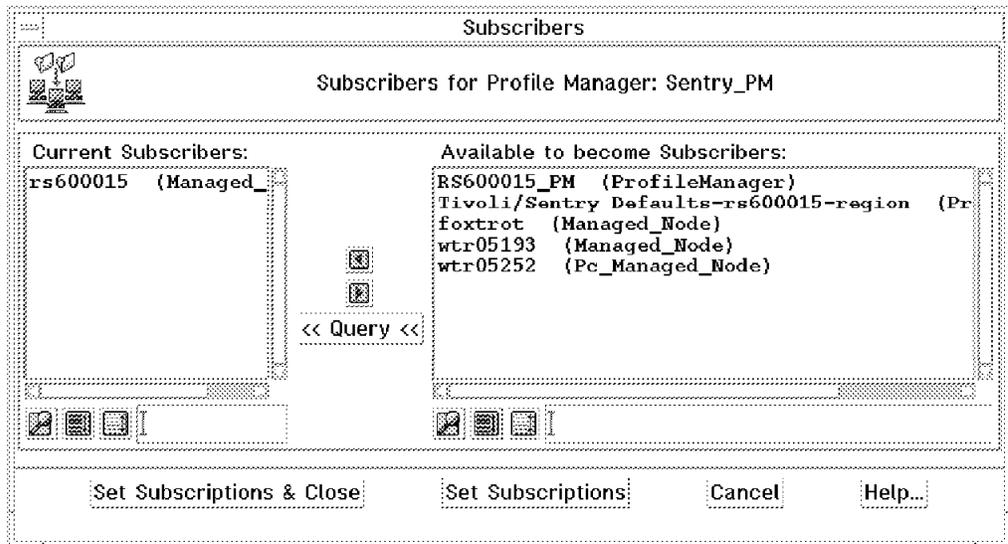
Figure 38. Subscribers Dialog

This dialog contains a list of all profile managers and endpoints that can subscribe to the current profile manager.

Select one or more subscribers from the Available to become Subscribers scrolling list and click the left-arrow button to move your selections from the available list to the Current Subscribers scrolling list.

Click on the **Set Subscriptions & Close** button to add the subscribers to the profile manager and dismiss the Subscribers dialog. Icons for any new subscribers are displayed in the lower half of the Profile Manager window.

Note: You must distribute the profile to establish the subscriber relationship and update subscribers system files. We explain how to distribute a TME 10 Distributed Monitoring profile after we have defined a monitor.

For more information about adding subscribers to a policy manager from the command line, refer to the wsub command in the *TME 10 Framework Reference Guide.*

3.3.2.5 Adding a Monitor to a TME 10 Distributed Monitoring Profile

Use the following steps to add a monitor to a TME 10 Distributed Monitoring profile.

From a Profile Manager window, double-click on a TME 10 Distributed Monitoring profile icon to display the TME 10 Distributed Monitoring Profile Properties window.
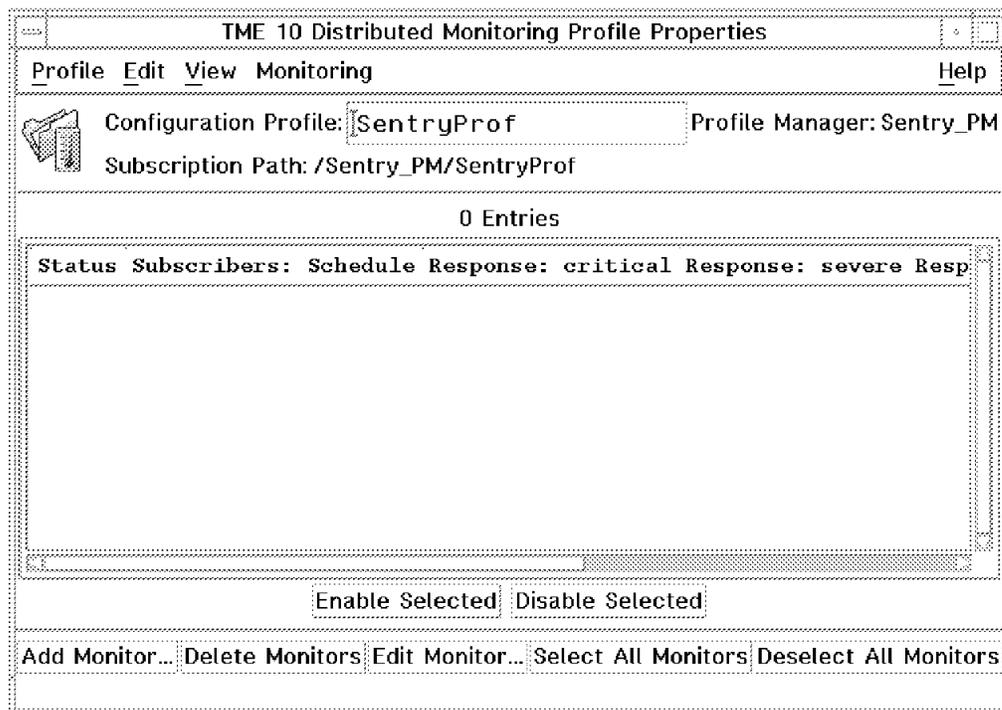
Figure 39. Profile Properties Dialog

Select the **Add Monitor...** button.
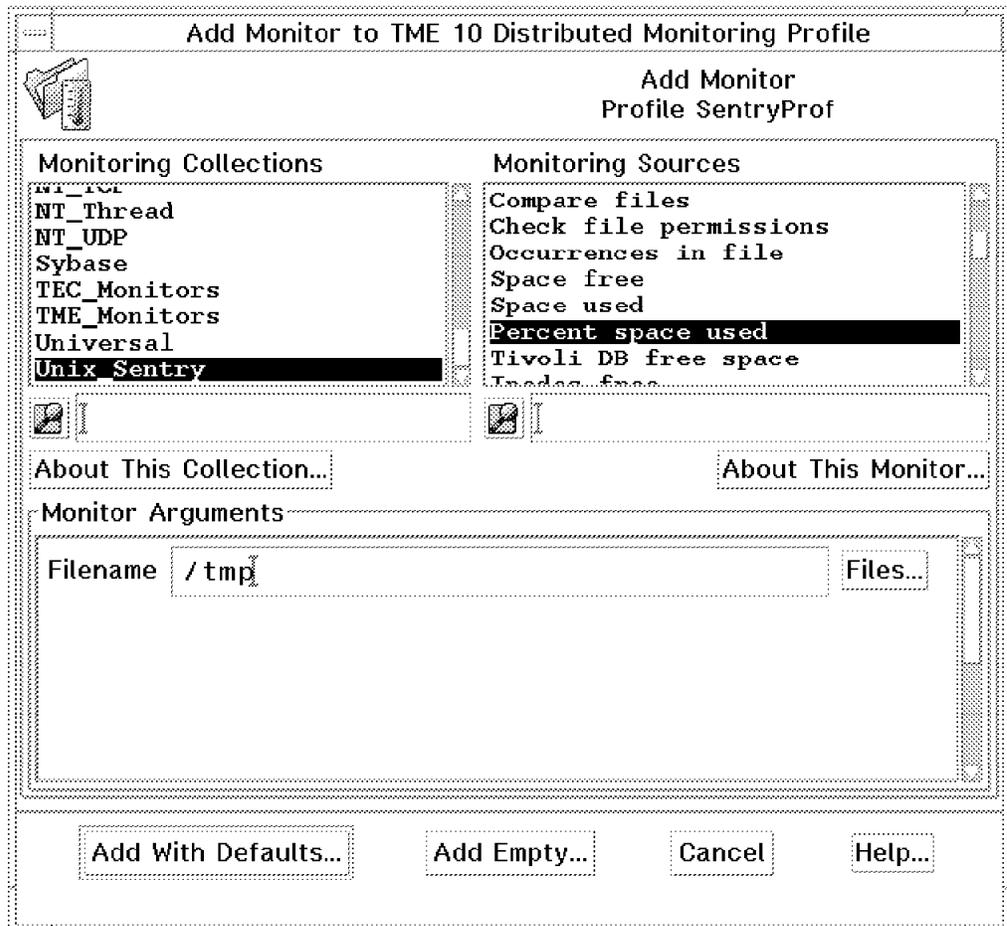
Figure 40. Add Monitor to TME 10 Distributed Monitoring Profile Dialog

Select a monitoring collection from the Monitoring Collections scrolling list. This list displays all the installed monitoring collections.

Select an option from the Monitoring Sources scrolling list. Each monitoring collection contains different monitoring sources. In this example, the Percent space used option has been selected.

Selecting a monitoring source may display one or more argument fields in the Monitor Arguments scrolling list. Argument fields are displayed only if monitoring source requires them.

The value or values that you enter in the Monitor Arguments area are also displayed at the top of the Edit Monitor dialog.

Enter the appropriate information in the Monitor Arguments field. In our example we used directory /tmp.

Click on either the **Add With Defaults** button or the **Add Empty** button to display the Edit Monitor dialog. If you click the **Add With Defaults** button, the dialog displays the default settings for the monitoring source, if any exist.

For more information about creating a monitor from the command line, refer to the waddmon command in the *TME 10 Framework Reference Guide.*

The Edit Monitor dialog enables you to configure how often TME 10 Distributed
Monitoring monitors a system resource, as well as how the application will
respond when a potential problem arises.
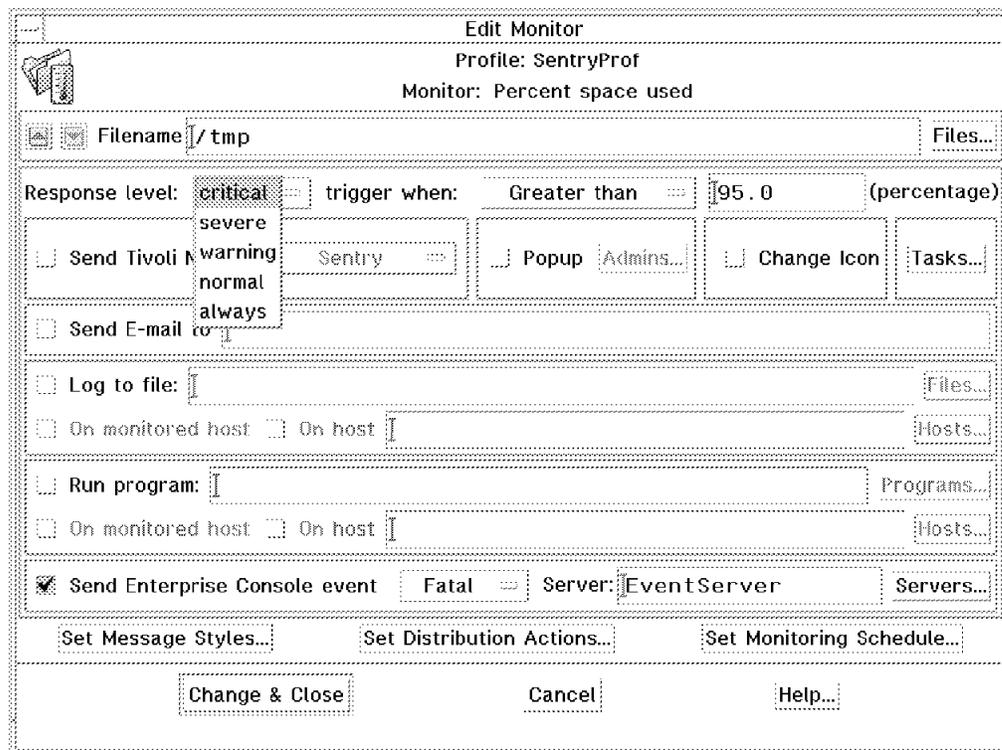
Figure 41. Edit Monitor Dialog

The values you entered in the Add Monitor to TME 10 Distributed Monitoring
Profile dialog are displayed at the top of the Edit Monitor dialog.

Setting up your monitors correctly is absolutely essential. This section provides
you with an overview. Response levels are mapped to T/EC severities and the
T/EC severity determines the color of the monitor when displayed on NGMF.
Refer to Table 1 on page 56 for an explanation of how T/EC severities map to
NGMF status.

 1. Click on the **Response level** pop-up menu and select a response level. TME
    10 Distributed Monitoring provides the following response levels:

    • Critical

      Indicates the highest response level. For some monitors, such as those
      that test for the availability of hosts and daemons, this is the only level to
      which triggered responses apply.

    • Severe

      Indicates a mid-level problem.

    • Warning

      Indicates a low-grade problem that may escalate if not attended to.

    • Normal

Indicates the response you want TME 10 Distributed Monitoring to make when normal conditions exist. This response does not accept a threshold setting.

- Always

    Indicates the response you want the application to always make, regardless of whether any of the other response levels are triggered.

2. Click on the **trigger when** pop-up menu and select a threshold option. The threshold options available depend on the source being monitored and are defined by the monitoring collection.

3. If applicable, enter a value in the argument field to the right of the response pop-up menu to define the threshold.

4. Set how you want the TME 10 Distributed Monitoring to trigger when the threshold conditions are met. You can select one or more of the following response actions:

- Send Tivoli notices

    Post a notice to a specified notification group.

- Popup Alarm

    Displays an alarm dialog on one or more administrator desktop.

- Change Icon

    Allows icons in an indicator collection to change states.

- Tasks

    Runs a task when a threshold is reached.

For more information about the different triggers, refer to the *TME 10 Distributed Monitoring User's Guide.*

The Send Enterprise Console Event option transmits TME 10 Distributed Monitoring data to the Tivoli/Enterprise Console. To enable this option, select the check box button and select a severity from the pop-up menu. Then select the host on which the Tivoli/Enterprise Console event server resides. You must also set up the event server to receive TME 10 Distributed Monitoring events. For more information about these procedures, see B.4.1, "Installation Shell Script - gemevent.sh" on page 175 or the *Tivoli Enterprise Console User's Guide.*

In our example we didn't use any triggers but used the Send Enterprise Console Event option for all response levels, except for always.

5. Repeat steps 2 through 4 until you have set the thresholds and responses for all the response levels appropriate for your environment.

```
 ┌──────────────────────────────────────────────────────────────────────┐
 │ ▄                        Edit Monitor                                  │
 │     ◸◤                  Profile: SentryProf                            │
 │    ◺◿               Monitor: Percent space used                        │
 │  ┌────────────────────────────────────────────────────────────────┐   │
 │  │ ▣ ▧ Filename ⌇/tmp                                      Files...│   │
 │  └────────────────────────────────────────────────────────────────┘   │
 │  ┌────────────────────────────────────────────────────────────────┐   │
 │  │ Response level:  │ normal ═ │                                   │   │
 │  │  ┌──────────────────────────────────────────────────────────┐  │   │
 │  │  │ ☐ Send Tivoli Notice │ Sentry ═ │  ☐ Popup │Admins...│ ☐ Change Icon │Tasks...││  │
 │  │  └──────────────────────────────────────────────────────────┘  │   │
 │  │  ┌──────────────────────────────────────────────────────────┐  │   │
 │  │  │ ☐ Send E-mail to ⌇                                       │  │   │
 │  │  └──────────────────────────────────────────────────────────┘  │   │
 │  │  ┌──────────────────────────────────────────────────────────┐  │   │
 │  │  │ ☐ Log to file: ⌇                               Files... │  │   │
 │  │  │ ☐ On monitored host ☐ On host ⌇                Hosts... │  │   │
 │  │  └──────────────────────────────────────────────────────────┘  │   │
 │  │  ┌──────────────────────────────────────────────────────────┐  │   │
 │  │  │ ☐ Run program: ⌇                             Programs... │  │   │
 │  │  │ ☐ On monitored host ☐ On host ⌇                Hosts... │  │   │
 │  │  └──────────────────────────────────────────────────────────┘  │   │
 │  │  ┌──────────────────────────────────────────────────────────┐  │   │
 │  │  │ ☑ Send Enterprise Console event │Harmless ═│ Server:│EventServer│  Servers...│  │
 │  │  └──────────────────────────────────────────────────────────┘  │   │
 │  │   Set Message Styles...      Set Distribution Actions...    Set Monitoring Schedule...│
 │  └────────────────────────────────────────────────────────────────┘   │
 │  ┌────────────────────────────────────────────────────────────────┐   │
 │  │        Change & Close          Cancel           Help...        │   │
 │  └────────────────────────────────────────────────────────────────┘   │
 └──────────────────────────────────────────────────────────────────────┘
```

Figure 42. Edit Monitor Dialog

A response level of normal will not give you the option to set any thresholds, which is also true for always.

For response level normal we choose Send Enterprise Console event but for always we turned it off. This is probably what you should do, or else your event server might be flooded.

To edit other optional settings for this monitor, select one or more of the following buttons:

- Set Message Styles...

  If you do not want to use the default format of TME 10 Distributed Monitoring messages.

- Set Distribution Actions...

  If you want to run a script or copy a file when the TME 10 Distributed Monitoring profile is distributed. (For example, you set up a monitor that looks for a file system percent used. If it reached a certain threshold, you have defined to run a program under Run program:. That file could be distributed at the same time as the monitor itself).

- Set Monitoring Schedule...

  If you do not accept the default of hourly monitoring.

- Change & Close

  To confirm the changes and dismiss the dialog. You do not need to select this button until you have set thresholds and responses for all required response levels.

In our case we want to change the schedules for our monitor so we select **Set Monitoring Schedule...**


```
┌─────────────────────────────────────────────────────────────┐
│ ▭              Set Monitoring Schedule                        │
├─────────────────────────────────────────────────────────────┤
│                      Monitoring Schedule                      │
│   ▨                    Profile SentryProf                     │
│                     Percent space used (/tmp)                 │
│                                                               │
│  ┌─────────────────────────────────────────────────────────┐ │
│  │ Start monitoring activity:                                │ │
│  │ [06     [18   [1997      1 ═    00 ═    ○ AM  ○ PM         │ │
│  │ Month   Day   Year                                        │ │
│  │                                                           │ │
│  │            Check monitor every [1      hours ═            │ │
│  └─────────────────────────────────────────────────────────┘ │
│                                                               │
│                      Restrictions...                          │
│                                                               │
├─────────────────────────────────────────────────────────────┤
│     Change & Close          Cancel           Help...          │
└─────────────────────────────────────────────────────────────┘
```
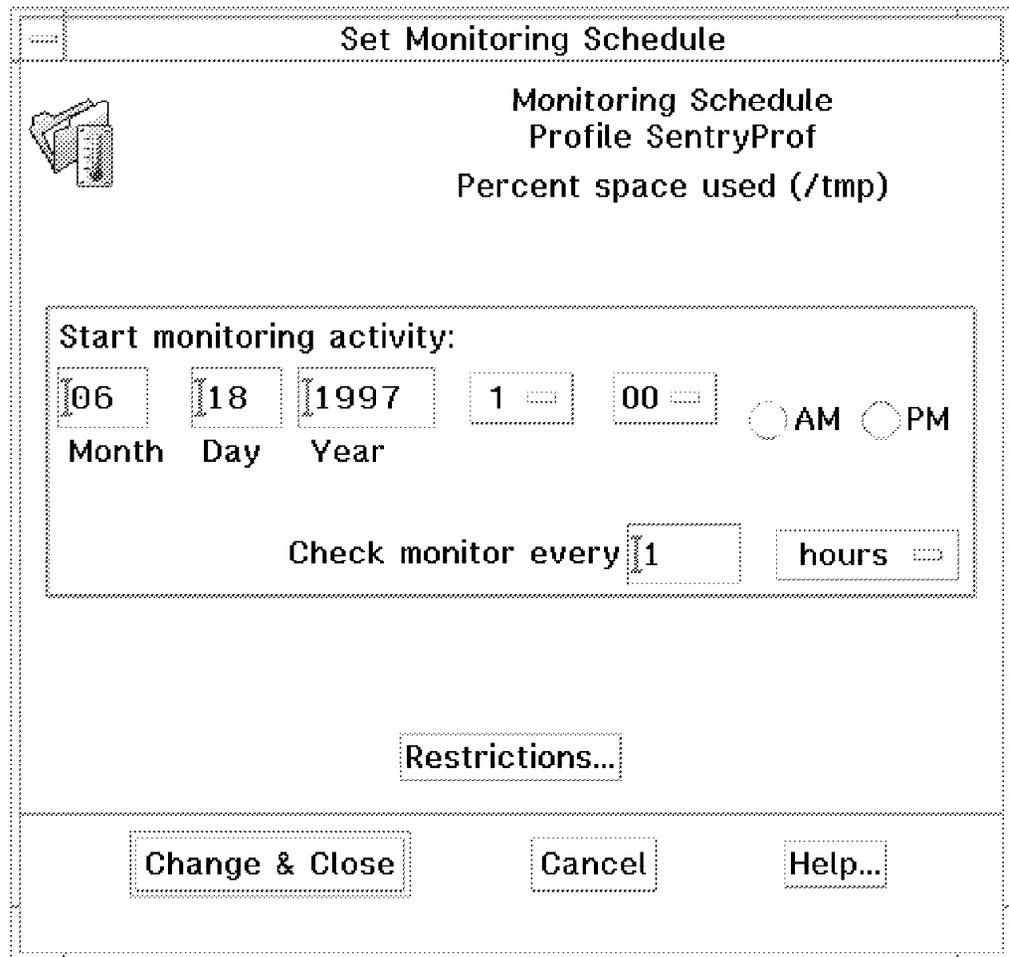
Figure 43. Set Monitoring Schedule Dialog

TME 10 Distributed Monitoring displays the Set Monitoring Schedule dialog. This dialog gives you the option to select how often a monitor checks the monitored resources. The maximum monitoring is once a minute. However Tivoli recommends that you monitor a resource no more than once every five minutes. The default monitoring schedule is to check for thresholds hourly.

Indicate the date and time you want the monitor to begin checking for thresholds in the Start monitoring activity fields. Select the hour and minute settings from the pop-up menus, and select either AM or PM.

The default start date is set to start immediately on distribution.

Indicate how often you want the monitor to check for thresholds by entering a numeric value in the Check monitor every field and selecting minutes, hours, days, weeks or months from the pop-up menu to the right of the argument field.

You can restrict monitoring to certain hours or days of week by clicking on the **Restrictions...** button.

```
┌─────────────────────────────────────────────────────────────┐
│ ▭    Monitoring Schedule Restrictions                         │
├─────────────────────────────────────────────────────────────┤
│          Monitoring Restrictions for Profile SentryProf       │
│                Monitor: Percent space used (/tmp)             │
│                                                               │
│                                                               │
│                  Allow Monitoring Activity:                   │
│                    ☑ During the day                           │
│                    ☑ At night                                 │
│                    ☑ During the week                          │
│                    ☑ On weekends                              │
│                    ☐ Custom hours                             │
│                    ☐ Custom days                              │
│                                                               │
│                                                               │
│                        Customize...                           │
├─────────────────────────────────────────────────────────────┤
│     Set & Close            Cancel            Help...          │
└─────────────────────────────────────────────────────────────┘
```
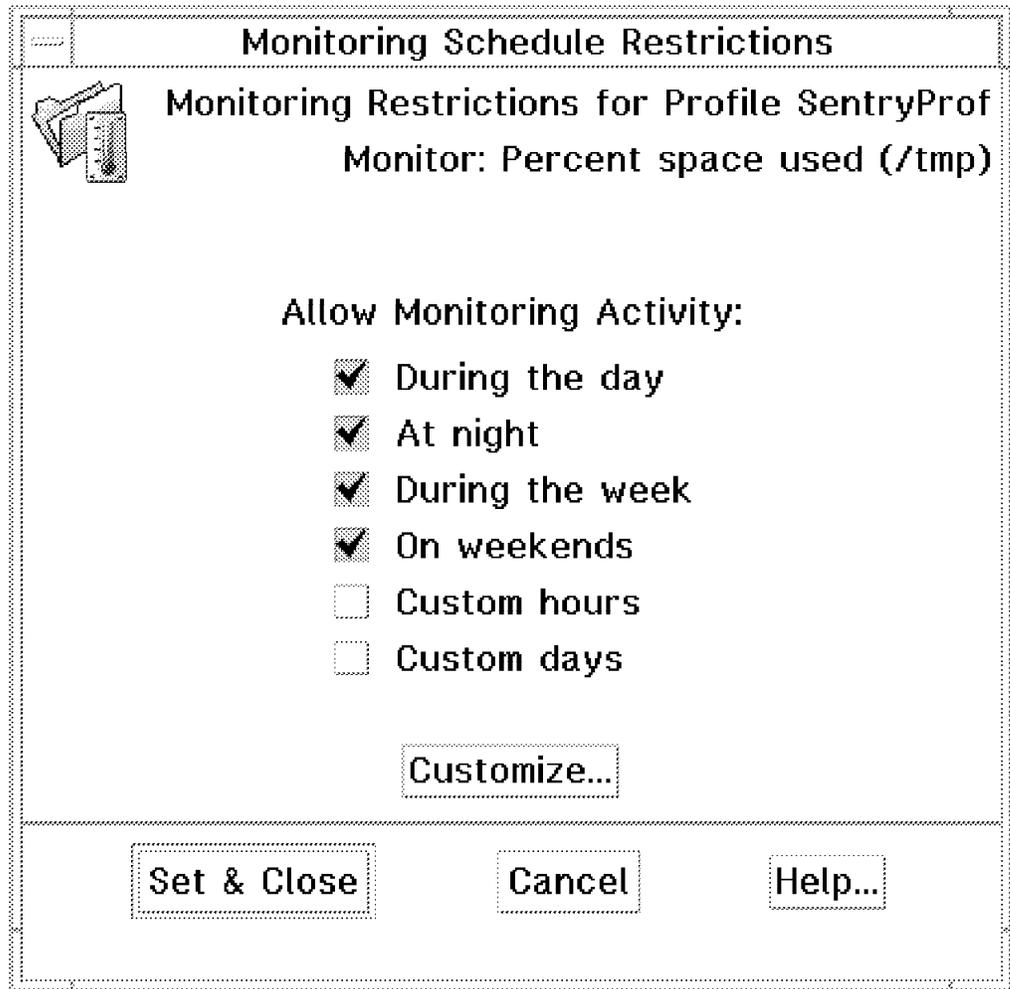
Figure 44. Monitoring Schedule Restrictions Dialog

The options on this dialog allow you to set broad restrictions on when monitoring activity can occur. The default restrictions allow monitoring all of the time.

Use the following steps to set monitor schedule restrictions:

• Choose when to allow monitoring from the options provided under Allow Monitoring Activity. Select the boxes for the times you want to allow monitoring.

• You can also redefine what is meant by During the Day, At Night, and so on, or define a user-defined restriction with the Customize option.

```
┌──────────────────────────────────────────────────────────────┐
│ ═  Customize Monitoring Schedule Restrictions                  │
│ ┌──┐    Customize Monitoring Schedule Restrictions             │
│ │▨▨│              Profile SentryProf                           │
│ └──┘           Percent space used (/tmp)                       │
│                                                                │
│                      From                    Through           │
│                                                                │
│ Daytime hours:      [ 8  ▭] AM            [ 5  ▭] PM            │
│                                                                │
│ Nighttime hours:    [ 6  ▭] PM            [ 7  ▭] AM            │
│                                                                │
│ Week days:          [Mon ▭]               [Fri ▭]              │
│                                                                │
│ Weekend days:       [Sat ▭]               [Sun ▭]              │
│                                                                │
│ [Custom hours    ]  [12 ▭] [AM ▭]         [11 ▭] [PM ▭]        │
│                                                                │
│ [Custom days     ]  [Sun ▭]               [Sat ▭]              │
│                                                                │
│      [Set & Close]        [Cancel]        [Help...]            │
│                                                                │
└──────────────────────────────────────────────────────────────┘
```
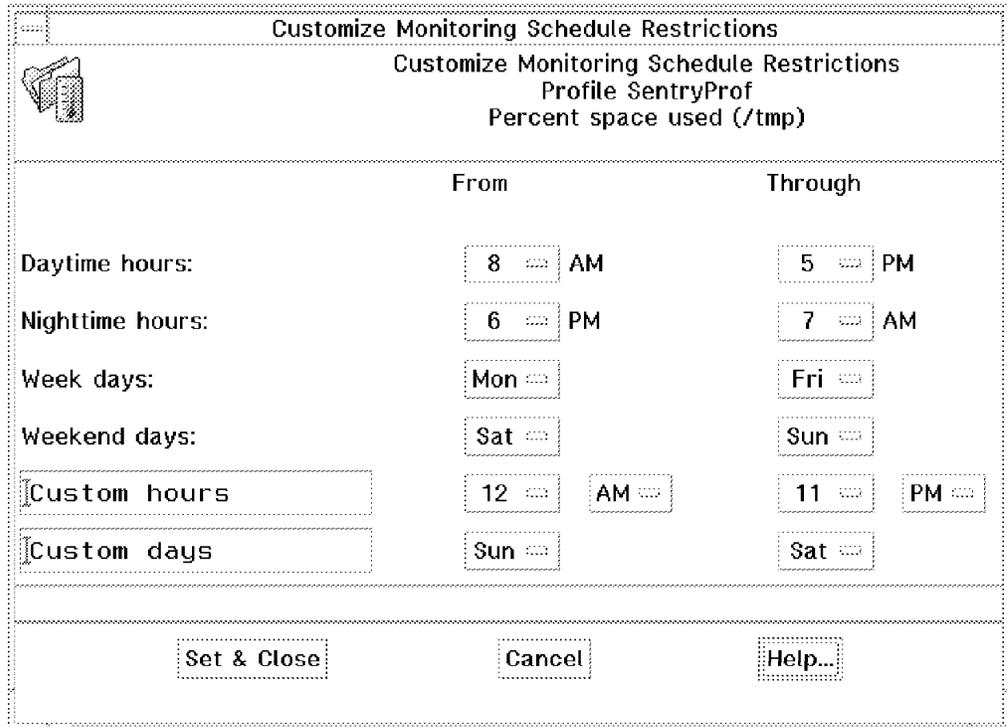
Figure 45. Customize Monitoring Schedule Restrictions Dialog

To redefine an option, or to define custom restriction parameters, choose the desired ranges from the pop-up option menus.

When you have set the options, click the **Set & Close** button to save your changes and return to the Monitoring Schedule Restrictions dialog.

When you have set all schedule restrictions to your liking, click the **Set & Close** button on the Monitoring Schedule Restrictions dialog to save your changes and dismiss the dialog and return to the Set Monitoring Schedule dialog.

Click on the **Change & Close** button to apply the schedule. This action dismisses the Set Monitoring Schedule dialog and returns to the Edit Monitor dialog.

Note: This procedure can be performed from the desktop only.

Now select **Change & Close** which will save your monitor and take you back to the TME 10 Distributed Monitoring Profile Properties dialog (see Figure 46 on page 49).

```
 ---|                        TME 10 Distributed Monitoring Profile Properties                  · ⬚
Profile  Edit  View  Monitoring                                                           Help

        ⬙         Configuration Profile: SentryProf              Profile Manager: Sentry_PM
        ⬙⬙        Subscription Path: /Sentry_PM/SentryProf

                                            1 Entries

                                 Status   Subscribers:              Schedule

         Percent space used (/tmp)   enabled    can edit    Every minute (restricte




                                   Enable Selected  Disable Selected

 Add Monitor...  Delete Monitors  Edit Monitor...  Select All Monitors  Deselect All Monitors
```

Figure 46. TME 10 Distributed Monitoring Profile Properties Dialog

You can now see the newly created monitor and you can also manipulate the
status of the monitor from this dialog.

Now it is time to distribute the Sentry profile to the subscribing managed nodes.

3.3.2.6 Distribute Profile to Managed Nodes

Select **Distribute...** from the Profile menu. TME 10 Distributed Monitoring
displays the Distribute Profile dialog (see Figure 48 on page 51).
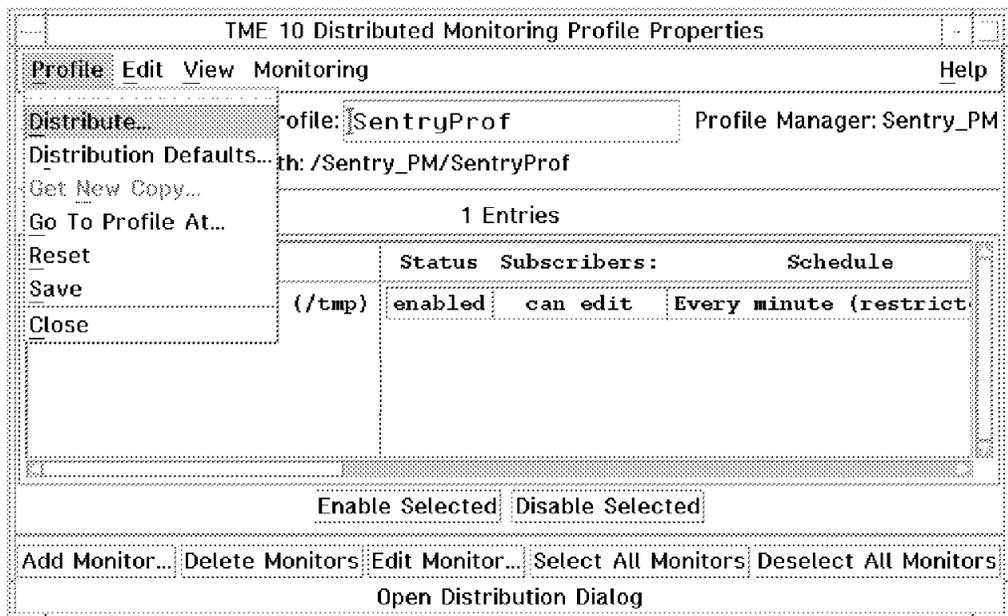
Figure 47. Distribute Profile Selection

```
┌────────────────────────────────────────────────────────────────────────┐
│ ⌐..┐                          Distribute Profile                        │
├────────────────────────────────────────────────────────────────────────┤
│  ▨▨        Distribute Copies of Configuration Profile: SentryProf        │
│  ▨▨                  in Profile Endpoint: Sentry_PM                      │
├────────────────────────────────────────────────────────────────────────┤
│  ┌Distribute To:────────┐ ┌Distribute Will:─────────────────────┐       │
│  │  ◯ Next Level Of      │ │  ◯ Preserve modifications in         │       │
│  │     Subscribers       │ │     subscribers' copy of the profile │       │
│  │                       │ │                                      │       │
│  │  ◯ All Levels Of      │ │  ◯ Make subscribers' profile an      │       │
│  │     Subscribers       │ │     EXACT COPY of this profile       │       │
│  └───────────────────────┘ └──────────────────────────────────────┘      │
│                                                                          │
│  Distribute To These Subscribers      Don't Distribute To These Subscribers│
│  ┌──────────────────────────┐▢      ┌──────────────────────────────┐▢   │
│  │ rs600015                 │       │                              │     │
│  │                          │  ◀    │                              │     │
│  │                          │  ▶    │                              │     │
│  │                          │       │                              │     │
│  └──────────────────────────┘▢      └──────────────────────────────┘▢   │
│  ▨ ▨ ▨ [                    ]         ▨ ▨ ▨ [                    ]        │
│  ┌──────────────────────────────────────────────────────────────────┐   │
│  │                          Schedule...                               │   │
│  └──────────────────────────────────────────────────────────────────┘   │
│  ┌───────────────┐  ┌──────────┐  ┌──────┐  ┌──────┐  ┌──────┐           │
│  │Distribute & Close│ │Distribute│  │Reset │  │Close │  │Help  │         │
│  └───────────────┘  └──────────┘  └──────┘  └──────┘  └──────┘           │
└────────────────────────────────────────────────────────────────────────┘
```

Figure 48. Distribute Profile Dialog

Select one of the Distribute To options based on the following:

- Next Level of Subscribers

  Distributes the profile only to subscribers named in the Distribute To These Subscribers scrolling list of the Distribute Profile dialog.

- All Levels of Subscribers

  Distributes the profile to all subscribers and all subscribers' subscribers.

Select one of the Distribute Will options.

Select the subscribers that you want to receive the profile by choosing them from the Don't Distribute To These Subscribers scrolling list and move them to the Distribute To These Subscribers list by clicking on the left arrow button.

Click the **Distribute & Close** to distribute the profile immediately and return to the TME 10 Distributed Monitoring Profile Properties dialog.
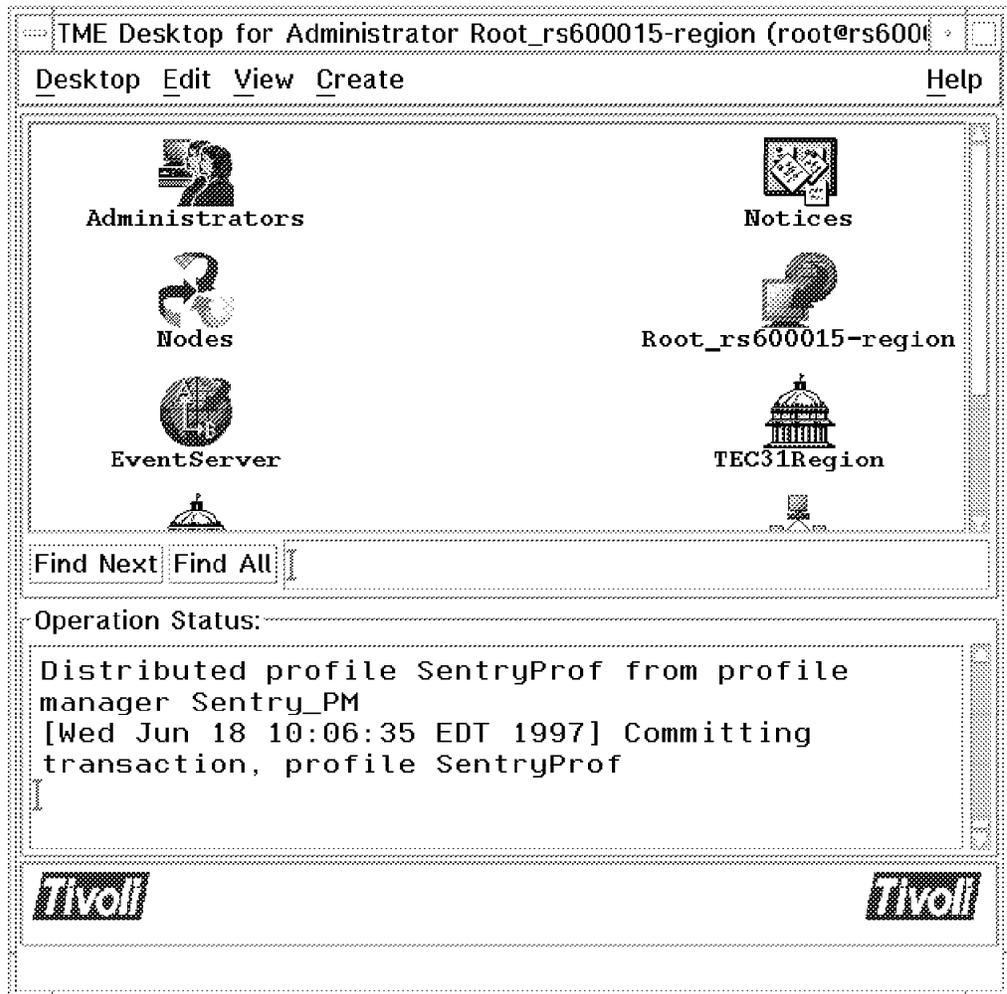
Figure 49. Tivoli Management Platform Operation Status

In the Operation Status area at the TME Desktop, you can notice that the profile SentryProf from profile manager Sentry_PM has been distributed.

To see TEC events open the console icon and go to the source group window and select SENTRY. This is explained in 3.4.2, "Verifying the TME 10 Distributed Monitoring Monitors" on page 60.

3.3.3 Customizing Topology Services

This section explains how to customize the MultiSystem Manager agent and explains the various programs that are supplied with the agent. Some high-level flow diagrams are included to explain the flows between the MultiSystem Manager agent and TME 10 NetView for OS/390.

3.3.3.1 Setting Up MultiSystem Manager Agent

The MultiSystem Manager Agent installation in the TMR, is documented in *TME 10 Global Enterprise Manager - Topology Service User′s Guide*, available with the product.

The MultiSystem Manager Agent installation for AIX puts the files in a directory named /usr/local/Tivoli/bin/generic_unix/GEM/MSMAgent; and creates a UNIX

and Tivoli user named MSMAgent. All the files in that directory must be owned by MSMAgent, and the permissions must be 700 (rwx------).

The MultiSystem Manager Agent is started using the program FLCT_cmdServer.pl. Prior to running it, some customization is necessary.

The MultiSystem Manager Agent opens a port for communication with TME 10 NetView for OS/390. By default this port is 3333, which matches the default port used by the GETTOPO command. If this port is used by any other application, you could change it by editing the MSMAgent.cfg file in the UNIX platform, and modifying the CMDSERVERPORT= parameter. Remember to modify the FLCSITME member to match the new port (refer to A.2.1, "FLCSITME Used for the TME Feature of MultiSystem Manager" on page 138).

```
# ****************************************************************
#      5697-B83 (C) Copyright IBM Corp. 1997.
#       All Rights Reserved.
#
# Name:          MSMAgent.cfg
#
#  Description: This file contains configuration parameters for the
#               TME MSMAgent. Each parameter is set to a default
#               which may be modified by the system administrator
#               by editing this file directly.
#
# ****************************************************************

LOGSIZE=10              # maximum size of MSMAgent.log in megabytes
                        # if the MSMAgent.log exceeds this size it will be
                        # overwritten.

CMDSERVERPORT=3333      # the port number on which cmdServer.pl will listen
                        # for connection requests from MultiSystemsManager
```

Figure 50. MSMAgent Configuration File

3.3.3.2 MultiSystem Manager INITTOPO and GETTOPO Flow

Before starting FLCT_cmdServer.pl, the event server must be started. When FLCT_cmdServer.pl starts, it reads the MSMAgent.cfg file for the initialization parameters, and creates a TCP/IP server that listens for commands coming from TME 10 NetView for OS/390. After starting, it sends an event to the T/EC console, informing you that the MultiSystem Manager Agent is up.

If the rules to forward events to TME 10 NetView for OS/390 are loaded, this event goes to TME 10 NetView for OS/390. The event is captured in the automation table, and then a GETTOPO TMERES command is issued.

The following is a high-level flow diagram of the INIT and GETTOPO command flow:
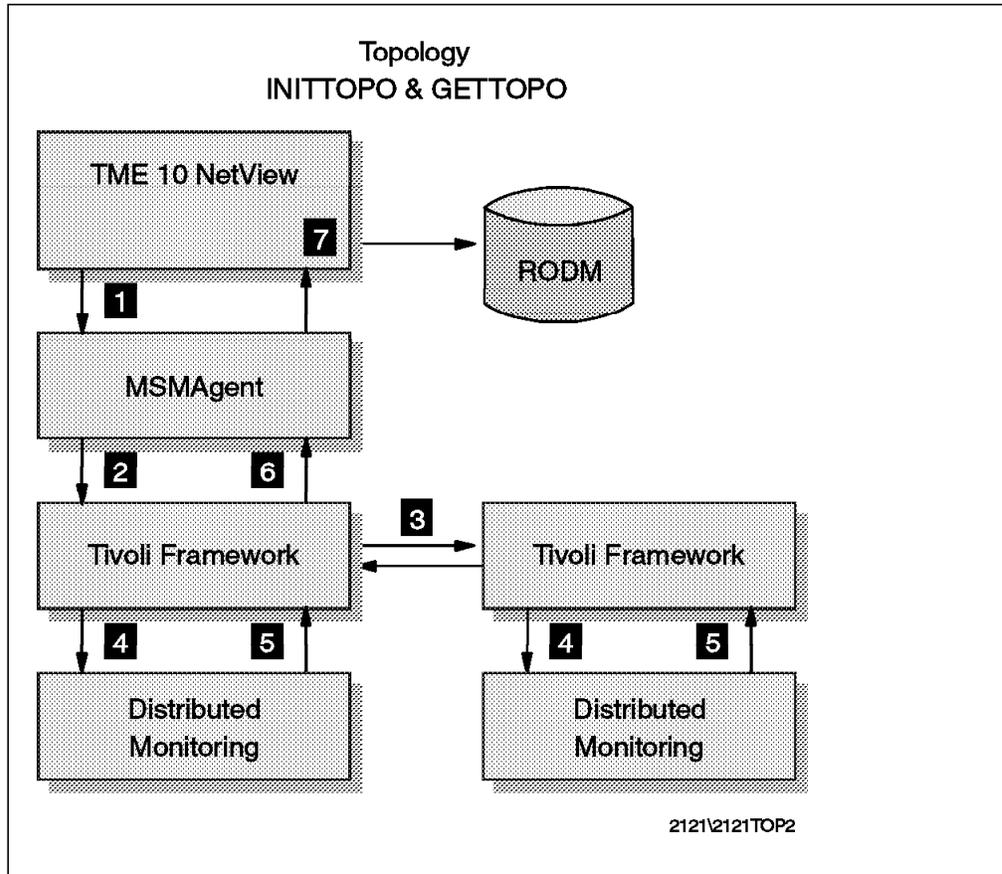
Figure 51. INITTOPO and GETTOPO Command Flow

1 GETTOPO is sent to the MultiSystem Manager Agent.

> **2** The MultiSystem Manager Agent starts programs that asks the framework for information as described below.

> **3** The local framework gathers information from itself, and from other interconnected regions.

> **4** The framework asks TME 10 Distributed Monitoring for information about monitors.

> **5** TME 10 Distributed Monitoring sends back the name and status of each monitor.

> **6** The framework sends back all the information gathered to the MultiSystem Manager Agent.

> **7** The MultiSystem Manager Agent formats that information and sends it back to TME 10 NetView for OS/390 that loads RODM.

> A more detailed description of what exactly happens in step **2** is provided below.

The MSM Agent receives the INITOPO or GETTOPO command and starts the following three programs:

1. FLCT_getServicePoint.pl: This program returns the name of the TMR where the MSM Agent is running, and the status of the local oserv daemon, and T/EC.

2. FLCT_getPhysical.pl -m: This program retrieves the physical topology of the local TMR and all the interconnected TMRs. Physical topology is composed of managed nodes and managed nodes acting as repeaters. Physical topology information is stored in a file called managedNodeDb.pag. This file contains information about managed nodes such as IP address, name and status. When the parameter -m is specified, the program executes another program named FLCT_getSentryMonitors.pl that gets the name and status of all the TME10 Distributed Monitoring monitors.

3. FLCT_getLogical.pl: This program retrieves the logical topology of the local TMR and all the interconnected TMRs. Logical topology is composed of managed nodes and TME 10 Distributed Monitoring monitors defined on them.

In **4** you should be aware of a specific consideration for program FLCT_getSentryMonitors.pl. The way that this program discovers the TME 10 Distributed Monitoring monitors in each managed node is by executing the TME 10 Distributed Monitoring command wlseng -l hostname. This command retrieves the name of all the monitors and the last TME 10 Distributed Monitoring status of all of them. Then, it associates the last response level obtained in each monitor with the T/EC severity defined for that level. (For instance, we associated a TME 10 Distributed Monitoring response level of NORMAL, to a T/EC event severity of HARMLESS). If the monitor has never been initialized (it doesn't have any status), the program will assign a status depending of the parameter INITIALSTATUS= specified in the MSMAgent.cfg file (refer to B.1.1, "MSM Configuration File - MSMAgent.cfg" on page 143).

3.3.3.3 MultiSystem Manager Flow for Status Changes

From now on, any change in the status of the TME 10 Distributed Monitoring monitors is sent as an event to the T/EC console; and if the forwarding rules to TME 10 NetView for OS/390 are loaded, they are also sent to TME 10 NetView for OS/390.

The following is a high-level flow diagram of topology status changes:
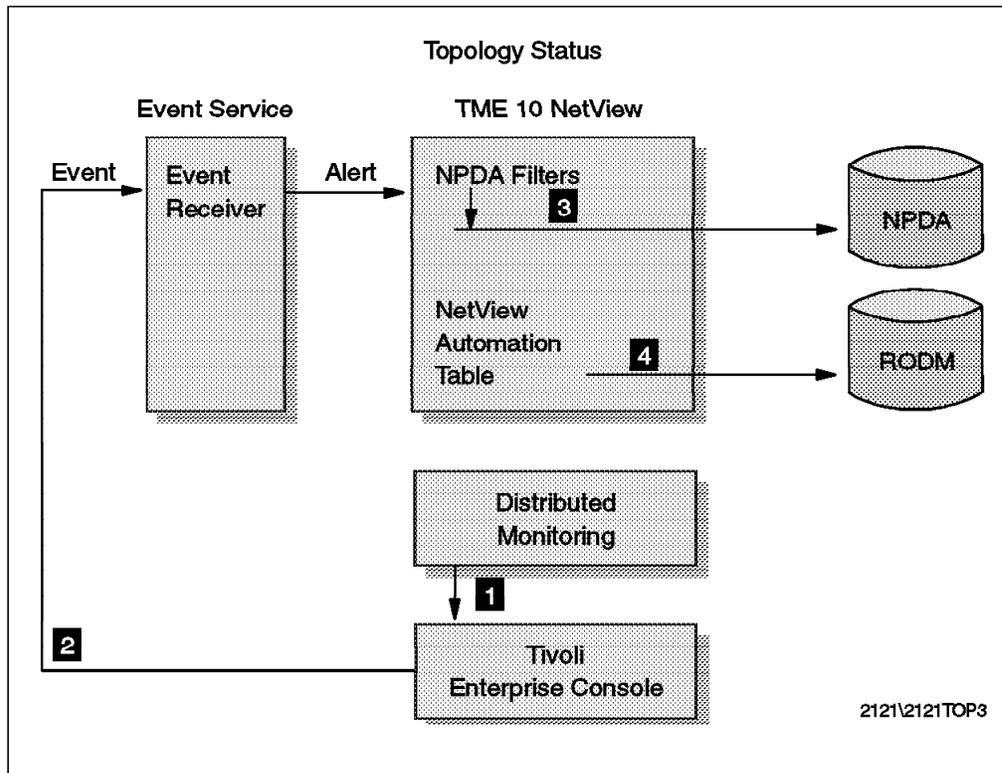
Figure 52. Status Changes Flow

1 TME 10 Distributed Monitoring sends an event to T/EC.

2 T/EC forwards the event to the Event/Automation Service in MVS, which transforms the event into an SNA alert.

3 The alert passes the NPDA filters and is written in the NPDA database.

4 The automation table gets the alert, and starts a process to modify data in RODM.

TME 10 NetView for OS/390 will change the status of the object on NGMF based on the severity of the event.

Table 1. Event Severity and NGMF Status Mapping	
T/EC Severity	**NGMF Status**
FATAL	UNSATISFACTORY/red
CRITICAL	UNSATISFACTORY/red
MINOR	INTERMEDIATE/white
WARNING	INTERMEDIATE/white
HARMLESS	SATISFACTORY/green
UNKNOWN	UNKNOWN/grey

3.3.3.4 Status of Managed Nodes

At this point we have the topology of the network loaded into RODM, and the monitors are able to generate alerts that change the status of the objects according to their severity. But we are not able to know when a managed node changes its status, from up to down or vice versa.

Only the framework knows the status of the managed nodes. The MSM Agent provides a program that checks against the framework the status of all the managed nodes. The program is named FLCT_updateMNStatus.pl. This program asks the framework for status and compares the answer with the file managedNodeDb.pag. If there is any change, it sends an event to the T/EC console informing of the change, and updates the file. When the event arrives at TME 10 NetView for OS/390, the automation table captures it, and changes the status of the object oserv().

The alert that arrives at TME 10 NetView for OS/390 is FLCT018I - MSM Managed Node Status Change.

The program FLCT_updateMNStatus.pl is also able to detect that a managed node has been added or deleted.

When this happens two events are sent to T/EC, and then to TME 10 NetView for OS/390:

- FLCT017I - MSM Managed Node Added or
- FLCT019I - MSM Managed Node Deleted and
- FLCT022I - MSM Topology Changes

The automation table captures the event, and then issues a GETTOPO command.

The program FLCT_updateMNStatus.pl needs to run scheduled and unattended. MSM Agent provides a program named FLCT_MSMTasks.pl, which creates a job that starts the program FLCT_updateMNStatus.pl every five minutes. Run this program only once to create the job. When you run this program you should be logged on as MSMAgent. If not, you will find that the files managedNodeDb.dir and managedNodeDb.pag will be owned by the user ID you used when running the program. The consequence of this is that the MSMAgent is unable to start since it checks that all files are owned by MSMAgent.

The default timer can be changed by editing FLCT_MSMTasks.pl prior to running it, or using the TME10 GUI.

The MultiSystem Manager Agent also provides a program named FLCT_viewDb.pl that can be used at any time to check the status of managed nodes. This program formats the file managedNodeDb.pag and shows the result on the screen:

```
1302594782.1=0,1302594782,1,rs600015,up,9.24.104.215,rs600015-region
1302594782.2=0,1302594782,2,WTR05193,up,9.24.104.239,rs600015-region
1585366483.1=0,1585366483,1,foxtrot,up,9.24.104.155,foxtrot-region
lock=0
```

3.3.3.5 Setting of T/EC Severity to Sentry Response Levels

MultiSystem Manager agent provides a program named FLCT_setMonitors.pl, which assigns T/EC severity to TME 10 Distributed Monitoring response levels for all the monitors in a specific TME 10 Distributed Monitoring Profile.

This program uses the parameter PROFILENAME.x= in the MSMAgent.cfg file, to select the Sentry profile to which you want to assign events.

The program assigns in each monitor of the Sentry profile, a response level to a T/EC severity. As a result of that the ALWAYS response level is assigned to a T/EC event severity. We don't think this is a desirable situation in a production environment. We therefore strongly recommend that you don't run this program until you have made sure you know exactly what it does, and if that is what you want.

3.3.3.6 Sending Commands to TME Resources from NetView/390

The MultiSystem Manager Agent can receive workstation commands from TME 10 NetView for OS/390, and execute them in the desired managed node.

The program FLCT_rcmd.pl receives the command we want to execute, and using TME 10 Framework communication, the attached command is executed in the desired workstation.
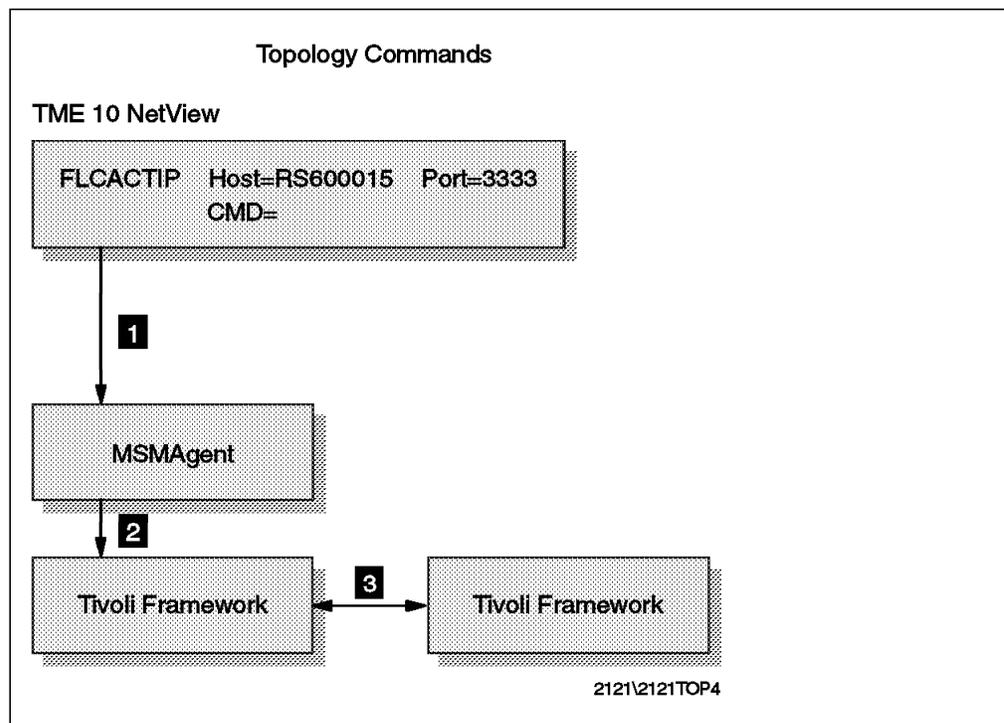
Figure 53. Topology Service Command Flow

1 NetView operator sends a workstation command to the MultiSystem Manager Agent, using the FLCACTIP command.

2 The MultiSystem Manager Agent sends the command to the local framework.

3 If the command is to be executed in a remote workstation, FLCT_rcmd.pl will use the local framework to send it to the remote managed node where it will be executed.

3.4 Verify the Installation

Before starting the MultiSystem Manager Agent, we should check that all the products involved in the Topology Service are up and running.

We do this in three steps:

1. Verify that events are forwarded to TME 10 NetView for OS/390.

2. Verify that TME 10 Distributed Monitoring monitors are able to send events to T/EC.

3. Verify that the MultiSystem Manager Agent starts with no errors.

3.4.1 Verifying Event/Automation Service

First ping the MVS host to check for connectivity. To verify that the Event/Automation Service is up and running, we need to check that the events received in T/EC are sent to TME 10 NetView for OS/390. An easy way to check it is to manually generate an event with the wpostemsg command.

For instance, we created this event of the source SENTRY by issuing the following command:

`wpostemsg -r CRITICAL -m "Testing Event" Sentry2_0_ulogins SENTRY`

```
                              aixterm
root@rs600015[/]# wpostemsg -r CRITICAL -m "Testing Event" Sentry2_0_ulogins SENT>
root@rs600015[/]# _
```

Figure 54. wpostemsg Command

We go to the T/EC console and open the SENTRY Event source. We should see a message as shown in Figure 55 on page 60.
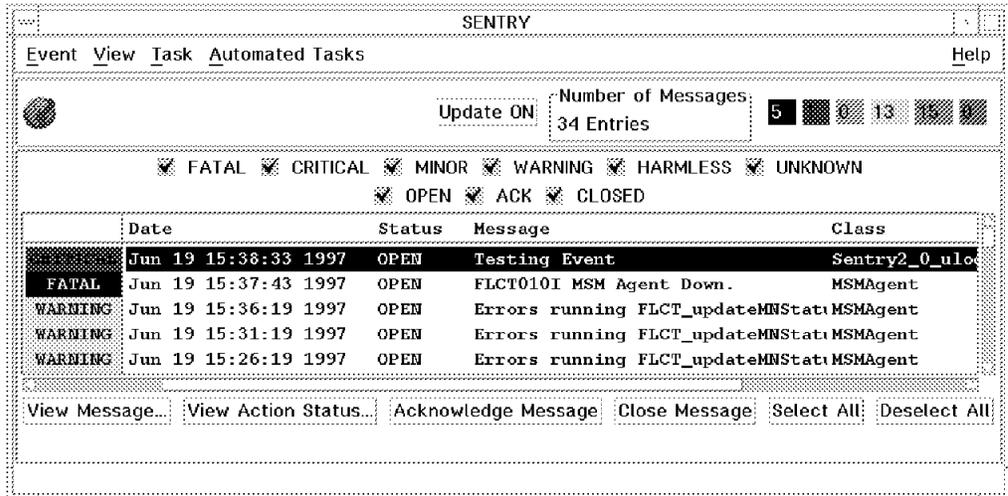
Figure 55. T/EC Testing Event

We go to NPDA in TME 10 NetView for OS/390 and see a message as shown in Figure 56.
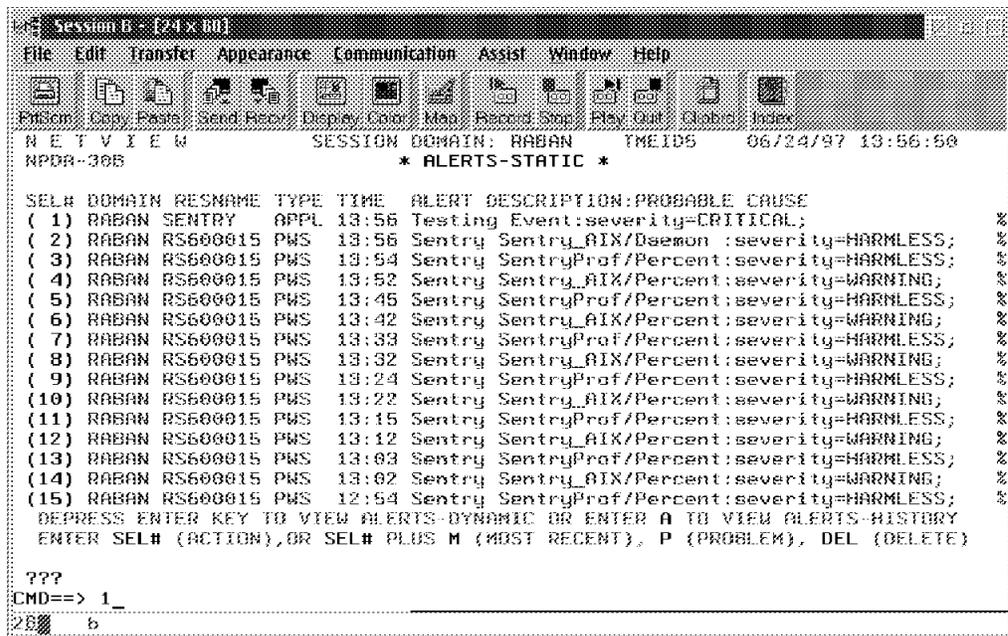
Figure 56. NPDA Testing Event

3.4.2 Verifying the TME 10 Distributed Monitoring Monitors

To verify that the monitor is capable of sending events to T/EC, we need to set up a monitor for testing purposes. Define in that monitor a CRITICAL and a NORMAL condition, associated to CRITICAL and HARMLESS T/EC events.

When the monitor starts you should see an event on the T/EC console, CRITICAL or HARMLESS depending of the result of the monitor. In NPDA, we also should see the alert related to that event.

3.4.2.1 Events in T/EC

The monitor we ran for this testing, checks the percentage of used space in the file system /tmp. The monitor checks each minute, and sends an event CRITICAL if the percentage is greater than 80%, and sends a HARMLESS event when it is lower than 70%.

To see that the events are coming, we go to the Event Console and open the SENTRY source events.

We see two events from the Sentry profile SentryProf, from the Monitor Percent Space Used, and with severities CRITICAL and HARMLESS (first and second event in Figure 57).

Figure 57. T/EC Console for SENTRY Events

Select the event you want to see more detailed information about and click the **View Message...** button.

```
┌─────────────────────────────────────────────────────┐
│ ▦               Event Group Message Viewer        ▫ □ │
├─────────────────────────────────────────────────────┤
│ ⬤                        Message: 21866646420         │
├─────────────────────────────────────────────────────┤
│ Sentry2_0_diskusedpct;                                │
│         server_handle=1;                              │
│         date_reception=866646420;                     │
│         event_handle=2;                               │
│         source=SENTRY;                                │
│         sub_source=SentryProf;                        │
│         origin=127.0.0.1;                             │
│         sub_origin=rs600015;                          │
│         hostname=rs600015;                            │
│         adapter_host=rs600015;                        │
│         status=OPEN;                                  │
│         administrator='';                             │
│         acl=[ admin];                                 │
│         severity=HARMLESS;                            │
│         date='Jun 18, 1997 15:07';                    │
│         duration=0;                                   │
│         msg='Sentry SentryProf/Percent                │
│ space used on host rs600015 Wed Jun 18                │
│ 15:07:00 1997                                         │
│                                                       │
│ Status: >>> normal <<<                                │
│                                                       │
│ Percent space used (/tmp)  (default                  │
│ result)                                               │
│ (Previous: 63.2918 (percentage) Current:              │
│ 63.2021 Effective: 63.2021)                           │
│                                                       │
│              ┌─────────────────────┐                  │
│              │ View Action Status… │                  │
│              └─────────────────────┘                  │
├─────────────────────────────────────────────────────┤
│        ┌───────┐              ┌───────┐                │
│        │ Close │              │ Help… │                │
│        └───────┘              └───────┘                │
└─────────────────────────────────────────────────────┘
```

Figure 58. T/EC Console for Sentry Events - View Message

A detailed description of the event will be displayed showing the percent space used in file system /tmp on host RS600015.

3.4.2.2 Alerts in NPDA
The event forwarding rule in T/EC will forward the events to TME 10 NetView for OS/390. The events will be converted to an alert and can then be displayed in NPDA.

The following section shows you detailed examples of such CRITICAL and HARMLESS alerts in NPDA.

Figure 59. NPDA Alerts-Static Screen

We see one alert from the Sentry profile SentryProf, from the Monitor Percent Space Used, and with severity CRITICAL.

Enter 1 to select the CRITICAL alert.

Figure 60. Recommended Action Screen

Enter DM for Detail Menu.

Figure 61. Event Detail Menu Screen

Enter 1 to see the Event Detail.

Figure 62. Event Detail Screen - Page 1

Press PF8 to go to the next screen.

Figure 63. Event Detail Screen - Page 2

Press PF8 to go to the next screen.
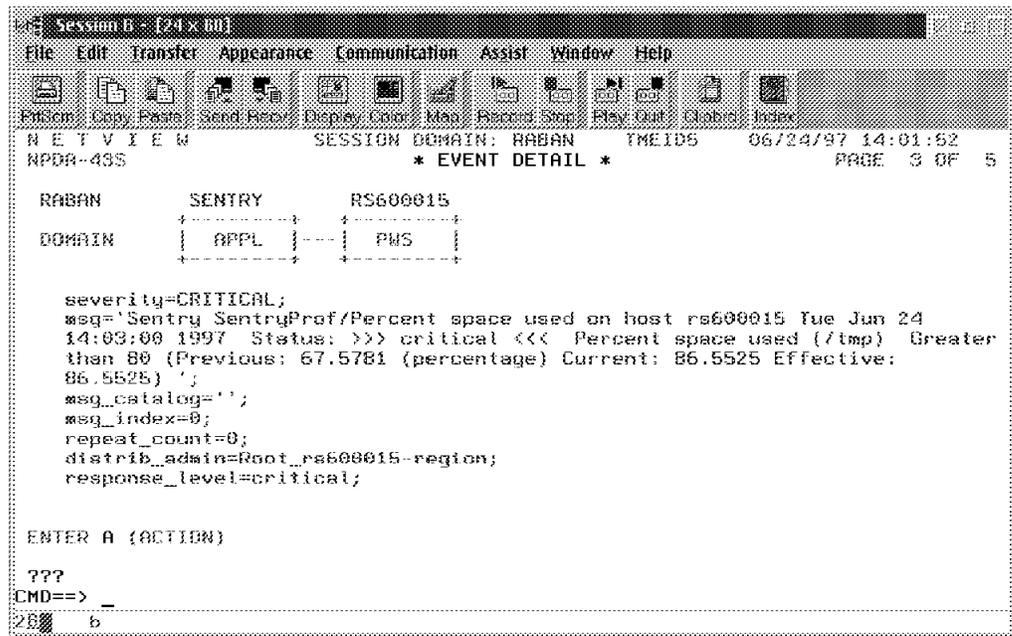
Figure 64. Event Detail Screen - Page 3

Press PF8 to go to the next screen.
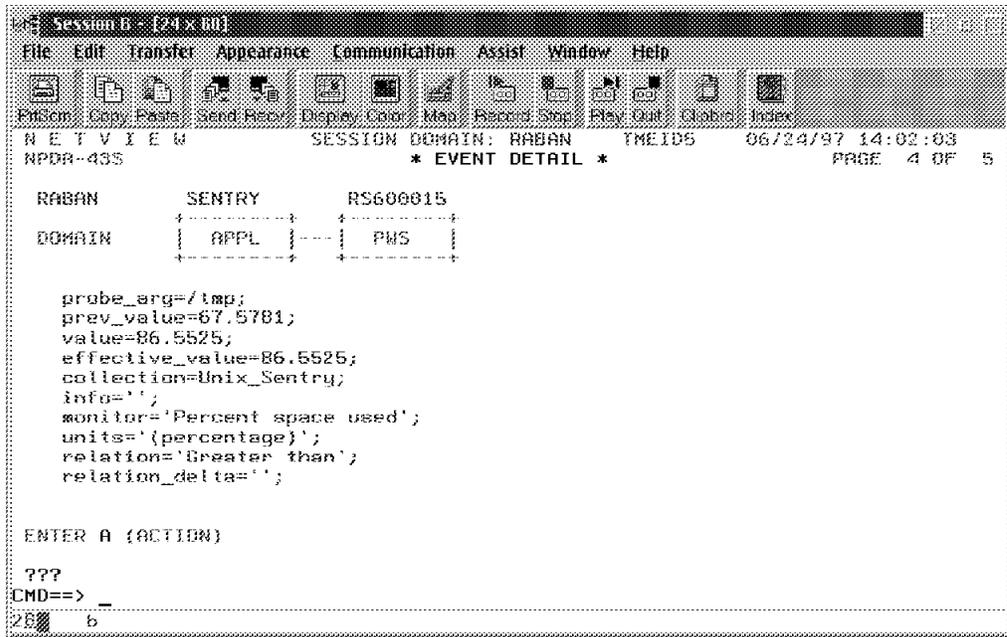
Figure 65. Event Detail Screen - Page 4

Press PF8 to go to the next screen.
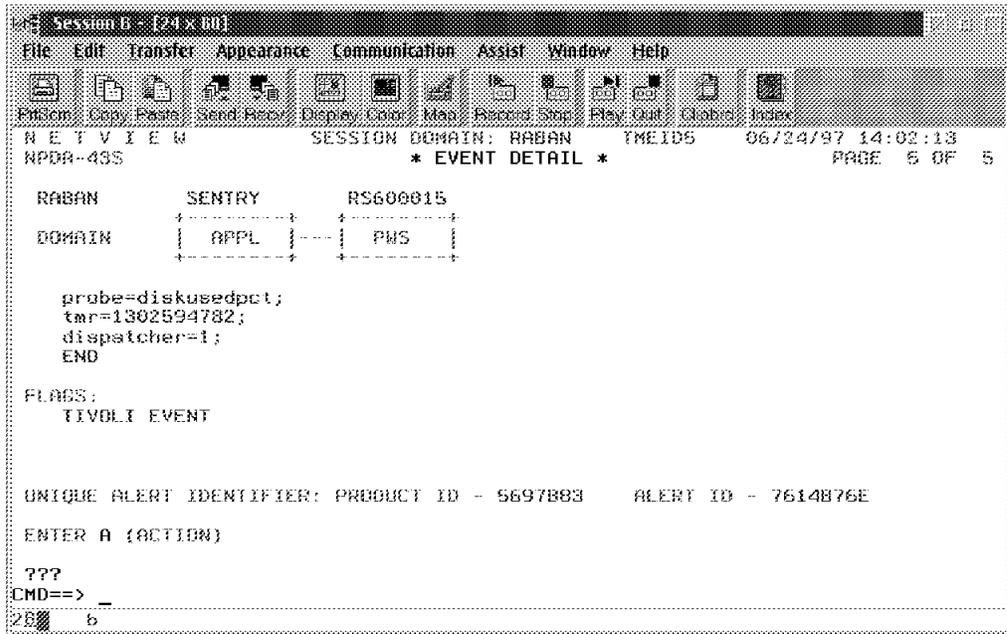
Figure 66. Event Detail Screen - Page 5

To see the HARMLESS alert go back to the Alert-Static screen, and select alert number **1**.

Figure 67. NPDA Alerts-Static Screen

Navigating through panels as we did in 3.4.2.2 (Alerts in NPDA), we find the condition triggering the harmless event. Current Space Used is 67% and the monitor was set up to trigger when utilization was below 70%.

Figure 68. Event Detail Screen - Page 3

3.4.3 Verifying the MultiSystem Manager Agent

Issue the command INITTOPO in TME 10 NetView for OS/390 to set up the MultiSystem Manager environment.

Check that the automation table is running and that the the MultiSystem Manager operators are logged on.

Log on as MSMAgent in AIX and start the program FLCT_cmdServer.pl.

Check that an alert MSM Agent Up arrives to NPDA.

Go to NPDA Alert Static panel, and look for an alert with the description of FLCT009I MSM Agent Up.

Figure 69. NPDA Alerts-Static Screen MSM Agent Up Event

Enter 1 to select the alert.

The NPDA event detail contains the MSM Agent Up event and it has a severity of harmless.
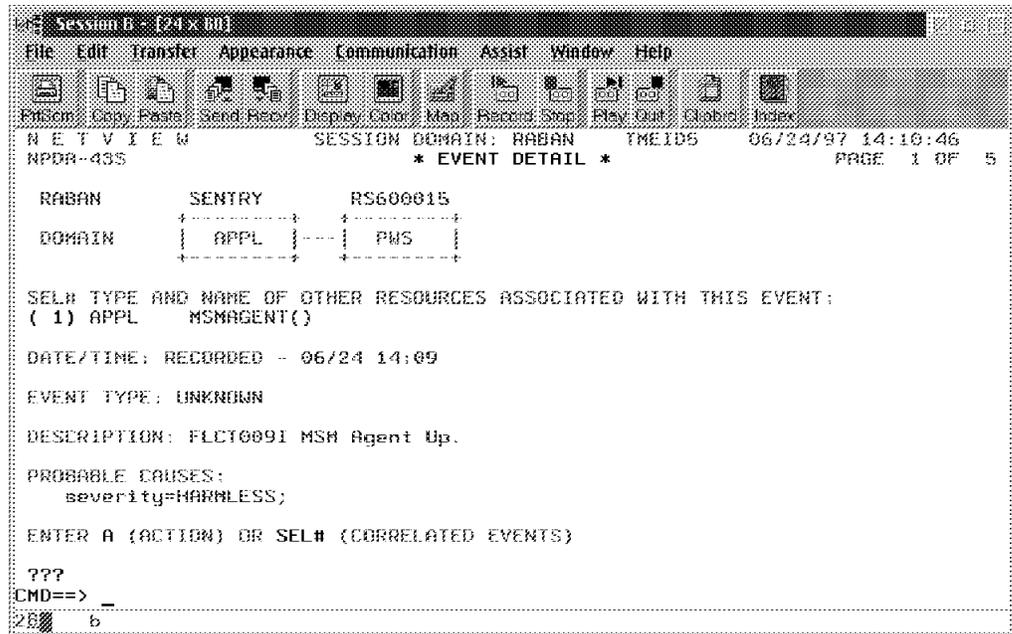
Figure 70. Event Detail Screen - Page 1

When the alert arrives, an automatic GETTOPO is sent to the agent.

Check that the agent in AIX is receiving the command and execute the following programs:

1. FLCT_getServicePoint.pl

2. FLCT_getPhysical.pl

3. FLCT_getLogical.pl

```
$ FLCT_cmdServer.pl
Port = 3333
Listening for connection 1....
accept ok
Listening for connection 2....
1: 4098 19562 192 168 238 11
on connection 1: executing cmd: FLCT_getServicePoint.pl
accept ok
Listening for connection 3....
2: 4098 19563 192 168 238 11
on connection 2: executing cmd: FLCT_getPhysical.pl -m
accept ok
Listening for connection 4....
3: 4098 19575 192 168 238 11
on connection 3: executing cmd: FLCT_getLogical.pl
```

Figure 71. MultiSystem Manager Agent Startup

3.4.4 Verifying Command Support

Send a command to the agent from an NCCF screen for instance, netvasis flcactip host=rs600015 port=3333 cmd=FLCT_rcmd.pl foxtrot uname as shown in Figure 72.

This command executes the uname command in the workstation foxtrot. The command is sent to the service point rs600015 where the MSM Agent is started through port 3333. FLCT_rcmd.pl will send it to the remote workstation named foxtrot and the command uname will be executed there.

```
Session B - [24 x 80]
File   Edit   Transfer   Appearance   Communication   Assist   Window   Help

PfCom  Copy Paste  Send Recv  Display Color  Map  Record Stop  Play Quit  Global  Index
NCCF                         N E T V I E W     RABAN TMEID5   06/24/97 14:20:23 A
* RABAN       flcactip host=rs600015 port=3333 cmd=FLCT_rcmd.pl foxtrot uname
| RABAN       Windows_NT
------------------------------------------------------------------------------


                                    _




???

25     b
```
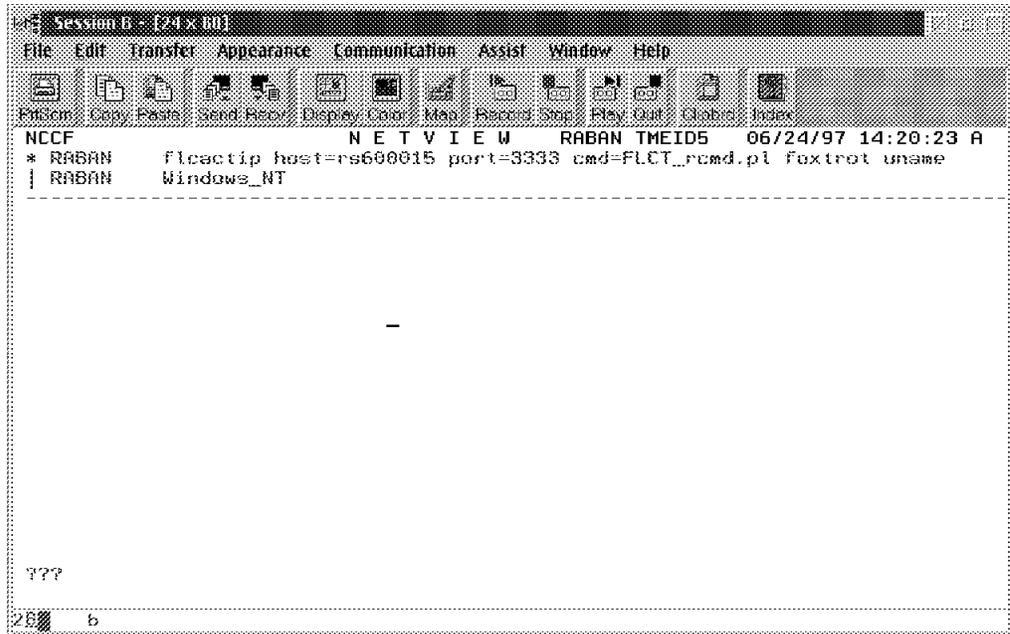
Figure 72. MultiSystem Manager Command and Response

3.5 Operational Example

We worked with interconnected regions to simulate a real environment. The monitors defined should be considered examples. We selected them for testing purposes only.

3.5.1 The Environment

The configuration used is shown in Figure 73 on page 71.

We have two TME regions:

1. rs600015-region

2. foxtrot-region

Both regions are interconnected with a two-way connection.

In rs600015-region we have two workstations:

1. rs600015 - AIX 4.1.4

2. wtr05193 - Windows NT 4.0

In foxtrot-region we have two workstations:

1. foxtrot - Windows NT 4.0

2. romeo - Windows NT 4.0

The MSM Agent is installed in the rs600015 workstation.

The TME 10 Enterprise Console is installed in the rs600015 workstation.
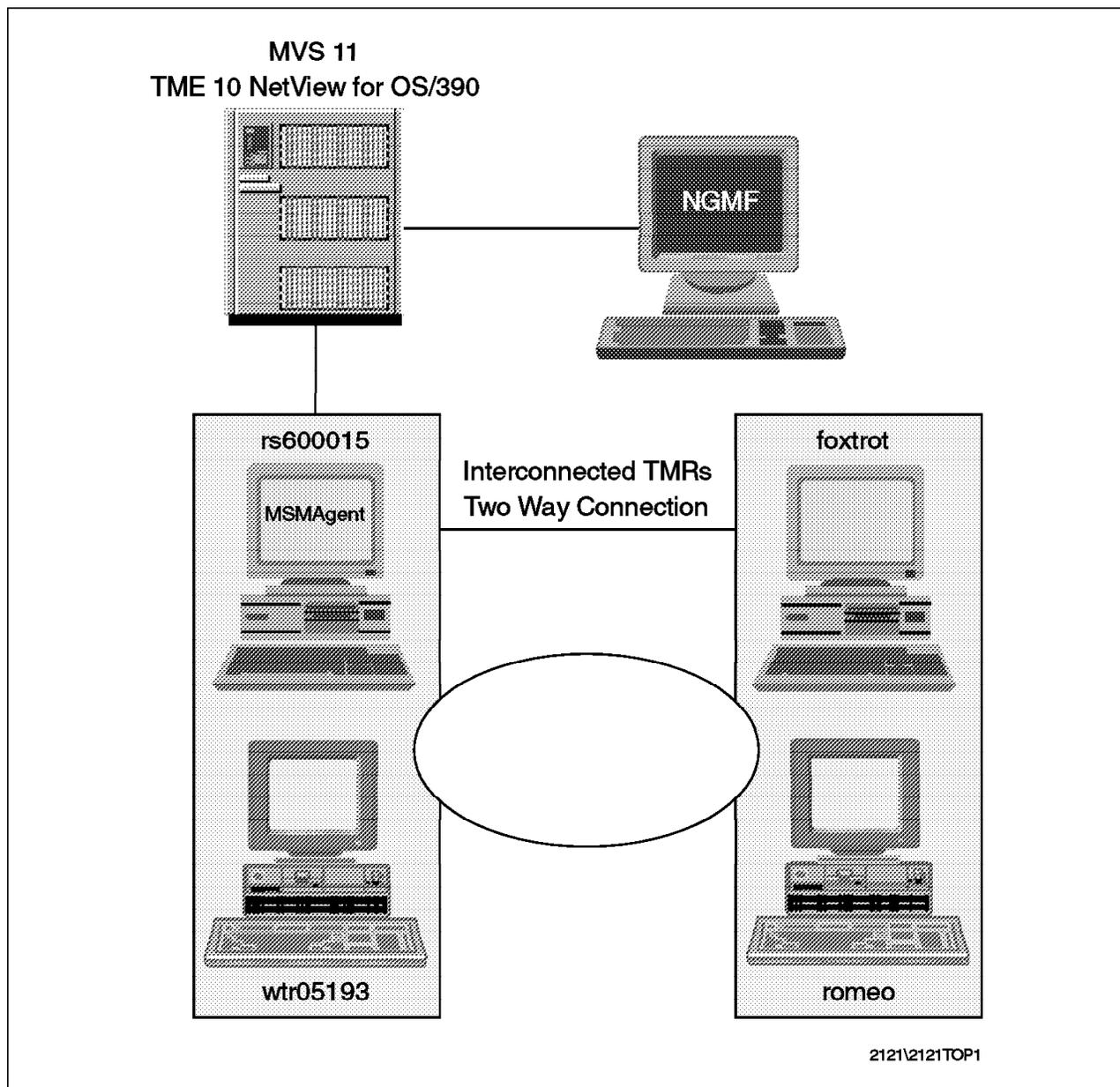
Figure 73. TME Configuration

3.5.2 The Monitors

In the rs600015-region we have created two Sentry profile managers with one TME 10 Distributed Monitoring profile in each of them.

The profiles are:

1. SentryProf

2. MSM_Color

In each of the profiles we have defined monitors from different classes. These monitors are:

1. diskusedpct(/tmp) checks the percentage of used space in the file system /tmp.

2. diskavail(/usr) checks the free disk space in the file system /usr.

3. ulogintot() checks the number of users logged in.

4. fileperm(/.profile) checks in the file .profile that the permissions are correct.

5. oserv(wtr05193) checks that the framework is started workstation on wtr05193.

6. appStatus(pool) checks if the application pool is started.

These monitors have been selected from the Universal Monitors and UNIX Monitors packages.

3.6 MSMAgent Topology in RODM

After you initialize network topology using the GETTOPO command, NGMF provides views of your TME 10 networks. Using the NGMF pull-down menus, you can navigate among the views.

The following is a list of objects that you might see in various TME 10 views:

- Aggregate objects, representing:
    - TME 10 network
    - Managed regions
    - Policy regions
    - Managed nodes
- Real objects, representing:
    - Topology agents
    - Monitor resources

3.6.1 NGMF Samples

On the NGMF workstation, we open the view defined in the TME initialization file, TME10_VIEW. The default name is MultiSysView, and we get the following screen:
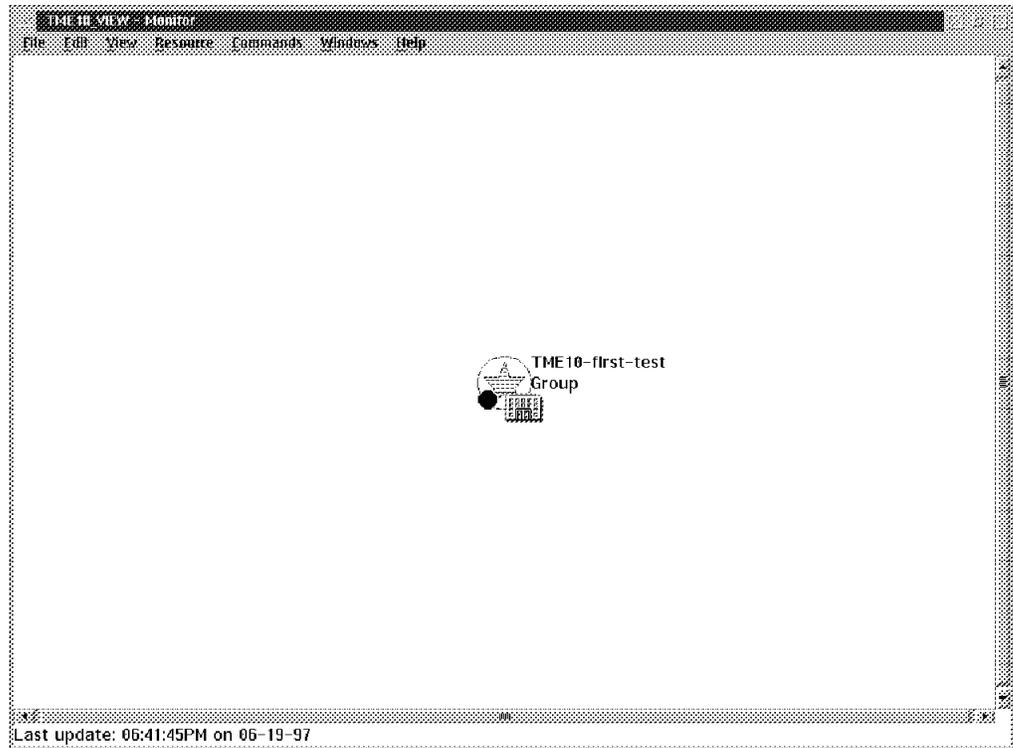
Figure 74. Top Level View

The first view we get is the TME 10 Network View. This view contains an object representing all the TME 10 networks defined in the initialization file of MultiSystem Manager. An example of the MultiSystem Manager initialization member can be found in A.2.1, "FLCSITME Used for the TME Feature of MultiSystem Manager" on page 138.

Double-click on the icon **TME10-first-test**, and you get a view representing the MultiSystem Manager Agent and the network where it is running.
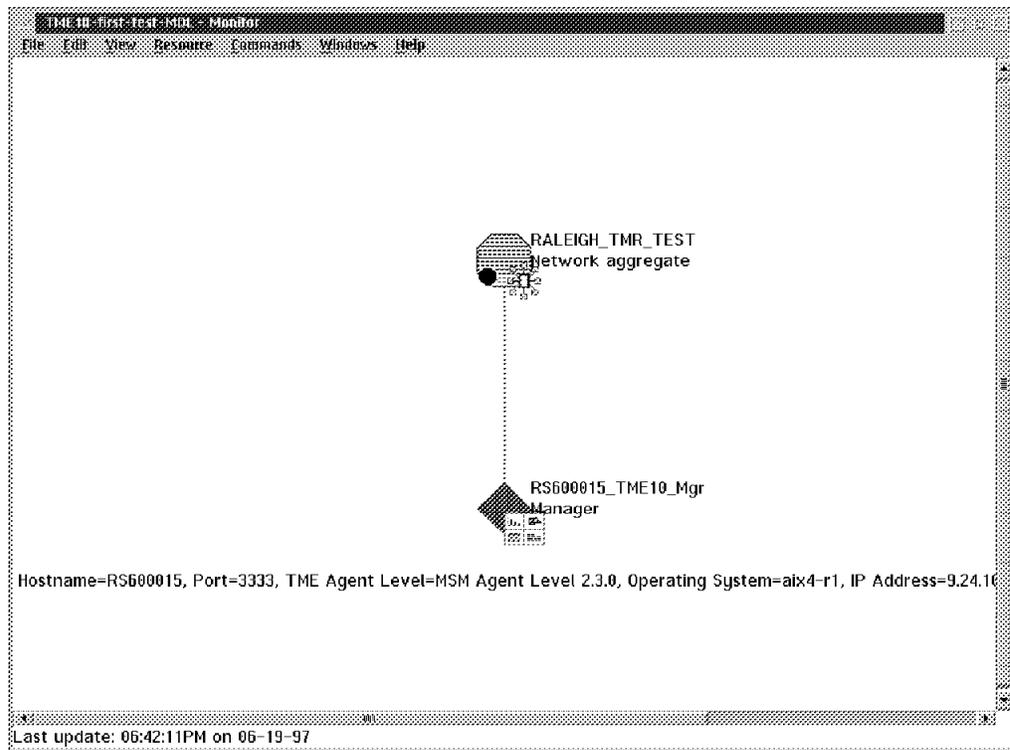
Figure 75. View of MultiSystem Manager Agent

Double-click on the icon **RALEIGH_TMR_TEST** and you get two different views.

The first view represents all the policy regions we have defined. As we are running interconnected regions, we get the policy regions of both TMRs, foxtrot and rs600015.
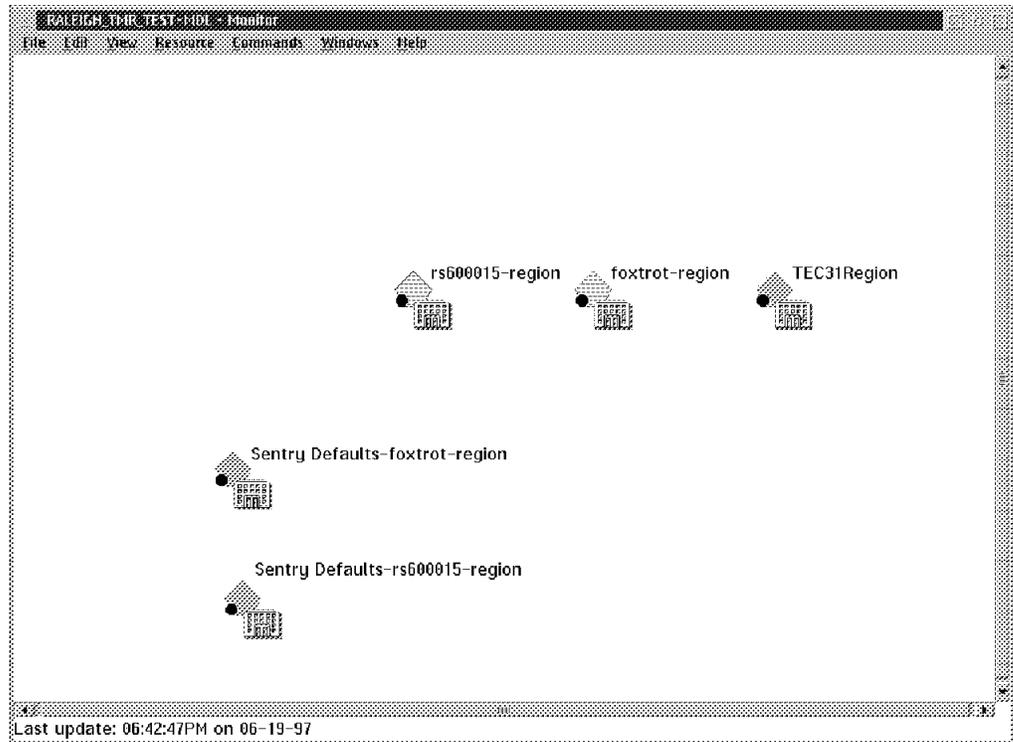
Figure 76. View of Policy Regions

The second view represents the interconnected regions.
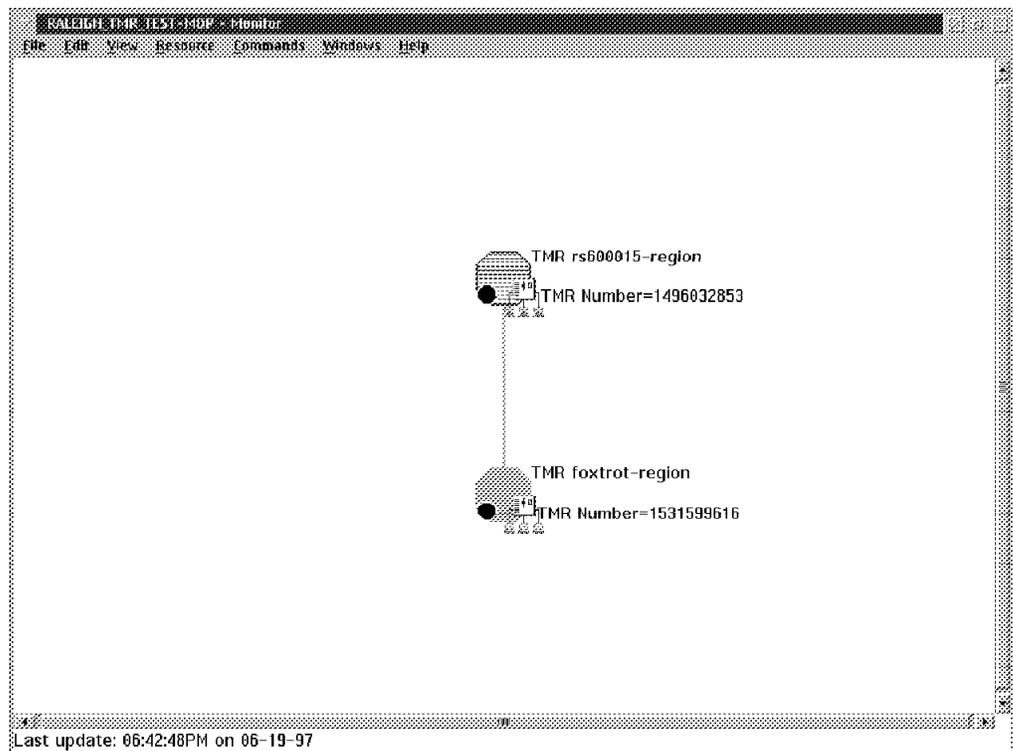
Figure 77. View of Interconnected Regions

Double-click on any of them, for instance rs600015-region and we get a view of the different managed nodes belonging to that region.

Note: PC managed nodes are not represented in views.
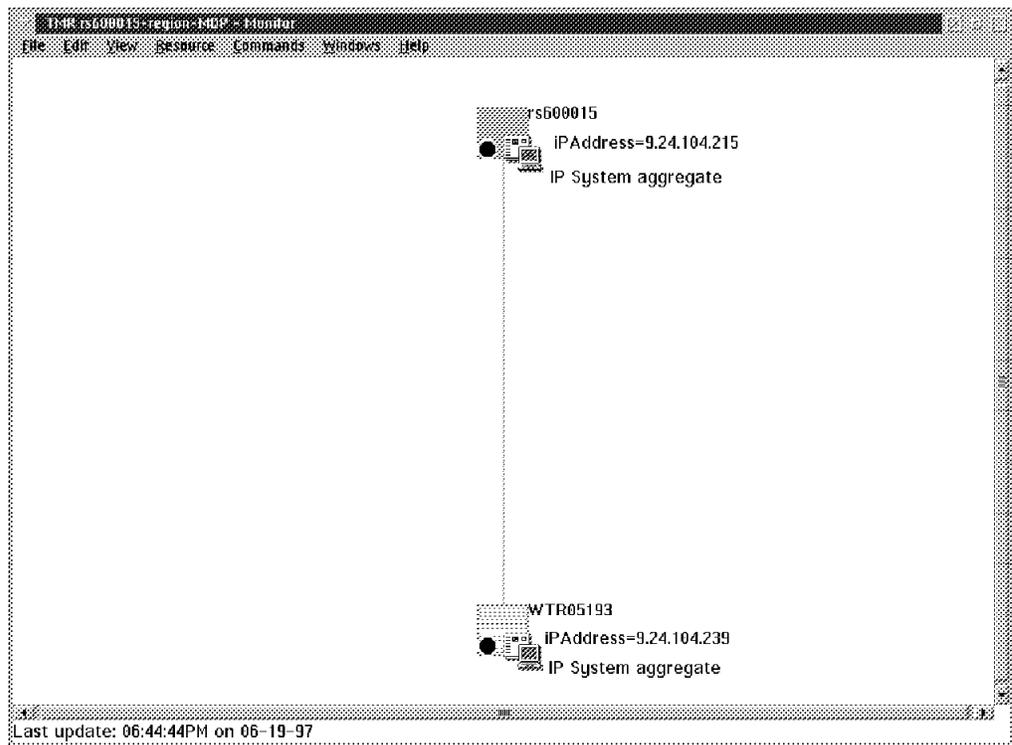
Figure 78. View of Managed Nodes

Double-click on any of the icons, for instance rs600015 and you get a view of the oserv daemon and all the monitors defined and distributed to that machine.
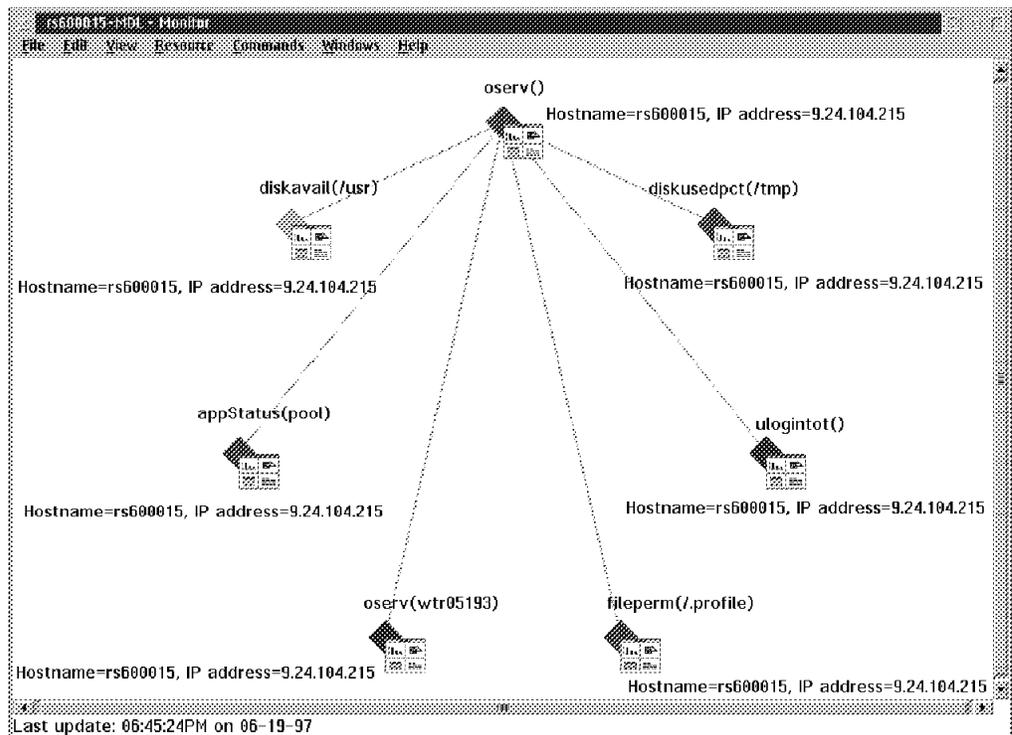
Figure 79. View of Monitors

To view all the relationships between the various objects, we created the following screen where the objects in our environment are represented and their relationships are shown.
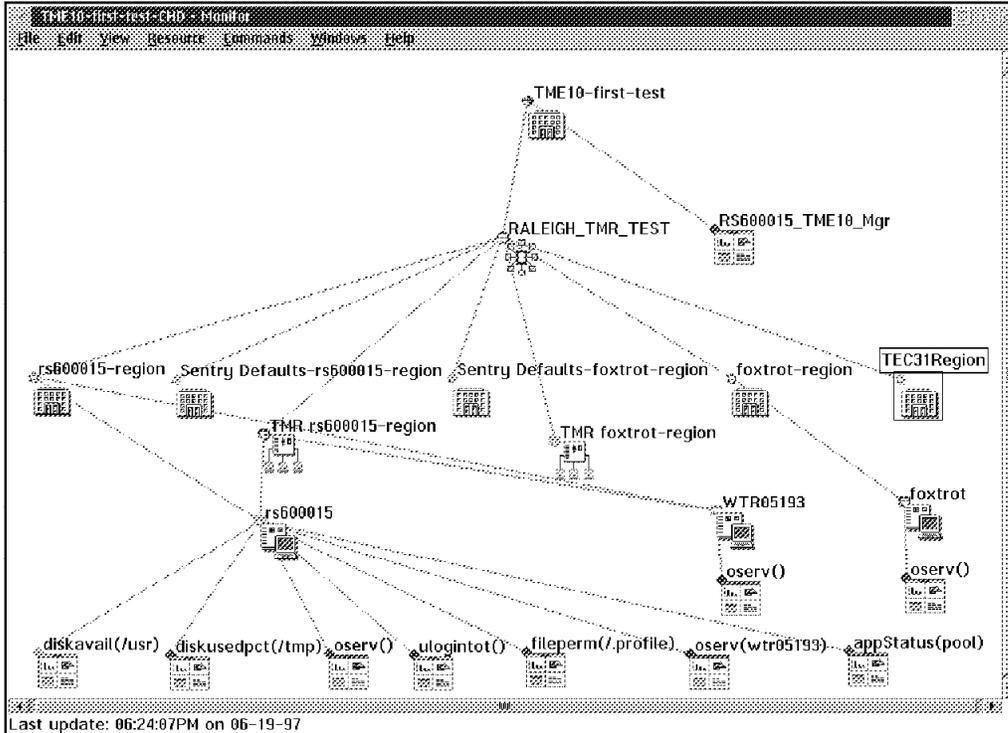
Figure 80. General Topology View

3.6.2 Exception Views

Exception views can be implemented in many different ways, to get dynamic views of failing objects. The objects go to this view when they change their DisplayStatus attribute in RODM.

In our environment, we created an exception view for all the objects in unsatisfactory status (refer to 3.2.1.3, "MultiSystem Manager Exception Views" on page 21).
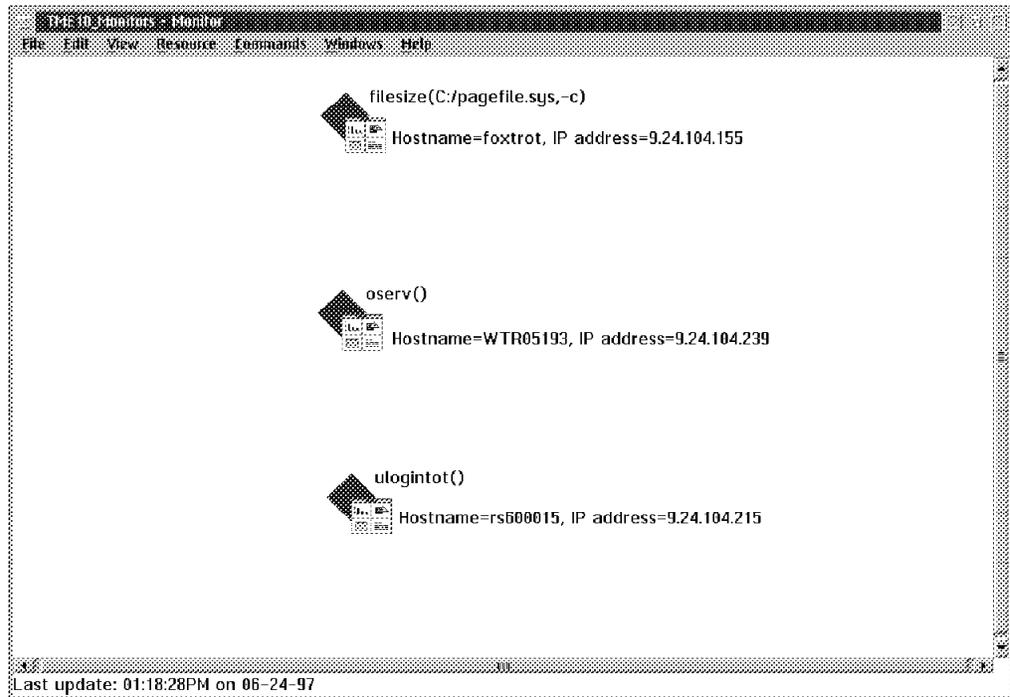
Figure 81. Exception View

The Display Name of the monitors is an abbreviated name based on the TME 10 Distributed Monitoring name of the monitor and the resource monitored.

The DisplayOtherData of the monitor is built using the IP name of the workstation, and its address. To see this field, you need to select on the **View** option and select display labels from the pull-down menu. Then check the other data from the displayed menu.

3.6.3 Resource-Specific Commands

MultiSystem Manager for TME 10 Networks provides a command interface based on Command Tree/2 that lets you select objects from the NGMF views and issue commands to these objects. The responses to the commands are returned in the Command Responses window.

Command Tree/2 is a customizable product, so you can add or delete commands from the command-specific view associated to each type of object. The following section shows you the default specific commands provided with the MultiSystem Manager for TME 10 Networks.

```
Command Tree Facility - TME 10 NetView for OS/390 resource specific commands
Edit  View  Options  Windows  Help
Networks Commands                                        Destination: NGMF API

  ▬ TME 10 resource specific commands
    └─▬ Topology manager
          ├──────REMVOBJS    (Remove objects)
          ├──────RESTOPO     (Resume topology manager)
          ├──────SETREMV     (Set removable attributes)
          └──────SUSPTOPO    (Suspend topology manager)
```

Figure 82. Group Commands Associated with TME 10 Network Objects

```
Command Tree Facility - TME 10 NetView for OS/390 resource specific commands
Edit  View  Options  Windows  Help
Manager Commands                                         Destination: NGMF API

  ▬ TME 10 resource specific commands
    ├─▬ Operating system commands
    │     ├──────Command line
    │     ├──────env          (Display system environment information)
    │     ├──────uname        (Display operating system information)
    │     └──────uname -a     (Display current operating system information)
    ├─▬ MSM agent commands
    │     ├──────FLCT_viewDb.pl    (Get state of agent's local topology database)
    │     └──────FLCT_unlockDb.pl  (Reset the lock on agent's local topology database)
    ├─▬ Platform commands
    │     ├──────odadmin        (Display Tivoli configuration)
    │     ├──────odadmin odlist list_od   (Get description of managed nodes in TMR)
    │     ├──────wlsconn        (List connected TMRs)
    │     ├──────wlseng         (List contents of the Sentry engine)
    │     └──────wlsinst -a     (List installed products and patches)
    └─▬ Topology manager
          ├─▬ GETTOPO
          │     ├──────TMEONLY    (TME agent only)
          │     └──────TMERES     (Service point system and managed resources)
          ├──────REMVOBJS    (Remove objects)
          ├──────RESTOPO     (Resume topology manager)
          ├──────SETREMV     (Set removable attributes)
          └──────SUSPTOPO    (Suspend topology manager)
```

Figure 83. Manager Commands Associated with MSM Agent Object
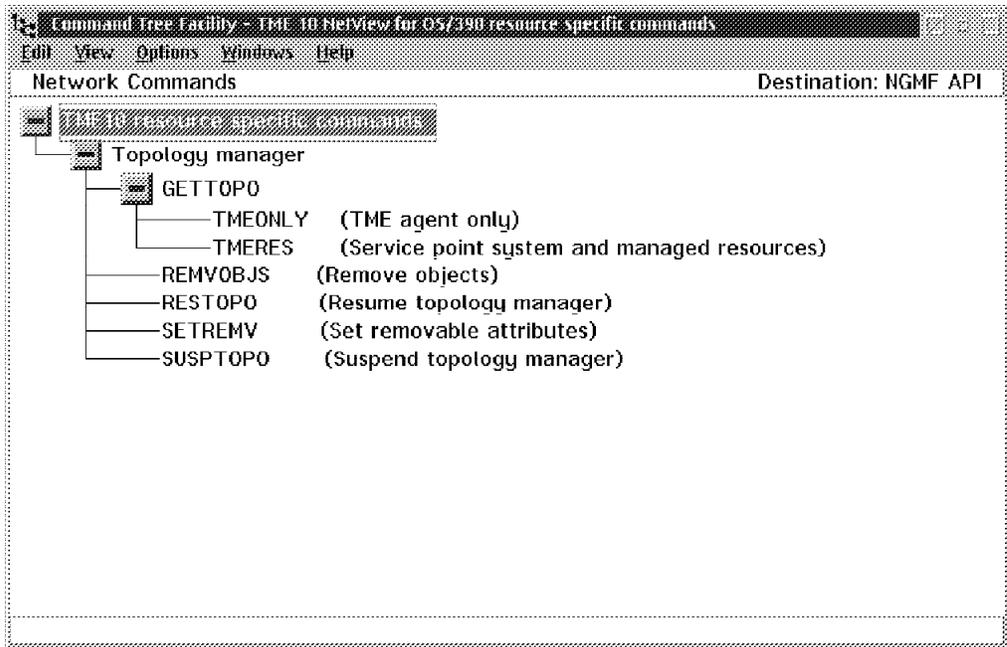
Figure 84. Network Aggregate Commands Associated with Network Objects
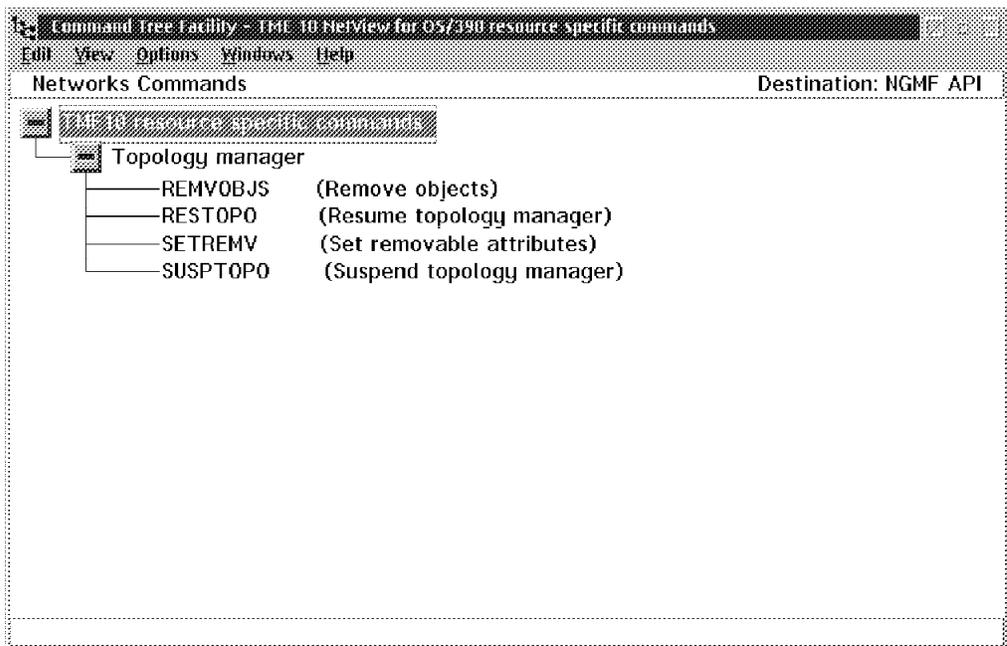
Figure 85. Policy Region Aggregate Commands Associated with Policy Regions Objects

```
Command Tree Facility - TME 10 NetView for OS/390 resource specific commands
Edit  View  Options  Windows  Help
Managed_Region_Aggregate Commands                          Destination: NGMF API

   TME 10 resource specific commands
      Operating system commands
         Command line
         env          (Display system environment information)
         uname        (Display operating system information)
         uname -a     (Display current operating system information)
      Platform commands
         odadmin      (Display Tivoli configuration)
         odadmin odlist list_od   (Get description of managed nodes in TMR)
         wlsconn      (List connected TMRs)
         wlseng       (List contents of the Sentry engine)
         wlsinst -a   (List installed products and patches)
      Topology manager
         REMVOBJS     (Remove objects)
         RESTOPO      (Resume topology manager)
         SETREMV      (Set removable attributes)
         SUSPTOPO     (Suspend topology manager)
```

Figure 86. Managed Regions Aggregate Commands Associated with Managed Regions Objects

```
Command Tree Facility - TME 10 NetView for OS/390 resource specific commands
Edit  View  Options  Windows  Help
Monitor Commands                                           Destination: NGMF API

   TME 10 resource specific commands
      Operating system commands
         Command line
         env          (Display system environment information)
         uname        (Display operating system information)
         uname -a     (Display current operating system information)
      Platform commands
         wlseng       (List contents of the Sentry engine)
      Topology manager
         REMVOBJS     (Remove objects)
         RESTOPO      (Resume topology manager)
         SETREMV      (Set removable attributes)
         SUSPTOPO     (Suspend topology manager)
```

Figure 87. Monitor Commands Associated with Monitors Objects

3.6.4 Status and Severity

The initial status of the objects depends on the last event received in T/EC. For the monitors that do not have an associated event, the default initial status is UNKNOWN (gray). But, we consider that a monitor that does not have any associated event is in a satisfactory condition, so we changed the parameter

INITIALSTATUS=HARMLESS to set the monitors to green (refer to B.1.1, "MSM Configuration File - MSMAgent.cfg" on page 143).

3.6.5 Stopping and Restarting the MSM Agent

When stopping the MSM Agent, an MSM Agent Down alert comes to NPDA in TME 10 NetView for OS/390. This alert is captured in the automation table, and all the objects in the TME 10 network go to UNKNOWN.

When starting the MultiSystem Manager Agent, an MSM Agent Up alert comes to NPDA in TME 10 NetView for OS/390. This alert is captured in the automation table, and the GETTOPO command is issued, as coded in the FLCSITME member of DSIPARM.

3.6.6 Others Events Coming from the Agent

As we told you earlier in 3.3.3.4, "Status of Managed Nodes" on page 57 the MultiSystem Manager Agent will send events related to status changes of managed nodes. These events are converted to alerts by the Global Enterprise Manager Event adapter and stored in NPDA. The following sections show the various alerts received from FLCT_updateMNStatus.pl.

3.6.6.1 Topology Changes

When the topology configuration changes because a managed node has been added or deleted, or because a connection to an interconnected region has been enabled or disabled, an alert is generated to inform you about the changed topology.
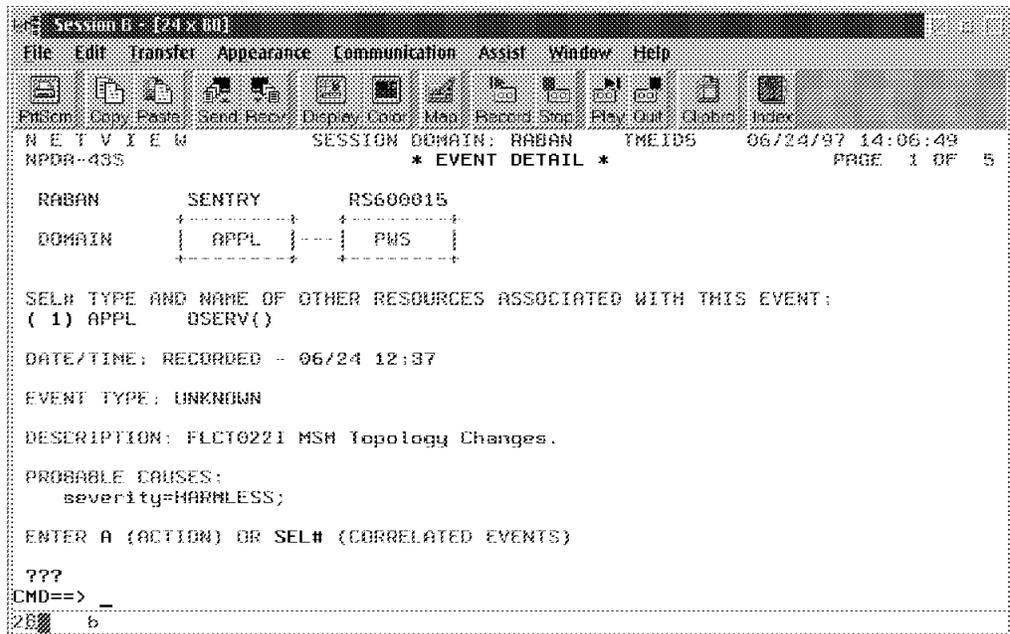
Figure 88. Topology Changes Alert

Adding or deleting a monitor is not considered a change of topology so no alert is generated for this kind of change.

3.6.6.2 Managed Node Added

When a managed node is added to the framework, it will be discovered by
FLCT_updateMNStatus.pl. An event will be sent to T/EC, and then forwarded to
TME 10 NetView for OS/390. The alert is captured in the automation table, and a
GETTOPO is issued to add the new managed node to the NGMF view.

Figure 89. Managed Node Added

3.6.6.3 Managed Node Deleted

When a managed node is deleted from the framework, the topology change will
be discovered by FLCT_updateMNStatus.pl. An event will be sent to T/EC and
then forwarded to TME 10 NetView for OS/390. The alert is captured in the
automation table, and a GETTOPO is issued to delete the managed node from
the NGMF view.
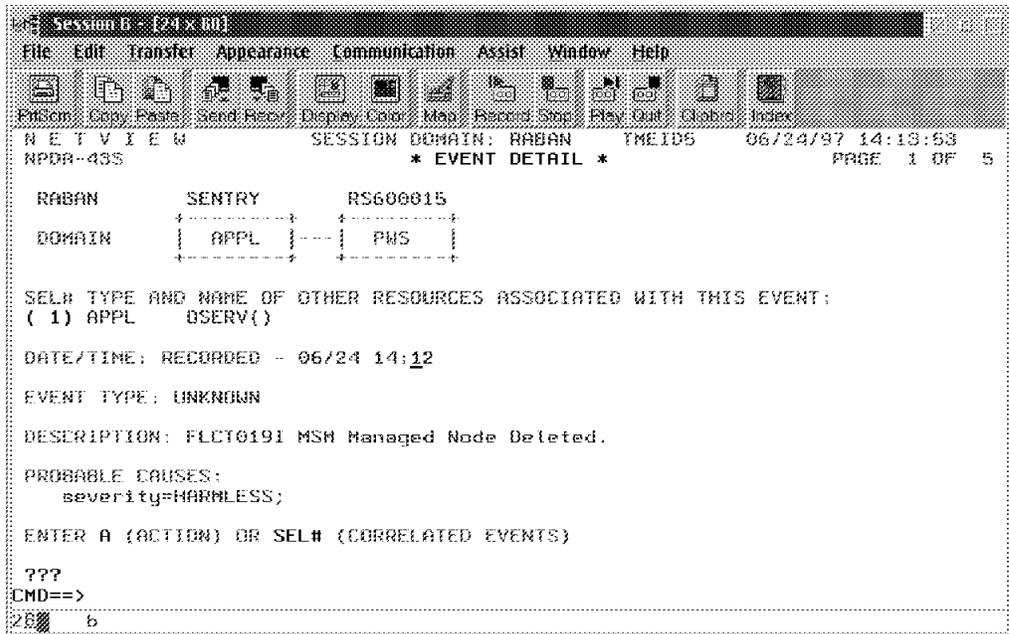
Figure 90. Managed Node Deleted

3.6.6.4 Managed Node Status Change

When the status of a managed node changes from up to down or vice versa, an event is generated, and when it arrives at TME 10 NetView for OS/390 as an alert, the status of the oserv() monitor changes.
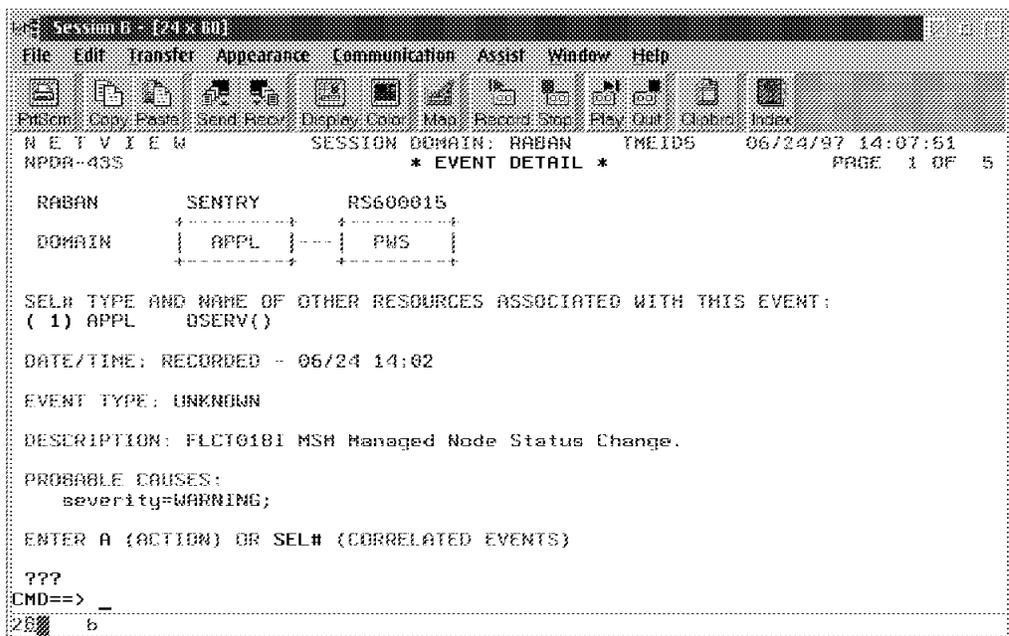
Figure 91. Managed Node Status Change

3.7 Advanced Scenario

The rules shipped with Global Enterprise Manager are samples and should be modified by the customers or the service provider as part of the installation. We found that the rules shipped could cause some unwanted situations if used as shipped. Therefore, this section explains another example that we created to show how the supplied samples rules can be modified and extended.

The assumption for this scenario was that the operators are using NGMF consoles and cannot manually close events on the T/EC console. Therefore, we needed a mechanism to close the events without operator intervention. In addition we wanted to create a rule that filtered duplicate events to minimize the number of events that flow between Tivoli Enterprise Console and TME 10 NetView for OS/390.

3.7.1 Reason for This Scenario

As we told you previously, MultiSystem Manager collects information from two different sources.

At initialization time information is gathered about the managed nodes and their status. The TME 10 Distributed Monitoring engine provides information about monitors defined and their status.

After that, only T/EC events change the status of the monitors in RODM.

MultiSystem Manager provides a rule to forward events to TME 10 NetView for OS/390. This rule is an example, and using it in combination with the sample provided for the Event/Automation Service can cause undesired situations. The MultiSystem Manager rule is called forwardSentry.rls and the Event/Automation rule is called tecad_nv390fwd.rls. You can find these sample rules in B.2.3, "Forwarding of Sentry Events to NetView/390 - forwardSentry.rls" on page 156 and B.2.4, "Forwarding of Events to NetView/390 - tecad_nv390fwd.rls" on page 156.

The forwardSentry rule works like this:

 1. Forwards any TME 10 Distributed Monitoring event to TME 10 NetView for OS/390.

 2. Changing the status of a TME 10 Distributed Monitoring event on the T/EC console will forward the event to TME 10 NetView for OS/390

    If your TME 10 Distributed Monitoring event is CRITICAL or FATAL and you close it on the T/EC console, it will be forwarded by the MultiSystem Manager rule. Since the event was CRITICAL or FATAL the sample Event/Automation rule will by default also forward it. That means that two events are forwarded and accordingly two resolution alerts will be created by the alert adapter for the same problem.

If you applied these sample rules in a production environment, they would cause an excessive flow of events to TME 10 NetView for OS/390 since duplicated events are not filtered in fowardSentry.rls.

For example, if you have defined a monitor that starts every five minutes, and have a non-desired situation in that monitor, you get an event and an alert every five minutes. The duplicated events will not provide any new information. They just confirm that the undesired situation still exists.

These situations can be avoided given the assumptions described apply to your situation by doing the following.

3.7.2 Setting Up TME 10 Distributed Monitoring Monitors

To provide an automatic mechanism to change the color of the monitor on NGMF we defined the appropriate response levels in TME 10 Distributed Monitoring and mapped them to appropriate T/EC severities for undesired situations. In each TME 10 Distributed Monitoring monitor we also defined a NORMAL response level that was mapped to a HARMLESS T/EC event severity. Doing this will assure that we receive T/EC events for error conditions and also normal conditions.

3.7.3 Setting Up the Rules

We also need to modify the sample rules to do filtering and handle other special situations. The rule we created only applies to TME 10 Distributed Monitoring and was created to handle these situations:

1. Forward any new event.

2. Discard all duplicate events.

3. If a duplicate event is received but with a new severity, close the previous one, and send the new one to TME 10 NetView for OS/390.

4. Forward any event related to the MSMAgent class that is, topology changes.

We created two new rules to accomplish this. Our rules with comments can be found in B.2.5, "ITSO Modified Rule - tecad_nv390fwd_itso.rls" on page 158 and B.2.6, "Sample T/EC Rule for MSM Used in ITSO - new_Sentry.rls" on page 160.

We applied these changes with a modified gemevent.sh that we called itso_gem.sh. This shell script is provided with comments in B.4.2, "Modified Installation Shell Script - itso_gem.sh" on page 188. We tested these rules and found that they worked well. The first event for each monitor is sent to T/EC and forwarded to TME 10 NetView for OS/390. Additional events will be sent to T/EC but if the new event is the same in terms of severity as the existing one it will be discarded. Accordingly no event will be sent to TME 10 NetView for OS/390.

If the new event received has a different severity from the existing event, the existing event will be closed automatically but the closed event will not be forwarded to TME 10 NetView for OS/390. The new event will be forwarded to TME 10 NetView for OS/390 and the alert received in NPDA. The severity of the new event will determine the color of the object on NGMF as described earlier.

As we said earlier, an exception is events related the MultiSystem Manager Agent. These events inform of changes in the topology or in the status of the managed nodes. The new rule forwarded all events for the MSMAgent class to TME 10 NetView for OS/390.

If the communication between TME 10 NetView for OS/390 and the MSMAgent goes down and is later recovered, the GETTOPO will cause NGMF to reinitialize the objects with the last T/EC event received. This means that as long as the event server is running, the rule will continue to work ensuring that we don't loose any information while communication is down.

3.7.4 Some Usage Considerations

In an environment with multiple interconnected regions there are several things to consider:

- Only one MultiSystem Manager Agent is needed.

- MultiSystem Manager Agent must be in a TMR with communication to the rest of the TMRs, typically where the event server is installed.

- Always interconnect regions with two-way connections.

  This will allow commands to be sent from the region where MultiSystem ManagerAgent resides to the remote TMR, and allow the remote regions to send events to the central event server.

- Register the event server in the remote TMR

  In remote interconnected regions execute the commands needed for sending T/EC events to the Event Server. These commands are:

  - wregister -i -r EventServer

  - wupdate -r EventServer rs600015

- Define the MSMAgent user ID in managed nodes.

  You must define a user ID named MSMAgent, with the right Tivoli permissions to execute commands, in every managed node that you want to send commands to.

3.8 Using a Secure TCP/IP Connection

The Topology Service supports a secure TCP/IP connection between TME 10 NetView and the MSMAgent. The secure TCP/IP connection is available with the PTF for OW29311 so this is just a summary of this function.

3.8.1 Secure TCP/IP connection on AIX

The secure TCP/IP connection works the same way as the secure TCP/IP connection used for the User Administration service. That means that a passticket key must be generated and used when the connection is established.

- OS/390 Connection Service

  The OS/390 Connection Service is a prerequisite on the AIX machine so using the install instruction described earlier you can install it from the Global Enterprise Manager CD.

- ihstukey command

  This command is located in /usr/local/Tivoli/bin/aix-r1/TME/MVSENDPOINT. It is used to create a file with the seed key that is used to establish the secure IP connection. The following is the command we used:

  ihstukey -f /tmp/msmkey -k 13579BDF2468ACE0.

  This creates a file called msmkey that contains the key as specified in the -k option.

- ihstsmsm command

  This command is located in /usr/local/Tivoli/bin/aix-r1/TME/MVSENDPOINT. It is used to start the MSMAgent when using the secure IP connection. On our AIX machine we issued:

```
ihstsmsm -k /tmp/msmkey -p 3333 -u TMEID1
```

The parameters used are:

− -k /tmp/msmkey

   This is the file we created with the ihstukey command in the previous
   step.

− -p 3333

   This is the port we want the MSMAgent to use. We used port 3333 which
   is the same port we used for the unsecure connection. This must match
   the port specified on the host as described earlier.

− -u TMEID1

   This is the NetView user ID authorized to communicate with the
   MSMAgent.

You will see message FLCT009I MSM Agent Up after you issue this command as
well as an MSM Agent Up event.

3.8.2 Secure TCP/IP Connection on MVS

When using a secure IP connection, MSM invokes a TP server host DLL. This
host DLL must be available for MSM to work.

You must add the library where you installed it to STEPLIB in your NetView
procedure. The TP server address space does not need to be started.

The required RACF definitions were added with this job:

```
//RACDEFS  JOB '222222','TMEID8',
//         CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*********************************************************
//* MAKE THE TME10 RACF DEFINITIONS
//* RUN UNDER A RACF SPECIAL USER
//*********************************************************
//*
//RACDEF  EXEC PGM=IKJEFT01,REGION=4096K,DYNAMNBR=20
//*
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//*
//SYSTSIN  DD *
 SETR CLASSACT(PTKTDATA)
 RDEF PTKTDATA MSMAGENT SSIGN(KEYMASKED(13579BDF2468ACE0))
 SETR RACLIST(PTKTDATA) REFRESH
```

The FLCSITME job creates a profile called msmagent in the ptktdata class and
refreshes this class to activate the new definitions.

You must update your FLCSITME member to include the SECURE=YES
parameter.

If the PTF for APAR OW29311 is not installed you will get a DSI002I message
saying that FCLACSAF is an invalid command when you issue the INITTOPO or
GETTOPO command.

Chapter 4. Installing and Customizing the Command Service

This chapter explains the Global Enterprise Manager Command Service as delivered in the Global Enterprise Manager. It enables the customer to send commands from an APM client to TME 10 NetView for OS/390. It also provides command line support with the new hostcmd command.

4.1 Introduction

As part of an overall management policy of your system management environment the customer should be able to send commands to any platform and from any platform regardless of the manager-client relationship. The MultiSystem Manager component of TME 10 NetView for OS/390 provides a way to send commands from TME 10 NetView for OS/390 to selected resources in the distributed environment. This is described in detail in 3.3.3.6, "Sending Commands to TME Resources from NetView/390" on page 58. If you are managing your business systems with the Application Policy Manager (APM), you need a vehicle to send commands from APM to TME 10 NetView for OS/390. With the command service of Global Enterprise Manager this can now be accomplished.
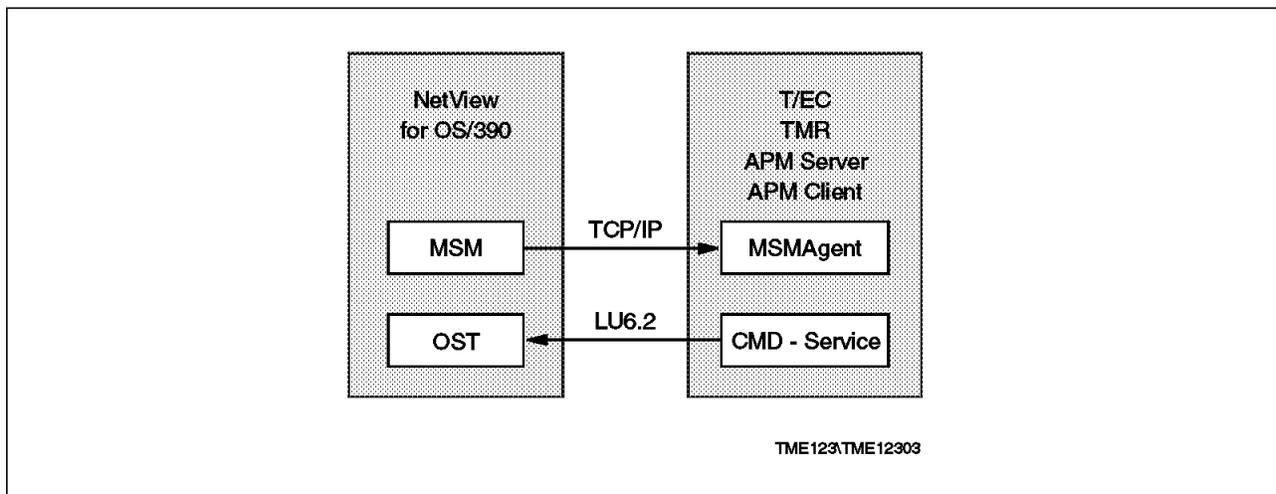
Figure 92. Command Service Data Flow

In the following paragraphs we show how to set up and use the command service which flows from the APM server to the TME 10 NetView for OS/390 host.

4.1.1 Hardware and Software Requirements

On the OS/390 side you will have TME 10 NetView for OS/390 as a requirement. You have a choice to set up an LU 6.2 session or a TCP/IP session between the APM server and TME 10 NetView for OS/390.

On the workstation side you must have the APM server installed.

The above picture implies that T/EC, the Application Policy Manager server and Application Policy Manager client can run in the same machine. We want to emphasize that this is not recommended in a production environment.

4.1.2 Software Installation on S/390

At the OS/390 it is assumed you have done just the basic TME 10 NetView for OS/390 installation and customization as described in the redbook *An Introduction to TME 10 NetView for OS/390*, SG24-4922.

4.1.3 Software Installation on the Workstation

The configuration described earlier was changed during the residency. We had the following configuration for our work with the Command Service:
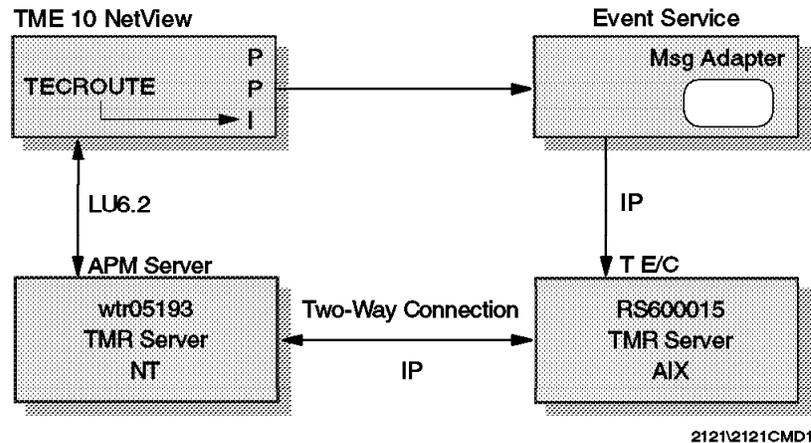
Figure 93. ITSO Configuration for the Command Service

We now have the APM server installed on an NT workstation called wtr05193. This new TMR server was interconnected to the existing TMR on RS600015 with a two-way connection.

For communications between the APM server and NetView/390 we used Communications Server for NT Version 5. We set up an LU 6.2 connection, based on the sample provided by the APM server. The configuration file we used is provided in B.1.3, "Communications Server for Windows NT" on page 146.

4.1.3.1 Application Policy Manager Server Customization

After installing Application Policy Manager server, we only need to customize one configuration file, ihsshstc.cfg, used to define a NetView operator ID and password used by the Command Service to send commands to TME 10 NetView for OS/390.

```
*******************************************************************************
*
* Licensed Materials - Property of IBM
* 5697-B83
* (C) Copyright IBM Corp. 1997.  All rights reserved.
*
* US Government Users Restricted Rights - Use, duplication or
* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*
*******************************************************************************
*
* PRODUCT          (TME 10 GEM APM)
*
* MODULE_NAME      (IHSSHSTC.CFG)
* DESCRIPTIVE_NAME (Configuration file for Command Exit
*                   that sends commands to NetView/390)
*
*******************************************************************************
*
* Function:
*   Defines the NetView 390 Operator id and password that will execute the
*   NetView 390 command string passed as input arguments to command exit
*   IHSSHSTC.
*
*   Keywords supported are:
*
*     "OPER_ID="  Defines the NetView 390 Operator id
*     "OPER_PW="  Defines the NetView 390 Operator password (optional)
*
*   Notes:
*
*     - The password is optional, when the NetView 390 Operator id does
*       not require a password.  In this case, either the keyword can be
*       removed from this file, or by specifying nothing after the "=".
*     - There must NOT be a space between the keyword and the "=".
*     - There optionally can be spaces between the "=" and the value.
*     - Comment lines are supported, which must have an '*' in column 1.
*     - Blank lines are ignored.
*     - Lines with unrecognized keywords are ignored.
*     - When the same keyword is repeated on multiple lines, the last one is
*       the one used.  The others are ignored.
*
*******************************************************************************

OPER_ID=TMEID2
OPER_PW=TMEID2
```

Figure 94. Configuration File ihsshstc.cfg

4.1.4 Operational Example

Command Services is used to send NetView commands from the APM server or the APM clients, by using the hostcmd command line interface, or the APM client GUI. In our scenario, we used the hostcmd command for MVS message automation.

The example we provide should only be considered as an example of how to set up the environment, and how Command Services can be integrated into a more complex scenario, where some of the automation tasks are performed from T/EC.

4.1.4.1 Automation Scenario

The MVS automation scenario we built is used to restart an MVS task that has been previously stopped.

For automation purposes, we recommend you build a rule that traps the T/EC event, and then start a task in the Application Policy Manager server that sends a command to TME 10 NetView for OS/390. The response to the command can be stored in a file. In our scenario, we started that task manually.
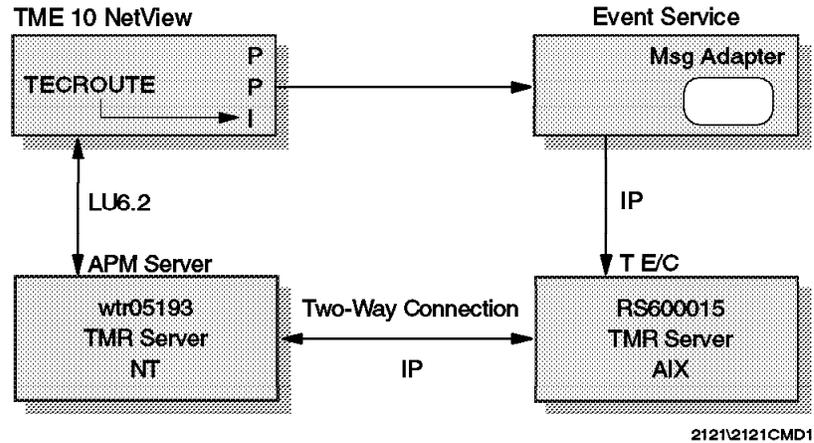
Figure 95. Command Services Data Flow

1 When TSO stops, the message IEF404I TSO - ENDED -... is captured in the NetView automation table and sent to the Event/Automation service.

2 In the Event/Automation service, the message is converted into a T/EC event.

3 The event arrives to T/EC, and then goes to the NV390MSG Console.

4 We manually start a task from T/EC that executes a program in the remote workstation where the APM server is running. As we said earlier this could be done automatically by coding a T/EC rule.

5 The program executes the hostcmd "mvs s tso" command.

6 This command flows through the LU 6.2 connection to NetView, with the permissions of the operator defined in the ihsshstc.cfg file.

7 The command is executed in NetView, and the response goes back to the APM server.

8 The response is then sent back to the task that started the program, and goes to a file or to a screen, as we had defined in the task.

4.1.4.2 APM Server Startup

The following operational procedures apply to an Application Policy Manager server installed on a Windows NT workstation. To get an operational environment between the Application Policy Manager server and the TME 10 NetView for OS/390, we first start the Application Policy Manager server, and then start the communications with the NETCONV command.

To start the APM server, go to the Application Policy Manager server directory. In our case it was located in D:/Tivoli/bin/w32-ix86/GEM/ApmServer/bin. Run the apmserver command.

Log on to NetView/390 with the same operator ID defined in the configuration file ihsshstc.cfg and run the NetView/390 command NETCONV ACTION=START LU=RA600150. The LU should be the independent LU defined in your communication server configuration file. This will start the communication between Application Policy Manager and TME 10 NetView for OS/390.

You could also use the NETCONV ACTION=START IP=WTR05193 command to establish a TCP/IP connection between Application Policy Manager and TME 10 NetView for OS/390. You would need to enable the TME 10 NetView for OS/390 support by changing the DUIFPMEM member as shown below:

```
* TCP/IP related initialization statements
* Allow communication via use of TCP/IP connections.
* The default is 'NO'.
* Accepted values are Y, N, YES, and NO.
* This keyword is optional.
USETCPIP = YES
* The TCP/IP address space name.
* This keyword is optional.
TCPANAME = T11ATCP
* The number of simultaneous netconv sessions.
* This keyword is optional.
SOCKETS  = 100
* The maximum wait time for IP communication.
* The value specified is in seconds.
* This keyword is optional.
TIMEOUT  = 60
* The port number on which the workstation server is listening.
* This keyword is optional.
PORTNUMBER = 4020
```

To verify this installation, we can send a NetView command from a DOS Window on the NT machine. As indicated in the picture you should be in the Application Policy Manager server directory when you do this. We run the command hostcmd "dis ra600150" and the command response is received in the same DOS window:

Figure 96. hostcmd Command and Response

4.1.4.3 Creating a Task

We decided to create a task that could be executed from T/EC and could execute the command to TME 10 NetView for OS/390 from the Application Policy Manager server.

As tasks are designed to execute programs more than commands, we needed to create a program that contained the command we wanted to execute from the Application Policy Manager server. We created a copy of the Global Enterprise Manager supplied hostcmd and named it cmd2host.bat and put it in the D: directory. In this file we executed two commands: ihsshstc "dis rabat" to check the status of the TSO VTAM application and ihsshstc "mvs s tso" to start the TSO application.

```
@rem ************************************************************************
@rem * Licensed Materials - Property of IBM
@rem * 5697-B83
@rem * (C) COPYRIGHT Tivoli Systems, an IBM Company 1997
@rem * All rights reserved
@rem *
@rem * US Government Users Restricted Rights - Use, duplication or
@rem * disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
@rem ************************************************************************

@rem ************************************************************************
@rem * Set up environment to run the Command Service which will run
@rem * a TME10 NetView for OS/390 command.
@rem *
@rem * Leave no spaces around the equal signs
@rem ************************************************************************
     @setlocal

     @call %windir%\system32\drivers\etc\Tivoli\setup_env.cmd

     @echo BINDIR is set to %BINDIR%

@rem ************************************************************************
@rem * Set the PATH environment variable
@rem ************************************************************************
 @SET PATH=%PATH%;%BINDIR%\gem\apmserver\bin;%BINDIR%\gem\apmserver\msg_cat\en_US

@rem ************************************************************************
@rem * Run IHSSHSTC which will send the command string to NetView for OS/390
@rem ************************************************************************

@rem * First we check the status of TSO application
     @IHSSHSTC "dis rabat"
@rem * Second we start TSO application
     @IHSSHSTC "mvs s tso"

     @endlocal
```

Figure 97. CMD2HOST File Used to Execute IHSSHSTC

This program can be created in any managed node (not necessarily in the Application Policy Manager server workstation), because the tasks get the contents of the program at creation time, and so, at execution time the program does not need to be in the remote workstation.

Now, using the TME 10 Framework GUI in the managed node where the T/EC is running, we created a task that should execute this program on wtr05193.

To create a task, we select the option **Create Task** from the pull-down menu in an existing task library.
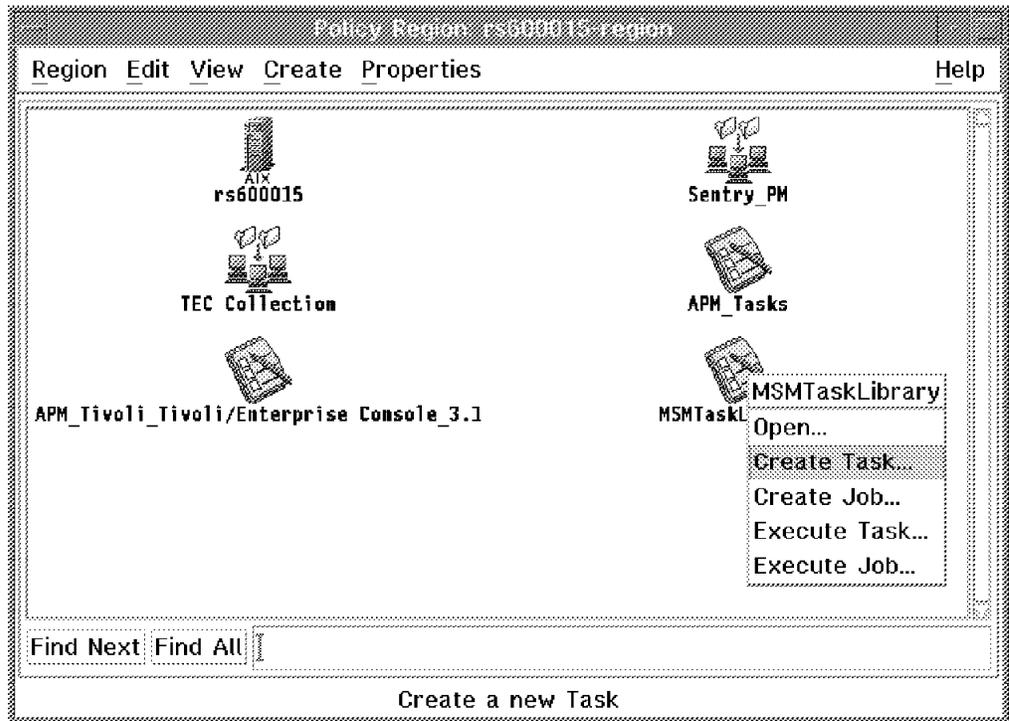
Figure 98. Create Task Dialog

We named the task hostcmd. This task will require a role of user to be executed, and will run on a Windows NT workstation.
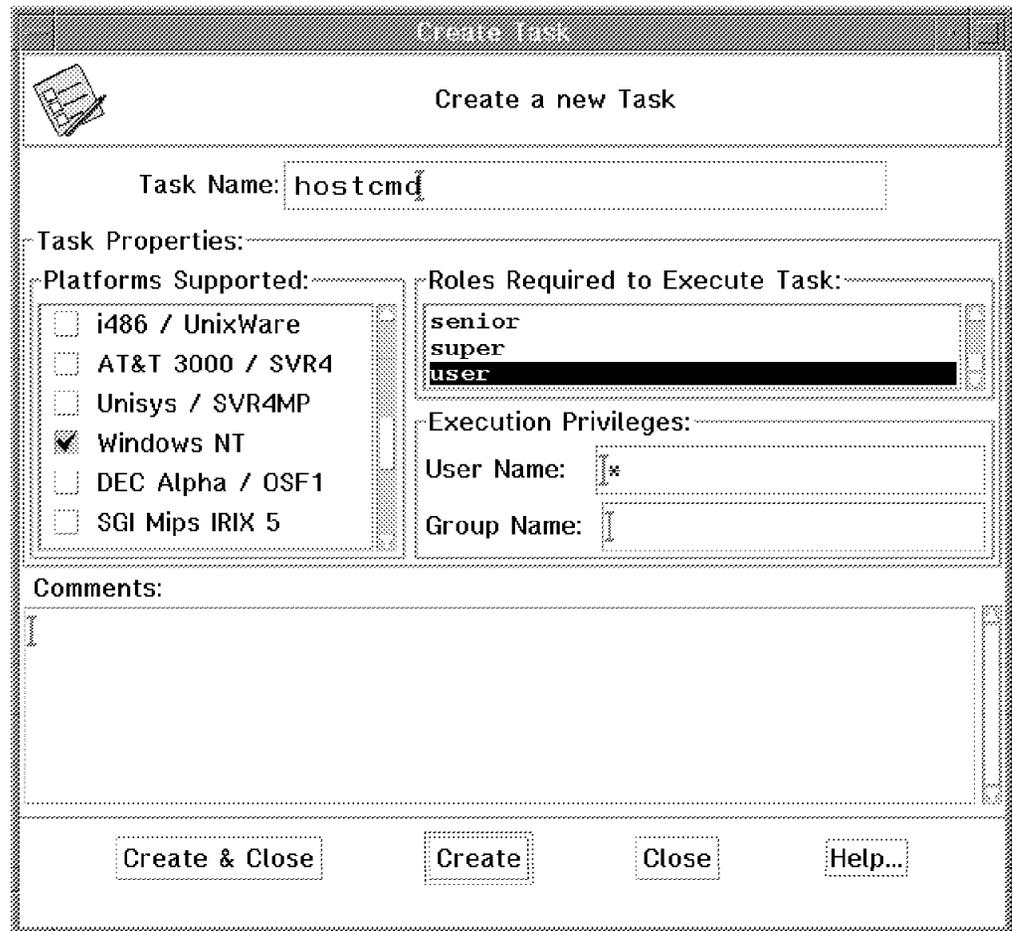
Figure 99. Create Task Window

When we checked the Windows NT box, a new window appeared prompting us for the file we want to execute, and the node where it resides. Fill the window with the required information, and select the **Set & Close** button to exit.
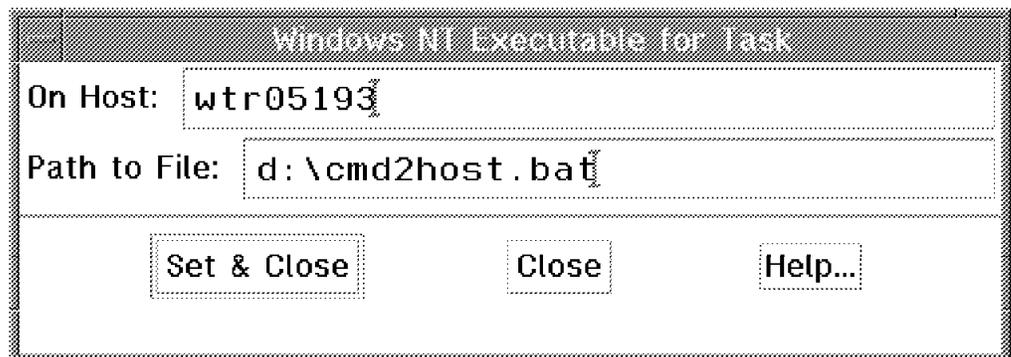
Figure 100. Path to File Window

Then click **Create & Close** from the Create Task window.

4.1.4.4 T/EC Scenario

We open the T/EC Console NV390MSG, and look for the message IEF404I TSO - ENDED - .... We select the **Task** option in the menu bar and then **Execute**.
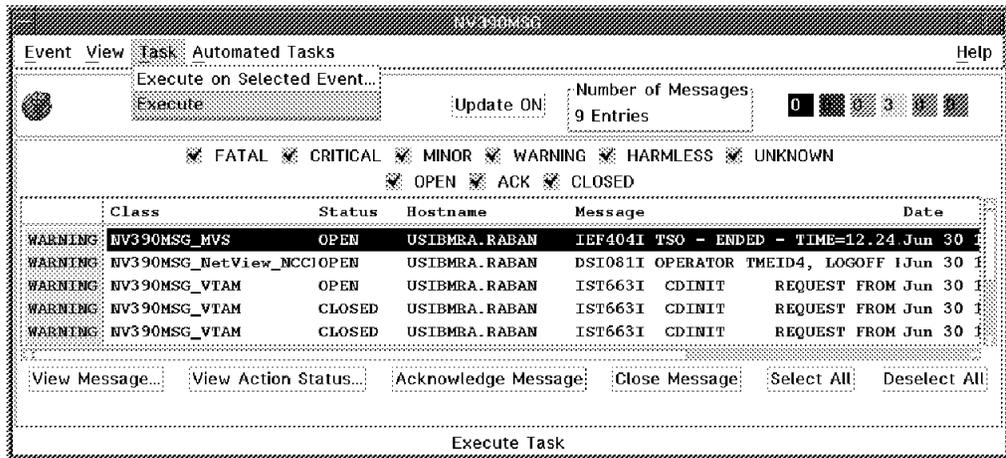
Figure 101. NV390MSG Console

The T/EC Tasks window appears. There, we select the library where the task is defined. The tasks list is upgraded with all the tasks defined in that library. We select the task we want to run, then select the options **Execute on Selected Managed Node** and **Execute and Close**.
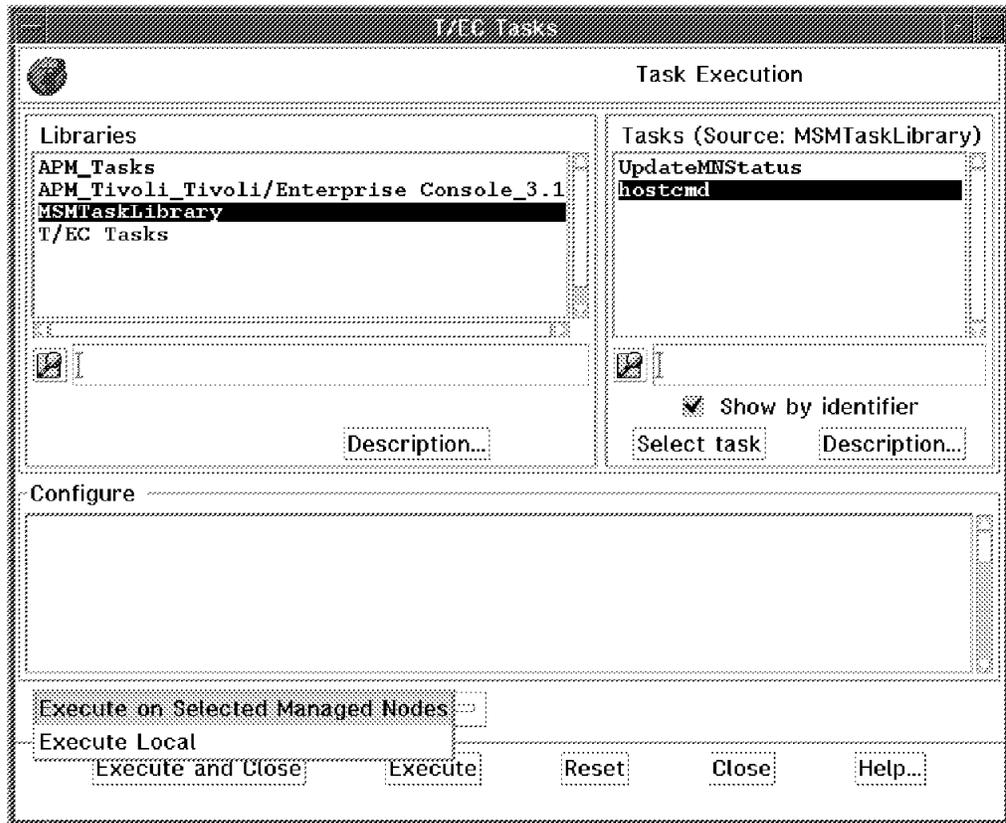
Figure 102. T/EC Tasks Window

A new window appears prompting you for the managed nodes where the task is to execute.
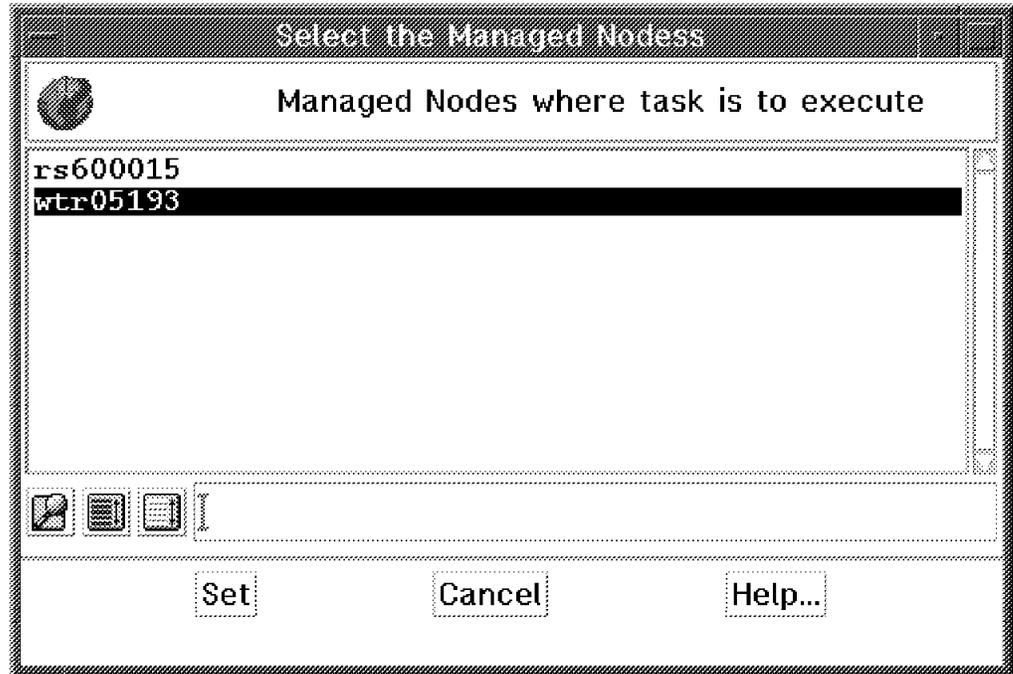
Figure 103. Select Managed Nodes Window

We select the managed node where the APM server runs and then we click on the **Set** button.

The task will then run on the Application Policy Manager server and send the commands to TME 10 NetView for OS/390. They will be executed and the response will be sent back to the machine where T/EC is running.

The result of the execution of the task appears in a window as shown in Figure 104 on page 100.

```
###############################################################################
Task Name:      hostcmd
Task Endpoint:  wtr05193 (ManagedNode)
Return Code:    0
------Standard Output------
Tivoli environment variables configured.┌
BINDIR is set to D:\Tivoli\bin\w32-ix86┌
TP ACM registration for 00005CA3┌
DISPLAY NET,ID=RABAT,SCOPE=ALL ┌
IST097I  DISPLAY  ACCEPTED ┌
IST075I  NAME = USIBMRA,RABAT     , TYPE = APPL ┌
IST486I  STATUS= ACTIV     , DESIRED STATE= ACTIV ┌
IST1447I  REGISTRATION TYPE = CDSERVR ┌
IST977I  MDLTAB=***NA*** ASLTAB=***NA*** ┌
IST861I  MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA*** ┌
IST934I  DLOGMOD=***NA*** USS LANGTAB=***NA*** ┌
IST1632I  VPACING = 10 ┌
IST597I  CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE ┌
IST231I  APPL     MAJOR NODE = RABBTSO ┌
IST212I  ACBNAME = TSO ┌
IST654I  I/O TRACE = OFF, BUFFER TRACE = OFF ┌
IST1500I  STATE TRACE = OFF ┌
IST271I  JOBNAME = TSO      , STEPNAME = TSO     , DSPNAME = IST7FE0B ┌
IST1050I  MAXIMUM COMPRESSION LEVEL - INPUT = 0    , OUTPUT = 0 ┌
IST1633I  ASRCVLM =  2000000 ┌
IST1634I  DATA SPACE USAGE: CURRENT =          0 MAXIMUM  =          0 ┌
IST171I  ACTIVE SESSIONS = 0000000000, SESSION REQUESTS = 0000000000 ┌
IST172I  NO SESSIONS              EXIST ┌
IST314I  END ┌
TP ACM deregistration for 00005CA3┌
TP ACM registration for 00005CA4┌
MVS S TSO ┌
TP ACM deregistration for 00005CA4┌
------Standard Error Output------
###############################################################################
```

Figure 104. Result of the Execution

Chapter 5. NetView/390 Java Client

This chapter describes the TME 10 NetView for OS/390 Java Client, which is the first shipment of Java support for TME 10 NetView for OS/390.

5.1 Overview

The TME 10 NetView for OS/390 Java Client provides access to TME 10 NetView for OS/390 using a TCP/IP connection. Using this client, you can access both the command facility and full-screen applications in TME 10 NetView for OS/390

Note: It does not provide NGMF capabilities.

The TME 10 NetView for OS/390 Java Client code can be obtained by downloading it from the NetView download page:
http://www.tivoli.com/n_nv390/downloads.html

This Web page should be accessed to obtain the latest information about the TME 10 NetView for OS/390 Java Client.

5.2 Software/Hardware Requirements

The TME 10 NetView Java Client has two components: a mainframe component and a workstation environment. The mainframe requires TME 10 NetView for OS/390 Version 1 Release 1. Supported client environments are OS/2, Windows NT, Windows 95 and AIX.

5.2.1 Software Requirements

Additional software specifics are listed below:

- A mainframe environment that includes MVS 5.1.1, MVS 5.2.2 or OS/390 as the operating system. TCP/IP Version 3 Release 2 is required for communication between client and the host. TME 10 NetView for OS/390 Version 1 with APAR OW26026 installed for the specific NetView support for the NetView Java Client. There are two prerequisite APARs for this APAR, OW26168 and the level set APARs OW25087 and OW25680 for English APARs. The Japanese level set APARs are OW25087, OW25629, OW25630, OW25631 and OW25670.

- IBM Operating System/2 (OS/2) Version 3.0 or higher with the Java 1.0.2 Runtime and TCP/IP installed. The NetView Java Client must be installed in an HPFS partition.

  Note: Early levels of the Java 1.0.2 for OS/2 (prior to April 1997) may not operate the Java Client. The symptom is error during logon. To find out what version you have installed use the following command:

  ```
  OS2 C:>java -version
  and this is what you get back
  JAVA.EXE version "JDK 1.0.2 IBM build o102-19970131"
  ```

  For more information carefully read the READ.ME file supplied with the code.

- IBM AIX Version 3.2.5 or higher with Java 1.0.2 Runtime and TCP/IP installed.

- Microsoft Windows 95 with Java 1.0.2 Runtime and TCP/IP installed.

- Microsoft Windows NT Version 4.0 with Java 1.0.2 Runtime and TCP/IP installed.

- Make sure that PATH statements are added to the system so that the Java Runtime libraries can be found. The Java Client installation is in fact a Java application, and both the Java Client and the install procedure require the PATH to be set before running the procedure.

- For all systems, ensure that the current path is specified as either .\; (for OS/2, Windows 95 and NT) or ./: (for AIX) in the PATH definition. Without this definition the Java Client may not be able to be installed or run.

5.2.2 Hardware Requirements

The following are the minimum requirements beyond those required for Intel platform workstations:

- Pentium 90 MHhz or faster CPU

- 32 MB memory

- 15 MB additional fixed disk space for the NetView Java Client code

- Screen resolution of 1024 by 768 with 256 colors

The following are the minimum requirements beyond those required for AIX workstations:

- RISC or Power PC with 133 MHz or faster CPU

- 64 MB memory, 128 MB preferred

- 15 MB additional fixed disk space for the NetView Java Client code

- Screen resolution of 1024 by 768 with 256 colors

5.3 Software Installation

Software installation of the TME 10 NetView for OS/390 Java Client code is not done in the old fashioned way. We explain how to get the code, how to install it and how to customize the code in this section.

5.3.1 Software Installation on MVS

Ensure that you have customized the following DSIPARM members:

- DSIDMNB - DSITCPIP task definition
- DSITCPCF - Task parameters for DSITCPIP
- DSITCPRF - TCP/IP logon encryption profiles

Some sample definitions for this members can be found in A.2.4, "Sample DSIPARM Member Definitions for Java Client" on page 141.

After you have customized the DSIPARM members, NetView client access is enabled when NetView is started.

5.3.2 Software Installation on the Workstation

We explain here how to install and customize the Java Client in an OS/2, Windows NT and an AIX environment.

5.3.2.1 Software Installation for OS/2

To install the TME 10 NetView for OS/390 Java Client in OS/2 perform the following steps in this order:

1. Download the client code from the NetView download page at URL http://www.tivoli.com/n_nv390/downloads.html to a drive and directory you want to install it from, for example D:\NVJAVA. The download is accomplished by using your favorite network browser.

2. Change the directory until you are in the directory that you wish to be the root directory for the client program directory tree. For example, if you are currently in directory D:\NVJAVA, the client program will expand into a directory structure D:\NVJAVA\ibmflb. This can also be a LAN drive, so that many users share a common copy of the program. Individual setting files are created later on local disks for each user.

3. Unpack the files by issuing D:\NVJAVA\FLB4OS2 (or change the directory until you are in the directory NVJAVA and then execute FLB4OS2) which will create the directory ibmflb and subdirectories below it containing the Java Client. You can erase the FLB4OS2.EXE file when you are satisfied with the unpack or the install process. This file is not used by the rest of the install process or the Java Client program.

4. Change directories until you are in the D:\NVJAVA\ibmflb directory. From there you can run the FLB_Install.cmd command which will set up your logon sessions, represented by an icon, in a folder on the desktop.

5. FLB_Install will ask you for several pieces of information that you must enter so that the Java Client can connect to NetView on the S/390 mainframe. There is help to explain what the various pieces of information are but the following is a quick summary:

   - A unique name, such as an operator ID that will appear as part of the program object icon, in the NetView Java Clients folder on the desktop.

- The TCP/IP host name for the S/390 mainframe.

- The port number to be used when communicating with the S/390 mainframe.

- The inbound and outbound encryption keys for the session.

- The directory path for the user settings data. If you specify a path already used by your other sessions, the settings are shared among all sessions. If you want separate settings for some of your sessions, use a separate directory for each case. The install process will create the data directories if they do not already exist. If the Java Client is installed on a local drive, the D:\NVJAVA\ibmflb\dat directory can be used to save the settings.
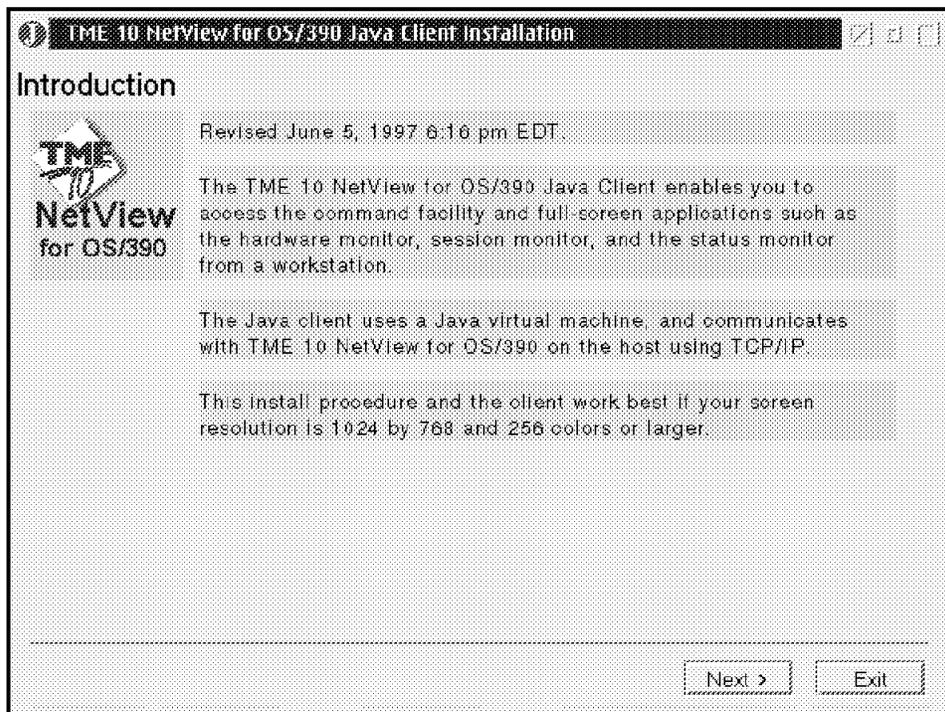
Figure 105. Java Client Introduction Panel

This is the first screen you will see when you have initiated the FLB_Install command. It gives you some information about Java Client. Read the information and click on **Next**.
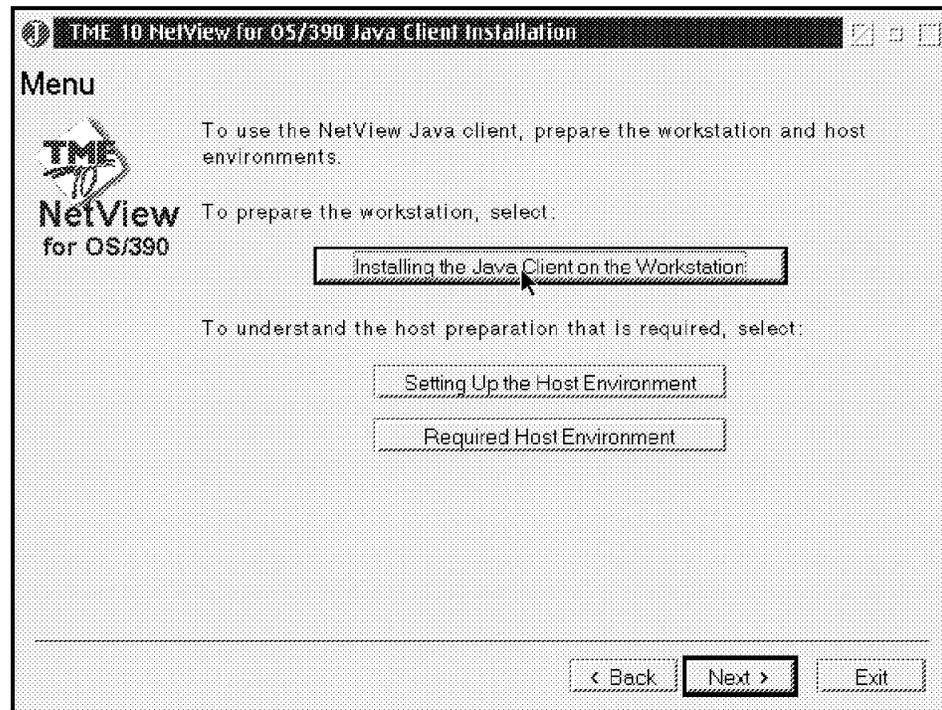
Figure 106. Java Client Menu Panel

This screen gives you information about the required host environment and how to customize the host environment. From this screen you are also allowed to install the Java Client on the workstation.

Select **Installing the Java Client on the Workstation**.
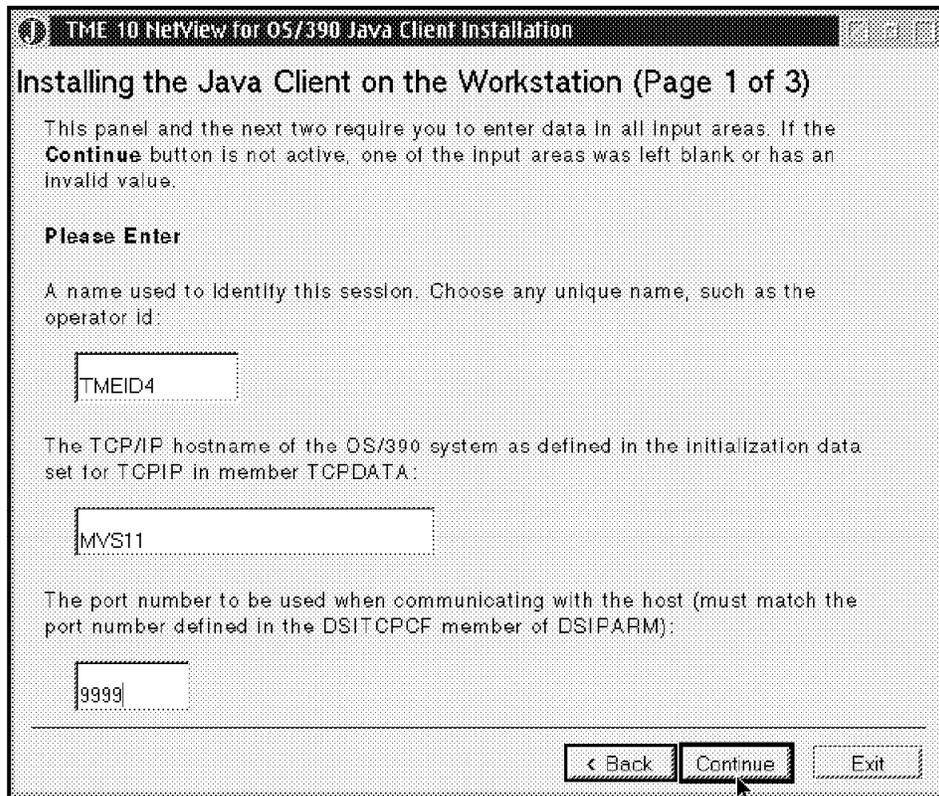
Figure 107. Java Client Install Panel (1 of 3)

The following three screens are the dialog where you have to define the installation options such as operator ID, hostname, port number and encryption options. In our example we used the following specifications:

- Name   = TMEID4
- Hostname = MVS11
- Port   = 9999 (which is the default used in DSITCPCF)

Select **Continue** when you have entered your data.
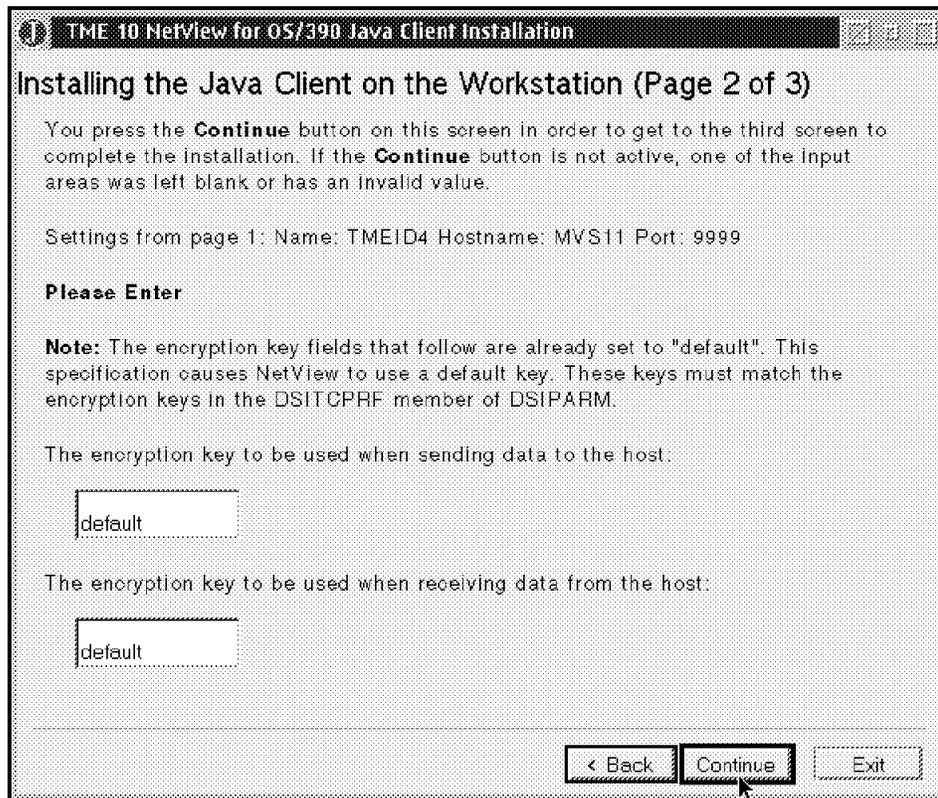
Figure 108. Java Client Install Panel (2 of 3)

This screen is used to define the encryption keys for sending/receiving data between the host and the workstation. If you decide to not use encryption you should type `disabled` in lowercase in both of these fields. In our example we used the default in both directions, where TME 10 NetView for OS/390 uses an internal key.

Select **Continue** when you have entered your data.
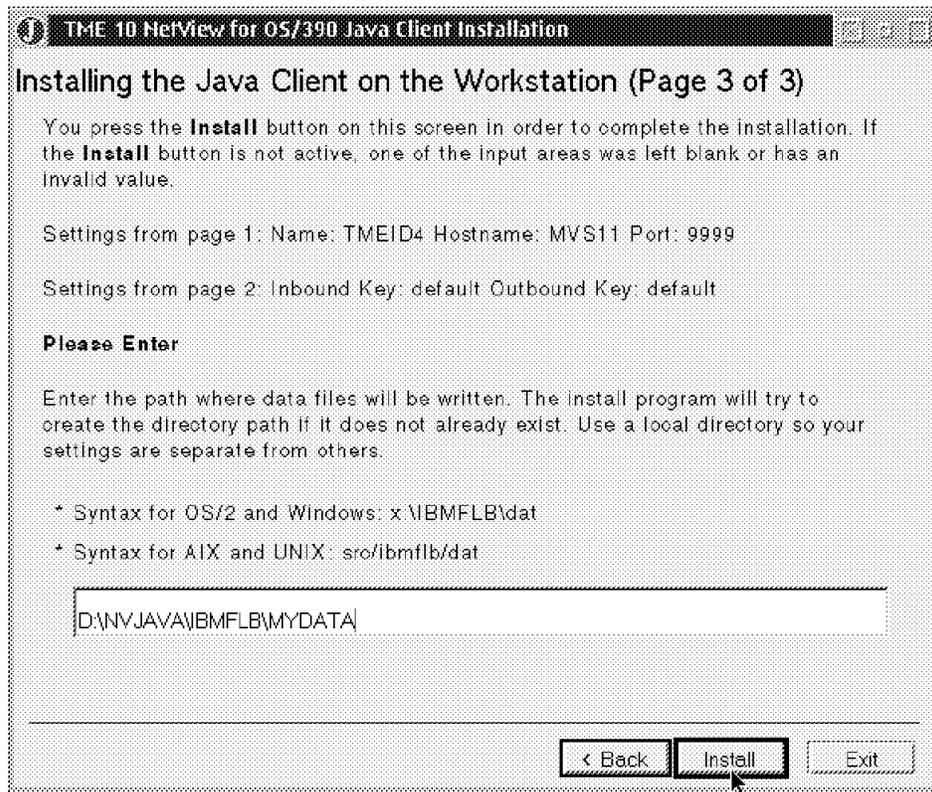
Figure 109. Java Client Install Panel (3 of 3)

Here you define the path where the user settings data is written. The install process will create the data directories if they do not already exist.

Select **Install** when you have entered your data.
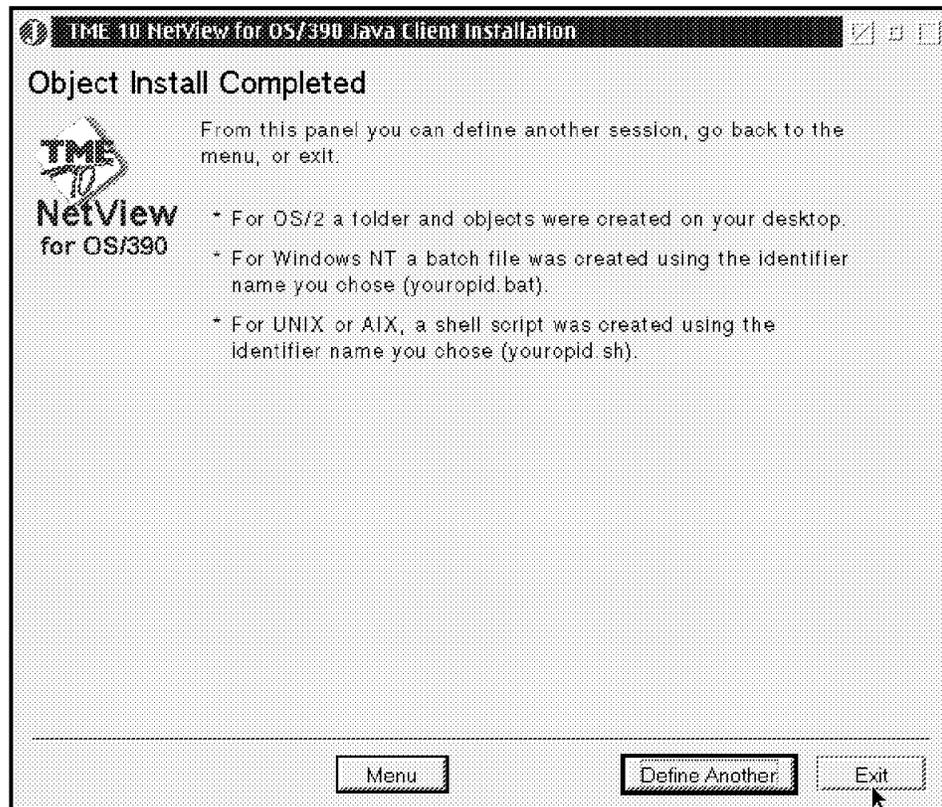
Figure 110. Java Client Install Completed Panel

After you have completed the installation you will see this panel which tells you that a folder is created on the desktop. Now you can define other user settings or exit.

Later we show you how to log on, using the just created icon and user settings.

5.3.2.2 Software Installation for NT

To install the TME 10 NetView for OS/390 Java Client in Windows NT is almost the same as in OS/2, except for the following:

1. Under step 3 in 5.3.2.1, "Software Installation for OS/2" on page 103 use the command FLB4WIN instead of the OS/2 command.

2. Under step 4 in 5.3.2.1, "Software Installation for OS/2" on page 103 use the command FLBINST.bat which will create a batch file for the sessions you want to define (for example, TMEID4.BAT).

The same install panels will appear as for OS/2 and you have to type in the same information on them. The only difference is that the batch file will include the information about hostname, port number and encryption keys.

After completion of the user settings you can choose to start the TME 10 NetView for OS/390 Java Client from the DOS prompt or define program shortcuts for the Java Client.

- To start the Java Client from a DOS prompt:

  1. Change directories to get to the data directory (for example, D:\NVJAVA\IBMFLB\MYDATA).

  2. Type TMEID4 and press Enter.

- To set up program shortcuts for the Java Client:

  1. Create a shortcut of FLBINST.BAT onto the desktop using drag and drop from a view of its containing folder (for example, D:\NVJAVA\IBMFLB\FLBINST.BAT).

  2. Create a shortcut of TMEID4.BAT onto the desktop using drag and drop from a view of its containing folder (for example, D:\NVJAVA\IBMFLB\MYDATA\TMEID4.BAT).

  3. To start the Java Client or install program, click on the shortcut.

5.3.2.3 Software Installation for AIX

To install the TME 10 NetView for OS/390 Java Client in AIX is almost the same as in OS/2, except for the following:

1. Under step 3 in 5.3.2.1, "Software Installation for OS/2" on page 103 use the command compress -d FLB4AIX.TAR.Z.

2. Under step 4 in 5.3.2.1, "Software Installation for OS/2" on page 103 use the command tar -xvf FLB4AIX.TAR which will create the directories that the Java Client runs in.

3. Under step 4 in 5.3.2.1, "Software Installation for OS/2" on page 103 use the command sh flbinst.sh which will create a shell script for the session you want to define (for example, TMEID4.sh).

The same install panels will appear as for OS/2 and you have to type in the same information on them.

After completion of the user settings you can choose to start the NetView Java Client from the AIX prompt or define program shortcuts for the Java Client.

- To start the Java Client from an AIX prompt:

  1. Change directories to get to the data directory (for example, cd /NVJAVA/IBMFLB/MYDATA).

  2. Type sh TMEID4.sh and press Enter.

- To set up program shortcuts for the Java Client:

  1. Change directories to the directory for the user settings data.

  2. Change the permissions settings for the file TMEID4.sh to make it an executable shell script.

  3. Drag a copy of the shell script from its folder in the file manager to the desktop, or onto the Create Icon box in the user applications tray (in the common desktop environment).

  4. To start the Java Client or install program, click on the shortcut.

5.3.3 How To Work with NetView Java Client

Here we give you same examples of how to use the NetView Java Client application:

- How to log on to TME 10 NetView for OS/390.

- How to use a Hardware Monitor full-screen session.

- How to create your own full-screen session.

5.3.3.1 Log On to TME 10 NetView for OS/390

Click on the desktop icon created when running the FLB_Install command to start the Java Client.
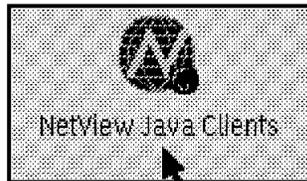
Figure 111. NetView Java Clients Desktop Icon

This will open the TME 10 NetView for OS/390 Java Client folder which contains two icons: TMEID4, which was created during the install process and the Add More Session Objects, which can be used to define or change an object in this folder. The same panels will be shown as during the install process.
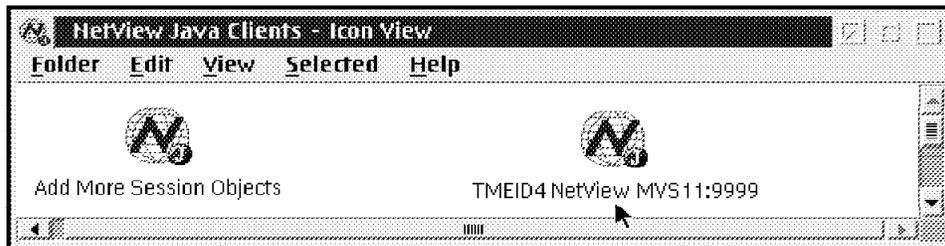
Figure 112. NetView Java Clients - Icon View

To log on to TME 10 NetView for OS/390, click on the icon **TMEID4 NetView MVS 11:9999**.
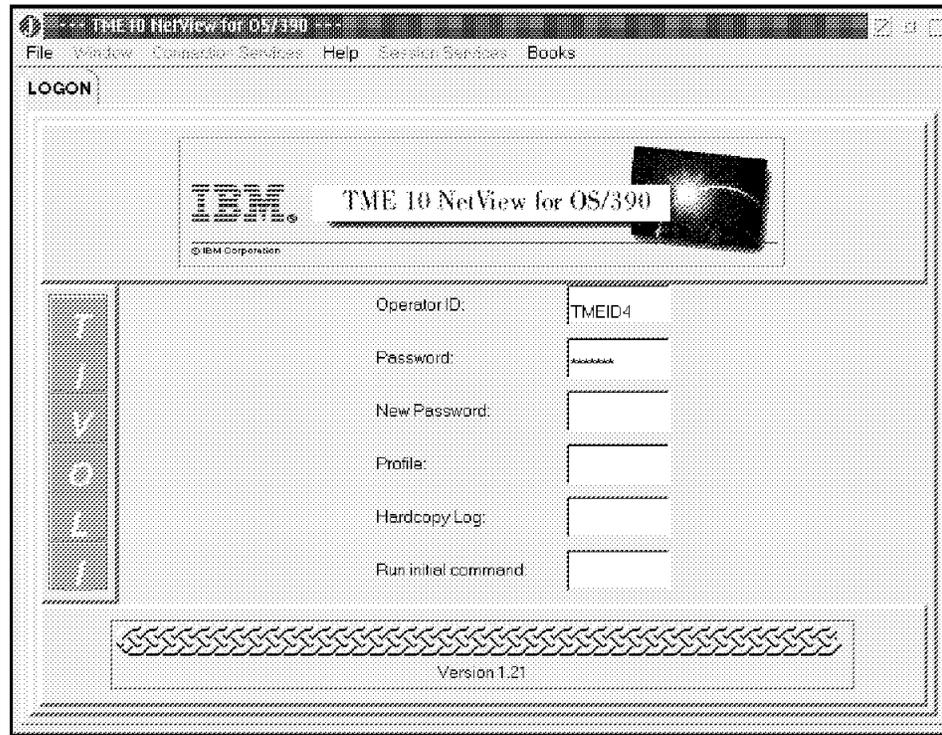
Figure 113. NetView Java Client Logon Screen

The logon panel contains entry fields for logging on to NetView.

After logging on, the Command Facility panel will be displayed.

5.3.3.2 Command Facility Session

On the top of the panel is the menu bar, which can be used to start NetView applications and customize the client workspace.

Figure 114. NetView Java Client, Command Facility Screen

No main menu is displayed after logging on to TME 10 NetView for OS/390. That is because no full-screen panels can be displayed under the Command Facility.

Some restrictions also apply to the PF keys; this is why the message BNH169I is displayed.

The command line is located at the bottom of the panel.

Figure 115. NetView Java Client, PF Keys

You must use the mouse to point and click on the PF keys.

Figure 116. NetView Java Client, Command Line

When issuing commands, first click with the mouse in the command line, then type in the command and press Enter.

The cursor up and cursor down keys can be used to retrieve commands.

Figure 117. NetView Java Client, Command Response 1 of 2

You can see that one of the tasks that should be active is the DSITCPIP task to be able to log on, using the TME 10 NetView for OS/390 Java Client.

The Session Identification Line, which usually is displayed as the top line on a TME 10 NetView for OS/390 screen, can be seen as the bottom line on this screen. MORE indicates that there is more data to be displayed. You can click on the CLR function key or place the cursor in the command line and press Enter to display the additional data. This is instead of the *** you see in the Response Area when you log on to NetView using 3270 communication.

Figure 118. NetView Java Client, Command Response 2 of 2

As you can see the address for the Java Client session is a bit different from what you are used to seeing. In a 3270 session the resource field contains the VTAM name of your terminal.

5.3.3.3 Hardware Monitor Session

Let's look at an example where we want to open a new session to another NetView component.
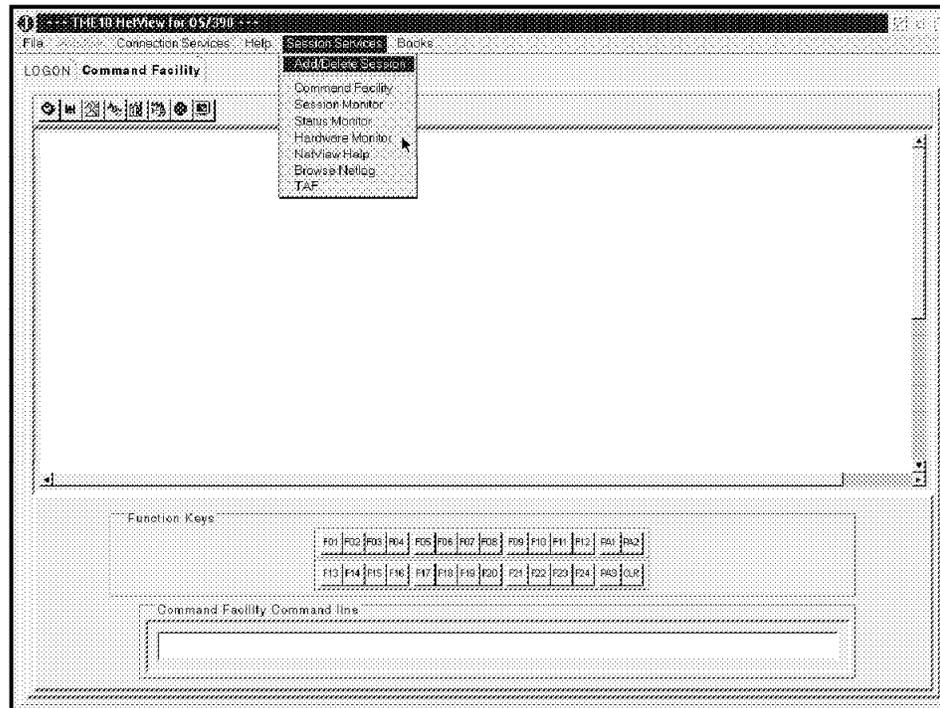
Figure 119. NetView Java Client, Session Services Window

The menu bar selections are:

- File

  Use this selection to close the client workspace.

- Window

  Controls the arrangement of the windowed panels.

- Connection Services

  Manages the connection (log on and log off) with the TME 10 NetView for OS/390 host.

- Help

  Displays help and problem determination information for the TME 10 NetView for OS/390 Java Client application.

- Session Services

  Controls the establishment of new sessions and the addition or deletion of sessions.

- Books

  Displays information on how to access selected TME 10 NetView for OS/390 documentation in HTML format.

To open a full-screen session with Hardware Monitor, open the Session Services window and click on **Hardware Monitor**.
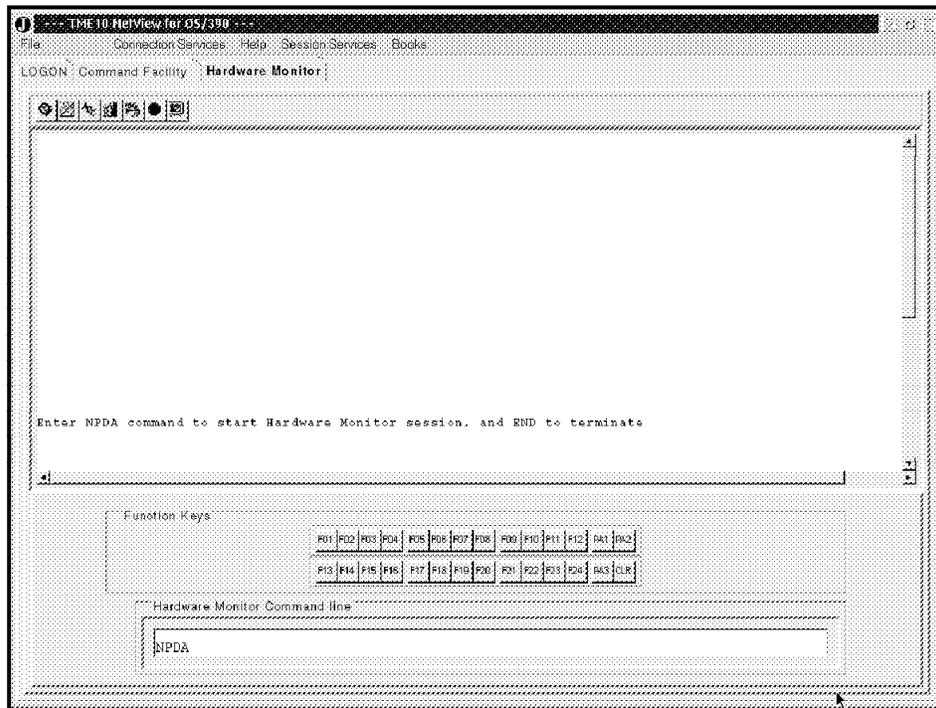
Figure 120. NetView Java Client, Hardware Monitor Session Logon

This will create a new window tab that is highlighted, and some information text on the screen. If the initial command used to start the application is defined as delay, the initial command is placed in the command line. Click with the mouse in the command line and press Enter.

If you don't want to add parameters to the initial command you can define it to execute immediately.
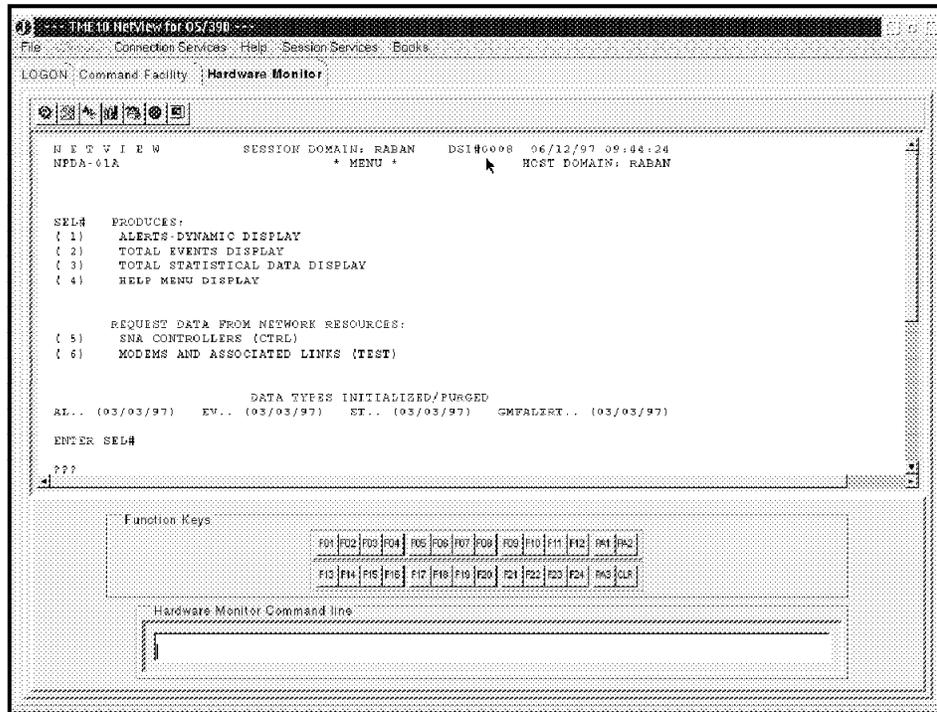
Figure 121. NetView Java Client, Hardware Monitor Menu Panel

This will bring up the Hardware Monitor menu. The only thing that is different from logging on using a 3270 session is the user ID DSI#0008. The Java Client will create a virtual operator station task (VOST), new in TME 10 NetView for OS/390, for this session.

To look at the Alerts-Dynamic screen, click with the mouse in the command line, type 1 and press Enter.
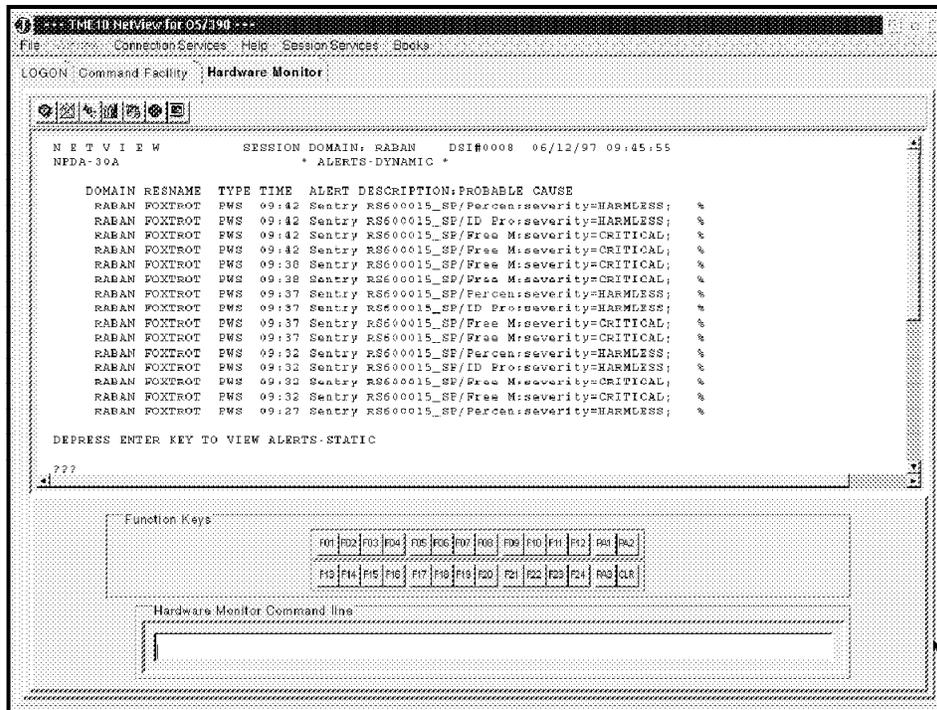
Figure 122. NetView Java Client, Hardware Monitor Alerts-Dynamic Panel

This is the ordinary Alerts-Dynamic screen. From here you can monitor every alert coming in to TME 10 NetView for OS/390, the same way as from a 3270 session.
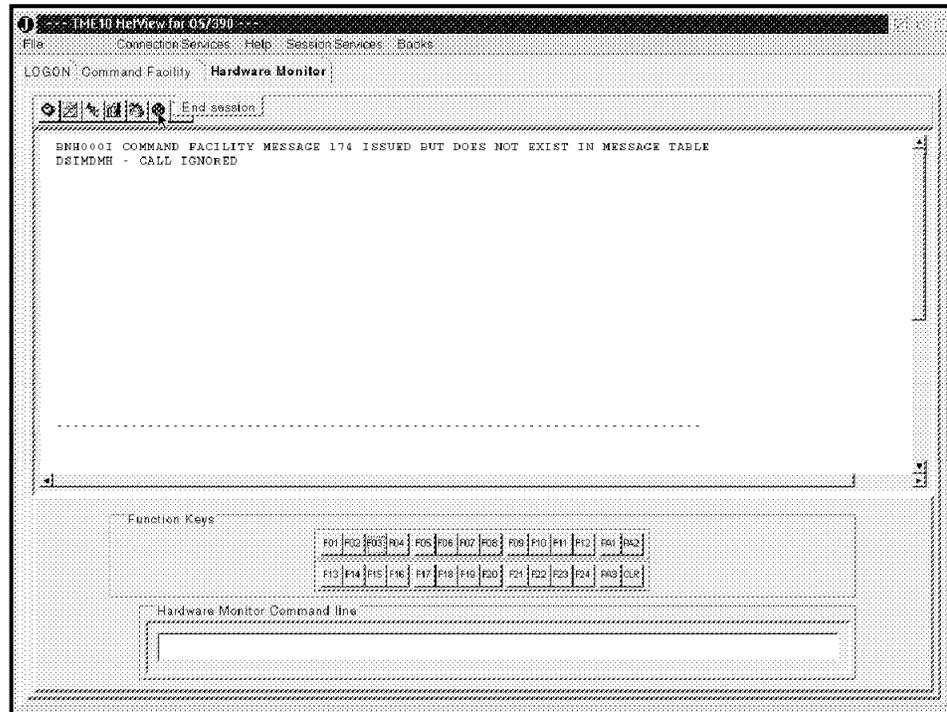
Figure 123. NetView Java Client, End Session Dialog Button

The following are the dialog icons supplied with the Java Client with which you can manipulate the Java Client pages:

- Help

  Gives you information about the dialog icons.

- Local User log

  Using this push button you can either save or clear the contents of the local user log.

- Set Function Keys

  Using this push button allows you to change the default settings.

- Set fonts

  Using this push button you can change the name, style and size of the fonts used for the Command Facility or other full-screen sessions.

- Customize colors

  Using this push button allows you to change color on the text you see on the screen for the different sessions.

- Select wallpaper/background colors

  Using this push button changes the background and the color of the background.

- End session

  Using this push button ends a session. Ending a session will take you back to the command facility.

- Windowed panel

  Using this push button allows you to arrange the active sessions as windows in a suitable way.

To close down the Hardware Monitor session, just click on the tab for **Hardware Monitor** and the **End session** button.

5.3.3.4 Define a New Full-Screen Session

The following pages guide you on how to define and use a new session called NetView Help.

First click on the **Session Services** button and select **Add/Delete sessions**.

Figure 124. NetView Java Client, Add/Delete Sessions Window

In the Full Screen Session Name field, type the name you want to call the new session. In the Start command String field, type the command you want to invoke to initialize the session. Click on the **Immediate** button if you want the command to be executed immediately. Click on the **Delay** button if you want the initial command to be displayed on the command line. After that, select **Add** then **Save** and **Done**.
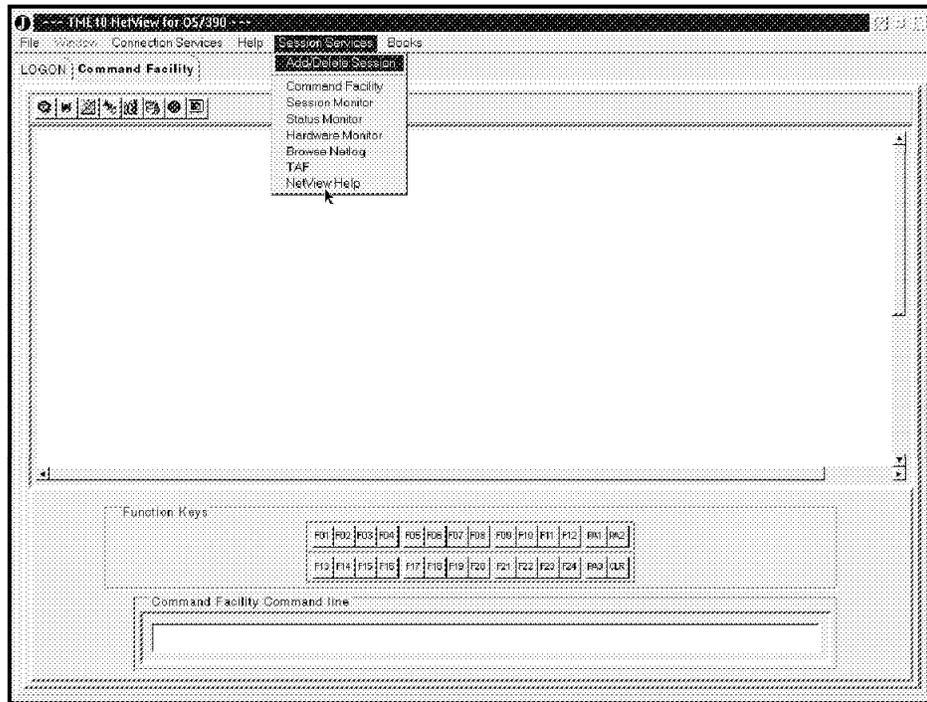
Figure 125. NetView Java Client, Session Services Window

If you now click on **Session Services**, you will see NetView Help as an available full-screen application.

Selecting **NetView Help** will open a new full-screen application. Write the command to start it in the command line, and some information about how to start and stop the application will appear in the Response Area on the screen.
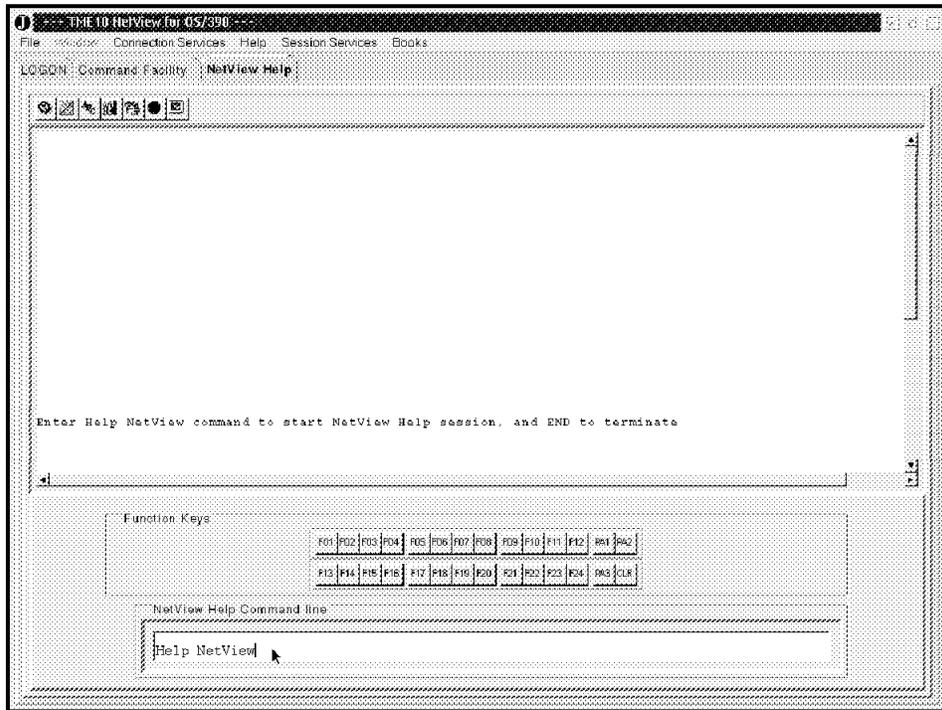
Figure 126. NetView Java Client, Full-Screen Help Session

Click with the mouse in the command line and press Enter.
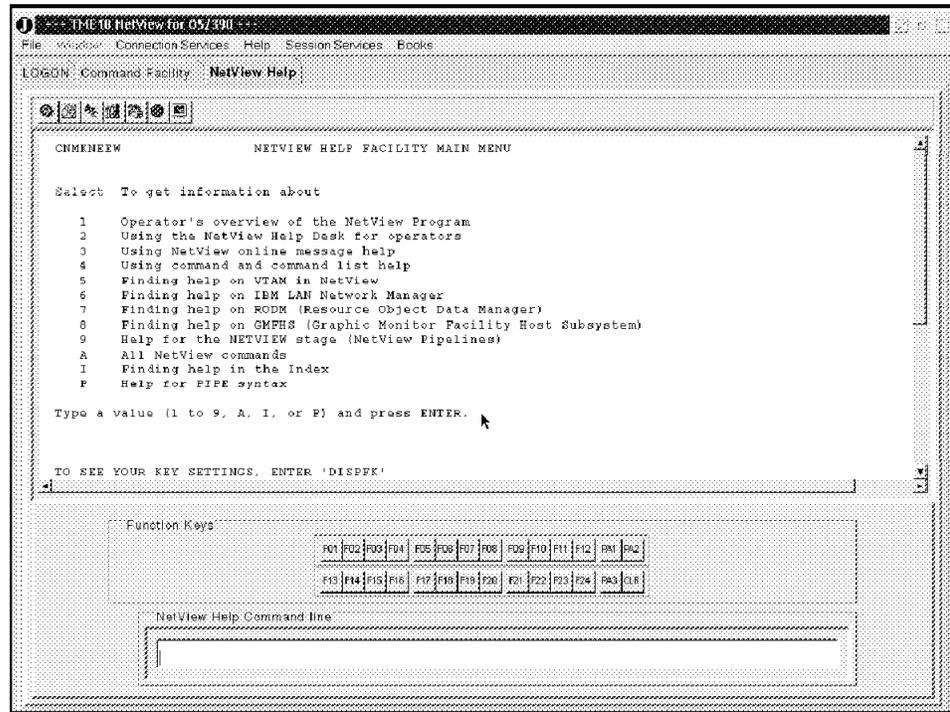
Figure 127. NetView Java Client, NetView Help Main Menu Panel

This will bring up the TME 10 NetView for OS/390 Help Facility main menu, in a full-screen mode, which can not be done for example, in the Command Facility session.

Appendix A. Sample Definitions OS/390 Environment

This appendix shows only the definition or customization samples related to this residency. The samples for the basic software installation and customization can be found in *An Introduction to TME 10 NetView for OS/390*, SG24-4922 and *TME 10 Global Enterprise Manager, Event/Automation and User Administration*, SG24-4921.

A.1 OS/390 JCL

This appendix contains those procedures and members customized on MVS.

A.1.1 Startup Procedures and Initialization Members

The following procedures and initialization members have been customized to run in our environment.

A.1.1.1 NetView Startup Procedure

This is the NetView procedure we used during the residency.

```
//CNMAPROC PROC Q1='TME10',        ** USER DSN HIGH LEVEL QUALIFIER
//       DOMAIN=RABAN,             ** NETVIEW DOMAIN NAME
//       DOMAINPW='RABAN',           ** NETVIEW DOMAIN PASSWORD
//       PROG=BNJLINTX,            ** PGM USED TO START NETVIEW
//       SQ1='TME10.V1R1',  ** SYSTEM DSN HIGH LEVEL QUALIFIER
//       VQ1='TME10',         ** HIGH LVL DSN QUALIFIER-VSAM DSNS
//       SOUTA='*',              ** DEFAULT PRINTED OUTPUT CLASS
//       REG=0,                  ** REGION SIZE(IN K) FOR NETVIEW
//       BFSZ=24,                ** BUFFER SIZE(IN K)
//       SLSZ=200,               **
//       ARM='*NOARM',           ** NETVIEW AUTOMATIC RESTART (ARM) USAGE
//       SUBSYM='*SUBSYM'        ** NETVIEW SYMBOLIC SUBSTITUTION SWITCH
//NETVIEW  EXEC PGM=&PROG,TIME=1440,
//       REGION=&REG.K,
//       PARM=(&BFSZ.K,&SLSZ,'&DOMAIN','&DOMAINPW','&ARM','&SUBSYM'),
//       DPRTY=(13,13)
//STEPLIB  DD   DSN=TME10.RABAN.USERLINK,DISP=SHR
//         DD   DSN=CEE.TME10.SCEERUN,DISP=SHR
//         DD   DSN=&SQ1..CNMLINK,DISP=SHR
//         DD   DSN=&SQ1..SEKGMOD1,DISP=SHR
//         DD   DSN=&SQ1..SFLCLINK,DISP=SHR
//         DD   DSN=REXX.V1R3M0.SEAGLMD,DISP=SHR
//         DD   DSN=&SQ1..SEZLLINK,DISP=SHR
//         DD   DSN=CEE.SCEERUN,DISP=SHR
//         DD   DSN=TCPIP.V3R2M0.SEZALINK,DISP=SHR
//         DD   DSN=TCPIP.V3R2M0.SEZALNK2,DISP=SHR
//DSICLD   DD   DSN=&Q1..&DOMAIN..CNMCLST,DISP=SHR
//         DD   DSN=&SQ1..CNMCLST,DISP=SHR
//         DD   DSN=&SQ1..SFLCCLST,DISP=SHR
//         DD   DSN=&SQ1..SFLCREX1,DISP=SHR
//         DD   DSN=&SQ1..SEZLCLST,DISP=SHR
//         DD   DSN=&SQ1..CNMSAMP,DISP=SHR
//         DD   DSN=&SQ1..SEKGSMP1,DISP=SHR
//DSIOPEN  DD   DSN=&Q1..&DOMAIN..SDSIOPEN,DISP=SHR
//         DD   DSN=&SQ1..SDSIOPEN,DISP=SHR
```

Figure 128. TME 10 NetView for OS/390 Procedure 1/2

```
//DSIPARM  DD   DSN=&Q1..&DOMAIN..DSIPARM,DISP=SHR
//         DD   DSN=&SQ1..SFLCSAMP,DISP=SHR
//         DD   DSN=&SQ1..SFLCSAM1,DISP=SHR
//         DD   DSN=&SQ1..DSIPARM,DISP=SHR
//DSILIST  DD   DSN=&Q1..&DOMAIN..DSILIST,DISP=SHR
//DSIVTAM  DD   DSN=&Q1..&DOMAIN..VTAMLST,DISP=SHR
//DSIPRF   DD   DSN=&Q1..&DOMAIN..DSIPRF,DISP=SHR
//         DD   DSN=&SQ1..SFLCSAMP,DISP=SHR
//         DD   DSN=&SQ1..DSIPRF,DISP=SHR
//DSIMSG   DD   DSN=&SQ1..SDSIMSG1,DISP=SHR
//         DD   DSN=&SQ1..SFLCMSGU,DISP=SHR
//         DD   DSN=&SQ1..SEZLMSGU,DISP=SHR
//BNJPNL1  DD   DSN=&SQ1..BNJPNL1,DISP=SHR
//BNJPNL2  DD   DSN=&SQ1..BNJPNL2,DISP=SHR
//CNMPNL1  DD   DSN=&Q1..&DOMAIN..CNMPNL1,DISP=SHR
//         DD   DSN=&SQ1..CNMPNL1,DISP=SHR
//         DD   DSN=&SQ1..SEKGPNL1,DISP=SHR
//         DD   DSN=&SQ1..SFLCPNLU,DISP=SHR
//         DD   DSN=&SQ1..SEZLPNLU,DISP=SHR
//DSILOGP  DD   DSN=&VQ1..&DOMAIN..DSILOGP,
//         DISP=SHR,AMP='AMORG,BUFNI=20,BUFND=20'
//DSILOGS  DD   DSN=&VQ1..&DOMAIN..DSILOGS,
//         DISP=SHR,AMP='AMORG,BUFNI=20,BUFND=20'
//DSITRCP  DD   DSN=&VQ1..&DOMAIN..DSITRCP,
//         DISP=SHR,AMP=AMORG
//DSITRCS  DD   DSN=&VQ1..&DOMAIN..DSITRCS,
//         DISP=SHR,AMP=AMORG
//AAUVSPL  DD   DSN=&VQ1..&DOMAIN..AAUVSPL,
//         DISP=SHR,AMP='AMORG'
//AAUVSSL  DD   DSN=&VQ1..&DOMAIN..AAUVSSL,
//         DISP=SHR,AMP='AMORG'
//BNJLGPR  DD   DSN=&VQ1..&DOMAIN..BNJLGPR,
//         DISP=SHR,AMP='AMORG'
//BNJLGSE  DD   DSN=&VQ1..&DOMAIN..BNJLGSE,
//         DISP=SHR,AMP='AMORG'
//BNJ36PR  DD   DSN=&VQ1..&DOMAIN..BNJ36PR,
//         DISP=SHR,AMP=AMORG
//BNJ36SE  DD   DSN=&VQ1..&DOMAIN..BNJ36SE,
//         DISP=SHR,AMP=AMORG
//DSIKPNL  DD   DSN=&VQ1..&DOMAIN..DSIKPNL,
//         DISP=SHR,AMP=AMORG
//DSISVRT  DD   DSN=&VQ1..&DOMAIN..DSISVRT,
//         DISP=SHR,AMP=AMORG
//EZLLOGP  DD   DSN=&VQ1..&DOMAIN..LOGP,
//         DISP=SHR,AMP='AMORG,BUFNI=10,BUFND=5'
//EZLLOGS  DD   DSN=&VQ1..&DOMAIN..LOGS,
//         DISP=SHR,AMP='AMORG,BUFNI=10,BUFND=5'
//EZLSTAT  DD   DSN=&VQ1..&DOMAIN..STATS,
//         DISP=SHR,AMP='AMORG'
//SYSPRINT DD SYSOUT=&SOUTA
//SYSTCPD  DD DISP=SHR,DSN=SYS1.TCPPARMS(TCPDATA)
```

Figure 129. TME 10 NetView for OS/390 Procedure 2/2

A.1.1.2 GMFHS Startup Procedure
This is our GMFHS procedure.

```
//GMFHS   PROC Q1='TME10', ** USER DSN HIGH LEVEL QUALIFIER
//              SQ1='TME10.V1R1', ** SYSTEM DSN HIGH LEVEL QUALIFIER
//              VQ1='TME10',   ** HIGH LVL DSN QUALIFIER-VSAM DSNS
//              DOMAIN=RABAN, ** NETVIEW DOMAIN NAME
//              PROG=DUIFT000, ** PGM USED TO START GMFHS HOST MAIN TASK
//              REG=32M,       ** REGION SIZE IN K FOR MAIN TASK
//              AGGRST=,       ** RUN AGG CALCULATION ON STARTUP
//              RESWS=         ** RESOURCE STATUS WARM START
//STEP1    EXEC PGM=&PROG,REGION=&REG,
//              PARM='&AGGRST,RESWS=&RESWS,DOMAIN=&DOMAIN'
//STEPLIB  DD DSN=&SQ1..CNMLINK,DISP=SHR
//         DD DSN=&SQ1..SEKGMOD1,DISP=SHR
//         DD DSN=CEE.SCEERUN,DISP=SHR
//CNMPARM  DD DSN=&Q1..&DOMAIN..DSIPARM,DISP=SHR
//         DD DSN=&SQ1..DSIPARM,DISP=SHR
//CNMMSG1  DD DSN=&SQ1..SDUIMSG1,DISP=SHR
//CNMDB    DD DSN=&VQ1..&DOMAIN..DUIDB,DISP=SHR
//CNMM     DD SYSOUT=A
//CNMD     DD SYSOUT=A
//CNMI     DD SYSOUT=A
//CNMO     DD SYSOUT=A
//CNMF     DD SYSOUT=A
//CNME     DD SYSOUT=A
//CNMV     DD SYSOUT=A
//CNMC     DD SYSOUT=A
//CNMS     DD SYSOUT=A
//CNMT     DD SYSOUT=A
//CNMP     DD SYSOUT=A
//CNMR     DD SYSOUT=A
//SYSTERM  DD SYSOUT=A
//EKGLANG  DD DSN=&SQ1..SEKGLANG,DISP=SHR
//EKGLUTB  DD DSN=&SQ1..SEKGLUTB,DISP=SHR
//EKGPRINT DD SYSOUT=A
//EKGIN3   DD DUMMY,DCB=BLKSIZE=80
```

Figure 130. GMFHS Procedure

A.1.1.3 The RODM Startup Procedure

This is our RODM procedure used during the residency.

```
//RODM     PROC SQ1='TME10.V1R1',  ** SYSTEM DSN HIGH LEVEL QUALIFIER
//         VQ1='TME10.RABAN',      ** HIGH LVL DSN QUALIFIER-VSAM DSNS
//         TYPE=W,                 ** SELECT A COLD OR WARM START
//         NAME=RODM1,
//         INIT=,
//         CLRSSB=NO,
//         CUST=EKGCUST
//********************************************************************
//START    EXEC PGM=EKGTC000,REGION=0K,TIME=1440,
//           PARM='&TYPE,&NAME,&INIT,&CLRSSB,&CUST'
//STEPLIB  DD DSN=TME10.RABAN.USERLINK,DISP=SHR
//         DD DSN=&SQ1..SEKGMOD1,DISP=SHR
//         DD DSN=&SQ1..SEKGMOD2,DISP=SHR
//         DD DSN=CEE.TME10.SCEERUN,DISP=SHR
//         DD DSN=CEE.SCEERUN,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSDUMP  DD SYSOUT=*
//SYSTERM  DD SYSOUT=*
//EKGLOGP  DD DSN=&VQ1..EKGLOGP,DISP=SHR,
//           AMP=('BUFND=10')
//EKGLOGS  DD DSN=&VQ1..EKGLOGS,DISP=SHR,
//           AMP=('BUFND=10')
//EKGCUST  DD DSN=TME10.RABAN.DSIPARM,DISP=SHR
//         DD DSN=&SQ1..SEKGSMP1,DISP=SHR
//EKGLANG  DD DSN=&SQ1..SEKGLANG,DISP=SHR
//EKGMAST  DD DSN=&VQ1..EKGMAST,DISP=SHR
//EKGTRAN  DD DSN=&VQ1..EKGTRAN,DISP=SHR
//EKGD001  DD DSN=&VQ1..EKGCK001,DISP=SHR
//EKGD002  DD DSN=&VQ1..EKGCK002,DISP=SHR
```

Figure 131. RODM Procedure

A.1.1.4 Event/Automation Service Startup Procedure

This is the procedure we used for the Event/Automation Service of the Global Enterprise manager.

```
//IHSAEVNT PROC PROG=IHSAC000,    ** EVENT/AUTOMATION SERVICE
//           REG=32M              ** REGION SIZE IN K FOR MAIN TASK
//           LE370='CEE.TME10',
//  TCPIPDAT='SYS1.TCPPARMS(TCPDATA)', **SET TCPIP.DATA EXPLICIT ALLOC
//           ALRTCFG=,            ** ALERT ADAPTER CONFIG FILE
//           MSGCFG=,             ** MESSAGE ADAPTER CONFIG FILE
//           PPI=,                ** PPI RECEIVER ID
//           INITFILE=            ** Initial File for GEM Services
//           ERCVCFG=,            ** EVENT RECEIVER CONFIG FILE
//           PPI=,                ** PPI RECEIVER ID
//           INITFILE=            ** EVENT AUTOMATION SVC INIT FILE
//************************************************************************
//*
//*
//STEP1    EXEC PGM=&PROG,TIME=1440,REGION=&REG,
//           PARM=('MSGCFG=&MSGCFG ALRTCFG=&ALRTCFG ERCVCFG=&ERCVCFG
//           PPI=&PPI INITFILE=&INITFILE')
//*
//STEPLIB  DD DSN=TME10.V1R1.SIHSMOD2,DISP=SHR
//         DD DSN=&LE370..SCEERUN,DISP=SHR
//         DD DSN=CEE.SCEERUN,DISP=SHR
//*
//*   INITIALIZATION PARAMETERS AND CONFIGURATION FILES DATASET
//IHSSMP3  DD DSN=TME10.RABAN.SIHSSMP3,DISP=SHR
//         DD DSN=TME10.V1R1.SIHSSMP3,DISP=SHR
//*
//*   EAS MESSAGES DATASET
//IHSMSG1  DD DSN=TME10.V1R1.SIHSMSG1,DISP=SHR
//*
//*   EAS OUTPUT DATASETS
//IHSC     DD SYSOUT=A
//IHSM     DD SYSOUT=A
//IHSA     DD SYSOUT=A
//IHSE     DD SYSOUT=A
//*
//*   TCP/IP EXPLICIT ALLOCATION OF TCPIP.DATA
//SYSTCPD  DD DISP=SHR,DSN=&TCPIPDAT
//*TCPDEBUG DD SYSOUT=A
//*
//*   RUNTIME LIBRARY MESSAGES
//SYSTERM  DD SYSOUT=A
```

Figure 132. Event/Automation Service Procedure

A.1.1.5 Event Automation Initialization File

This is the default initialization member for the Event/Automation service of Global Enterprise Manager.

```
#
# This is the initialization file for the GEM Event/Automation Service.
#
# Note: All keywords and values must be in uppercase.
#$01=OW25491,V1R1,03/10/97,RLF: TME10 GEM EVENT/AUTOMATION SERVICE V1R1
#                                   LEVELSET.
#$02=OW27082,V1R1,05/20/97,RLF: NEW FUNCTION.
#
# The following keywords are supported:
#   PPI     - Specifies the PPI receiver ID used by the Event/Automation
#               Service
#   ALRTCFG - Specifies the alert adapter configuration file
#   MSGCFG  - Specifies the message adapter configuration file
#   ERCVCFG - Specifies the event receiver configuration file
#   OUTPUT  - Specifies the destination of trace messages
#   NOSTART - Specifies which tasks are not to be started
#   TRACE   - Specifies trace logging settings
#
# The syntax for each keyword follows:
#   PPI=receiver_id
#   ALRTCFG=alert_config_file
#   MSGCFG=message_config_file
#   ERCVCFG=event_receiver_config_file
#   NOSTART - Specifies which tasks are not to be started
#   TRACE   - Specifies trace logging settings
#
# The syntax for each keyword follows:
#   PPI=receiver_id
#   ALRTCFG=alert_config_file
#   MSGCFG=message_config_file
#   ERCVCFG=event_receiver_config_file
#   OUTPUT=output_destination
#   NOSTART TASK=task_designator
#   TRACE TASK=task_designator LEVEL=trace_level EIF=eif_trace_level
#      (EIF keyword is only valid for tasks ALERTA & MESSAGEA)
#
#   where:
#     receiver_id - Specifies the input PPI receiver ID used by both the
#                     message adapter and alert adapter.  The default
#                     value is IHSATEC.
#
#     alert_config_file - Specifies the name of the alert adapter
#                           configuration file.  The file will be read
#                           at initialization time from the IHSSMP3
#                           dataset.  The default value is IHSAACFG.
#
#     message_config_file - Specifies the name of the message adapter
#                             configuration file.  The file will be read
#                             at initialization time from the IHSSMP3
#     output_destination - specifies the location where the trace
#                             messages will be printed, the values are:
#       SYSOUT - Specifies the system output log, default
#       GTF    - Specifies the GTF trace log/dataset
#                             If both are specified the syntax is:
#                               (SYSOUT,GTF)
```

Figure 133. IHSAINIT Event/Automation Initialization File 1/2

```
#
#     task_designator - is one of the following:
#        CONTROL  - Specifies the control task
#        ALERTA   - Specifies the alert adapter task
#        MESSAGEA - Specifies the message adapter task
#        EVENTRCV - Specifies the event receiver task
#
#        Note : CONTROL is not valid on the NOSTART statement
#
#     trace_level - is one of the following:
#        OFF     - Specifies no tracing, default
#        LOW     - Specifies a minimal tracing
#        NORMAL  - Specifies a normal tracing
#        VERBOSE - Specifies a verbose tracing
#
#     eif_trace_level - is one of the following:
#        OFF     - Specifies no tracing, default
#        LOW     - Specifies a minimal tracing
#        NORMAL  - Specifies a normal tracing
#        VERBOSE - Specifies a verbose tracing
#
#
# PPI Receiver ID
PPI=IHSATEC
#
# Alert Adapter Configuration File
ALRTCFG=IHSAACFG
#
# Message Adapter Configuration File
MSGCFG=IHSAMCFG
#
# Event Receiver Configuration File
ERCVCFG=IHSAECFG
#
# Trace Message Output Destination
OUTPUT=SYSOUT
#
# Tasks not started at initialization (example)
#NOSTART TASK=ALERTA
#NOSTART TASK=MESSAGEA
#
# Control Task
TRACE TASK=CONTROL LEVEL=OFF
#
# Alert Adapter Task
TRACE TASK=ALERTA LEVEL=OFF EIF=OFF
#
# Message Adapter Task
TRACE TASK=MESSAGEA LEVEL=OFF EIF=OFF
#
# Event Receiver Task
TRACE TASK=EVENTRCV LEVEL=OFF
```

Figure 134. IHSAINIT Event/Automation Initialization File 2/2

A.1.1.6 RODM Loader Job
We used the following job to load RODM.

```
//GEORGESY JOB O-111111,MSGCLASS=O,MSGLEVEL=(1,1),REGION=4M,CLASS=I       00001002
//LOADDM   EXEC EKGLOADP,
//  RODMNAME=RODM1,         name of GMFHS's RODM (change or del. parm)
//  EKGIN1=NULLFILE,        EKGIN1 overridden - see below
//  EKGIN3=NULLFILE,        not needed for structure load
//  LOAD=STRUCTURE,
//  OPER=LOAD,
//  LISTL=ERRORSYNTAX,
//  SEVERITY=WARNING
//*
//LOADRODM.EKGIN1 DD DSN=TME10.V1R1.CNMSAMP(DUIFSTRC),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM1),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM2),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM3),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM4),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM5),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM6),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM7),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM8),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDM9),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMA),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMB),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMC),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMD),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDME),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMF),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMZ),DISP=SHR
//*
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMH),DISP=SHR
//         DD DSN=TME10.V1R1.CNMSAMP(FLBTRDMI),DISP=SHR
//*   MSM data model
//*
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM1),DISP=SHR       <-CLASS/FIELD DEF
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM2),DISP=SHR       <-CLASS/FIELD DEF
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM3),DISP=SHR       <-CLASS/FIELD DEF
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM4),DISP=SHR       <-CLASS/FIELD DEF
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM5),DISP=SHR       <-CLASS/FIELD DEF
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6),DISP=SHR        <-MSM    EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6A),DISP=SHR <-ATM       EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6H),DISP=SHR <-NETFIN EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6I),DISP=SHR <-IP        EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6L),DISP=SHR <-LNM       EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6M),DISP=SHR <-LMU       EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6N),DISP=SHR <-NETWAR EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6O),DISP=SHR <-OPEN      EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM6T),DISP=SHR <-TME       EXC VIEW
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM7),DISP=SHR        <-IPM SAMPLE
//  DD DSN=TME10.V1R1.SFLCSAMP(FLCSDM8),DISP=SHR        <-CORRELATE SAMPL
```

Figure 135. RODM Loader Job

A.1.1.7 Assemble/Linkedit DUIFSMT

This the job we used to assemble and linkedit the status mapping table. These changes are then loaded into RODM.

```
//GEORGESY JOB O-111111,MSGCLASS=O,MSGLEVEL=(1,1),REGION=4M,CLASS=I     00001002
//* ASM STEP - ASSEMBLE THE STATUS MAPPING TABLE                    *
//ASM      EXEC PGM=ASMA90,
//            PARM='NODECK,OBJECT,XREF(SHORT)',REGION+1024K
//SYSLIB   DD  DSN=TME10.V1R1.MACLIB,DISP=SHR
//         DD  DSN=SYS1.MACLIB,DISP=SHR
//SYSUT1   DD  DSN=&&SYSUT1,UNIT=SYSDA,SPACE=(1700,(600,100))
//SYSPRINT DD  SYSOUT=*,DCB=BLKSIZE=1089
//SYSLIN   DD  DSN=&&OBJSET,UNIT=SYSDA,SPACE=(3120,(24,12)),
//            DISP=(MOD,PASS),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSIN    DD  DSN=GEORGES.JOB.JCL(DUIFSMT),DISP=SHR
//LKED     EXEC PGM=HEWL,REGION=4096K,
// COND=(O,LT,ASM),
// PARM='XREF,LET,LIST,RENT,AMODE=31,RMODE=ANY,MAP,COMPAT=LKED'
//SYSUT1   DD  DSN=&&SYSUT1,UNIT=SYSDA,SPACE=(1024,(50,20))
//SYSLIB   DD  DSN=CEE.SCEERUN,DISP=SHR
//SYSPRINT DD  SYSOUT=*,DCB=(RECFM=FB,LRECL=121,BLKSIZE=1210)
//SYSIN2   DD  DSN=TME10.RABAN.USERLINK,DISP=SHR
//         DD  DSN=TME10.V1R1.SEKGMOD2,DISP=SHR
//SYSLMOD  DD  DSN=TME10.RABAN.USERLINK,DISP=SHR
//SYSLIN   DD  DSN=*.ASM.SYSLIN,DISP=(OLD,DELETE)
//         DD  DDNAME=SYSIN
//SYSIN    DD  *
 ENTRY   DUIFCRDC
 INCLUDE SYSIN2(DUIFCRDC)
 NAME    DUIFCRDC(R)
/*
//INSTLOAD  EXEC EKGLOADP,COND=(O,NE),
//          RODMNAME=RODM1,         NAME OF GMFHS'S RODM
//          EKGIN1=NULLFILE,        EKGIN1 NOT USED
//          EKGIN3=NULLFILE,        EKGIN3 OVERRIDDEN BELOW
//          LOAD=INSTANCE,
//          OPER=LOAD,
//          LISTL=ALLSYNTAX,
//          SEVERITY=WARNING
//LOADRODM.EKGIN3  DD *
--
--     Refresh the DisplayStatus change method
--
OP EKG_Method.DUIFCRDC.EKG_Refresh INVOKED_WITH;
--
--     Invoke the DisplayStatus change method on the
--     GMFHS_Monitorable_Objects_Parent_Class with a
--     value of 0x55555555 to update the Status Mapping
--     table.
--
OP GMFHS_Monitorable_Objects_Parent_Class..DisplayStatus
   HAS_VALUE (INTEGER) 1431655765;
--
--     Invoke the method DUIFRFDS in order to trigger
--     DUIFCRDC in order to get the Exception View
--     updates immediately. Otherwise the updates will
--     occur with the next DisplayStatus change.
--
OP DUIFRFDS INVOKED_WITH;
/*
```

Figure 136. Job to Assemble and Link Edit DUIFSMT

A.2 Customization Members

Some of the members we customized for the examples in this redbook are included here. The additional customization of TME 10 NetView for OS/390 is described in 3.2.1, "MultiSystem Manager Customization" on page 18.

A.2.1 FLCSITME Used for the TME Feature of MultiSystem Manager

This is the complete FLCSITME initialization member that we used for our examples.

```
***************************************************************************
* The command name GETTOPO is required.  TMERES or TMEONLY keywords      *
* are valid when used in the initialization file.  TMERES and            *
* TMEONLY are positional parameters and must be specified as the         *
* first keyword on the GETTOPO statement.                                *
***************************************************************************
GETTOPO TMERES,
***************************************************************************
* SP is a required keyword.  Specify the TCP/IP hostname of the          *
* machine running the MSM TME agent.                                     *
***************************************************************************
SP=RS600015,
POLICY=YES,
***************************************************************************
* MONITORS is optional.  Valid values are YES or NO.  If MONITORS is     *
* NOT specified, MONITORS = YES is the default.                          *
***************************************************************************
MONITORS=YES,
***************************************************************************
* PORT is optional.  Valid values are port names or port numbers for     *
* the MSM TME agent.  If PORT is NOT specified, PORT = 3333 is the        *
* default.                                                               *
***************************************************************************
PORT=3333,
***************************************************************************
* NETWORK_AG_OBJECT allows you to provide a specific name for the        *
* network aggregate view.  This keyword is optional.  If not             *
* specified, a network aggregate view named TME_Networks will be         *
* created.  If NETWORK_AG_OBJECT=NONE is specified, the network          *
* aggregate object view will NOT be created.                            *
***************************************************************************
NETWORK_AG_OBJECT=TME10-first-test,
***************************************************************************
* NETWORK_NAME allows you to provide a specific name for the network     *
* aggregate object.  This keyword is optional.  If not specified, a      *
* default name will be assigned.                                        *
***************************************************************************
NETWORK_NAME=RALEIGH_TMR_TEST,
***************************************************************************
* NETWORK_VIEW allows you to provide a specific name for the network     *
* view and associated annotation.  If not specified, the default         *
* network view and annotation will be used.                             *
***************************************************************************
NETWORK_VIEW=TME10_VIEW/TEST1
```

Figure 137. MultiSystem Manager Initialization Member FLCSITME

A.2.2 FLCSDM6T Loaded into RODM for Exception View Processing

This is our FLCSDM6T member that we used to create the exception view for our TME monitors.

```
--**********************************************************************
--*      5697-B82 (C) Copyright IBM Corp. 1997.                        *
--*      All Rights Reserved.                                          *
--**********************************************************************
--*                                                                    *
--*  Sample Name: FLCSDM6T                                             *
--*                                                                    *
--*  Description: Exception_View_Class objects                         *
--*                                                                    *
--*   This data model sample loads instances of Exception_View_Class   *
--*   for the TME Topology Feature                                     *
--*                                                                    *
--*  Change Activity:                                                  *
--*                                                                    *
--**********************************************************************
--
-- Create Exception View for TME10 Monitors (Monitor)
--    Sample type on NGMF Resource Information panel is 'Monitor'
--    NOTE: The MultiSystem Manager NetFinity Feature also uses
--          the 'Monitor' class. The Exception_View_Class with a
--          MyName of 'NetFinity_Monitor' is defined for qualifying
--          NetFinity Monitor objects in sample file FLCSDM6H.

CREATE INVOKER  ::= 0000001;
      OBJCLASS ::= Exception_View_Class;
      OBJINST  ::= MyName = (CHARVAR) 'TME10_Monitors';
      ATTRLIST
      Annotation ::= (CHARVAR) 'TME10 Sentry Monitors',
      ExceptionViewName ::= (CHARVAR) 'TME10MON';
END;
--**********************************************************************
-- NOTE:    Real objects of the NetFinity, LMU and TME Features of
--          MSM are contained in objects of the common Open Topology
--          class '1.3.18.0.0.6464' (aggregateSystem).  In addition,
--          Real objects of all other MultiSystem Features may be
--          contained in aggregateSystem objects if the customer
--          initializes the dynamic topology correlation function. This
--          function can create or update aggregateSystem objects based
--          upon correlation of LAN MAC Addresses and TCP/IP Addresses.
--            A sample exception view for aggregateSystem objects is
--          defined as 'OPAGGSYS' in RODM Loader file 'FLCSDM6'.
--          The sample Exception View 'OPAGGSYS', if created, will include
--          workstations and IP Systems created by ALL Features.
--            Sample types on NGMF Resource Information panel for
--          aggregateSystem objects are 'LAN Workstation' and 'IP System'.
--
```

Figure 138. FLCSDM6T Exception View Object in RODM

A.2.3 DUIFSMT Including the FLCSSMT

This is the status mapping table including our customized FLCSSMT member.

```
DUIFSMT   TITLE 'STATUS MAPPING TABLE'
DUIFSMT   CSECT
          DUIFSMTE CLASS=APPNNN,                                       C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=INTERCHANGENODE,                              C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=MIGRATIONDATAHOST,                            C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=T5NODE,                                       C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=APPNTRANSMISSIONGROUP,                        C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=APPNTRANSMISSIONGROUPCIRCUIT,                 C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=T4NODE,                                       C
                XCPT=(UNSAT,UNKWN,DS152,DS153,DS154,DS155,DS156,DS157,DSC
                158,DS159,MEDUN,LOWUN)
          DUIFSMTE CLASS=GMFHS_Managed_Real_Objects_Class,            C
                XCPT=(UNSAT,DS152,DS153,DS154,DS155,DS156,DS157,DS158,DSC
                159,MEDUN,LOWUN)
* All DUIFSMTE statements for specific classes must be added before
* the next statement.
*    Modified FLCSSMT for residency
*--    agent
          DUIFSMTE CLASS=1.3.18.0.0.3315.0.3.1,                        C
                XCPT=(UNSAT)
*--    monitors
          DUIFSMTE CLASS=Monitor,                                      C
                XCPT=(UNSAT)
*--    end of FLCSSMT
          DUIFSMTE CLASS=ALL,                                          C
                XCPT=(UNSAT,DEGRD,SDGRD,DS152,DS153,DS154,DS155,DS156,DSC
                157,DS158,DS159)
LAST      DUIFSMTE END
          END
```

Figure 139. DUIFSMT Status Mapping Table

A.2.4 Sample DSIPARM Member Definitions for Java Client

The following members need to be customized for the TME 10 NetView for OS/390 Java client.

A.2.4.1 DSITCPCF

This is the DSITCPIP initialization member for our TME 10 NetView for OS/390 Java client.

```
* Sample DSITCPIP TCPIP Java Init File
*
* The TCPANAME is the name of the TCP/IP address space
*
        TCPANAME=T11ATCP
*
* The PORT defines which port on TCP32 the NetView will be.  This is
*  defined also to the Java Client to define which NetView to connect
*  to.
*
          PORT=9999
*
* SOCKETS defines how many users can logon to the NetView using TCP/IP.
*        TCP/IP gives you a minimum of 50 sockets, so numbers less than
*        50 are not very useful.
*
        SOCKETS=50
*
* TIMEOUT defines how many seconds elapse before a request times out.
*            This value should not need to be adjusted.
*
        TIMEOUT=60
*
* DIAGNOSE creates additional debugging messages in the log.
*            This value should not need to be adjusted.
*
        DIAGNOSE=NO
*
```

Figure 140. DSITCPCF TCP/IP Configuration Member

A.2.4.2 DSIDMNB

This is the task statement we added to DSIDMNB for the DSITCPIP task.

```
**************************************************************************
     TASK    MOD=DSITCPIP,TSKID=DSITCPIP,MEM=DSITCPCF,PRI=6,INIT=Y
**************************************************************************
```

Figure 141. DSITCPIP Task Added to DSIDMNB

A.2.4.3 DSITCPRF

These are the encryption keys we defined in DSITCPRF for our NetView userids.

```
* These cards define encryption keys for each operator.  The operator
* id is followed by a colon and then any number of blanks.  The first
* non-blank field is the encryption key for the data flowing from the
* Java Client to NetView (command flow).  The second non-blank field is
* the encryption key from NetView to the Java Client.  These keys are
* 8 bytes in length. Shorter values are expanded with a non-published
* sequence of characters. Longer values are truncated. The Java Client
* must be defined to use the same keys.  The keys are never sent on
* any session, and the DSITCPRF member should be put into a secure
* (DSIPRF DD) library.  If both keys are the word "disabled", no
* encryption is used.  This is best used for debugging session problems
* and is suitable only in low-risk networks.  This member only defines
* the encryption keys.  NetView operator id's must be defined by
* the existing methods.  As a security benefit, only operators defined
* in this file can logon to NetView via a Java Client. In other words
* unknown ids are treated as intruders. If both keys are the word
* "default", NetView uses internal keys.  Each operator using "default"
* will have the same keys.
*
* You use the keyword "ANY_OTHER:" to specify the encryption keys for
* all operator ids that have specific definitions in DSITCPRF.  This
* makes defining everyone using a single encryption key easier.
* If you leave out the "ANY_OTHER:" keyword, only the operators you
* define in DSITCPRF can logon to NetView using TCP/IP.
* This sample has "ANY_OTHER:" as a comment to limit the authorization
* until you decide to change it.  Notice this keyword is 9 characters in
* length to avoid conflicts with real operator names.
*
* Example of user specified keys
*OPER1: XXXXXXXX YYYYYYYY
*
* Example of default keys
*OPER2: default default
*
* IF OPERX is not defined to NetView, it should get DSIO29I msg
*OPERX: AAAAAAAA BBBBBBBB
*
* Encryption keys can be mixed case, and generally should be chosen
* to be "random printable non-blank characters".
*
*OPER4: a1s2d3fg lonibuvy
*
TMEID4: default default
*TMEID5: disabled disabled
*
* The ANY_OTHER: is a special keyword that defines a key to be used
*    by any operator that is not listed by name in this DSITCPRF.
*    Your can remove the "*" from this statement and use other encryption
*    keys or "disabled" for all of these operators to use.
*
ANY_OTHER: default default
```

Figure 142. DSITCPRF Initialization Member

Appendix B. Configuration and Customization Files

This appendix contains the configuration files, baroc files and rule files that we used during the residency. In addition the Global Enterprise Manager provided shell script for installation and our modified shell script are included here.

B.1 Configuration Files

The following are the configuration files that need to be changed for the Topology and Command Services.

B.1.1 MSM Configuration File - MSMAgent.cfg

This is our customized MSMAgent configuration file. Note the parameter INITIALSTATUS=HARMLESS as discussed earlier.

```
# ****************************************************************
#     5697-B83 (C) Copyright IBM Corp. 1997.
#       All Rights Reserved.
#
# Name:          MSMAgent.cfg
#
#  Description: This file contains configuration parameters for the
#               TME MSMAgent. Each parameter is set to a default
#               which may be modified by the system administrator
#               by editing this file directly.
#
# ****************************************************************

LOGSIZE=10              # maximum size of MSMAgent.log in megabytes
                       # if the MSMAgent.log exceeds this size it will be
                       # overwritten.

CMDSERVERPORT=3333     # the port number on which cmdServer.pl will listen
                       # for connection requests from MultiSystemsManager

DBLOCKRETRIES=60       # The number of times the MSM TMEAgent will retry
                       # accessing the local topology database when it is in
                       # a locked state before assuming deadlock and forcing
                       # the database to an unlocked state.
                       # The TMEAgent retries the locked local database
                       # every 10 seconds, so the default value of 60
                       # is equivalent to waiting 10 minutes before forcing
                       # an unlock. 10 minutes is more than adequate for the
                       # TMEAgent in a medium size network to gather topology
                       # of several hundred manages nodes with several connected
                       # TMRs. However, it is possible that for very large
                       # networks the TME Agent may validly require more than
                       # the default of 60 retries. The TME administrator
                       # can estimate the elapsed time it takes to gather the
                       # topology by inspecting the start and stop times in the
                       # MSMAgent.log file and adjusting the DBLOCRETRIES
                       # parameter accordingly, if necessary.
                       #
```

Figure 143. MSMAgent Configuration File 1/2

```
SYSERRMSGSTOHOST=YES      # This parameter specifies whether or not to send
                          # system or TME generated error messages written to STDERR
                          # NetView. Set this parameter to NO to keep system or
                          # TME messages encountered at the ServicePoint from
                          # being forwarded to NetView. STDERR messages will be
                          # written to the MSMAgent.log at the ServicePoint in
                          # either case.

INITIALSTATUS=HARMLESS    # Sets the initial Status of Monitor objects in NGMF.
                          # You may want this to be HARMLESS so the object
                          # initialize to GREEN looking active, rather then Gray.

PROFILENAME.0=ALL         # To set up Events for all your Sentry Profile Monitors
                          # run the SetMonitors.pl command at customization time.
                          # SetMonitors.pl will look for the PROFILENAME.0 parm.
                          #
                          # When PROFILENAME.0=ALL        ** DEFAULT **
                          #    - Uses the command "wlookup -r SentryProfile -a"
                          #        to get ALL Sentry Profile's and adds Events to
                          #        any monitor that DOESN'T have one.  Doing ALL
                          #        can take a while if you have a lot of profiles.
                          #
                          # When PROFILENAME.0=AnyProfileName
                          #    - Uses the command "wlsmon AnyProfileName" to get
                          #        a list of monitors for that profile and adds
                          #        Events to any monitor that DOESN'T have one.
                          #
                          # To Process a list of Profiles, the format is:
                          #    - PROFILENAME.0=MissysProfile
                          #    - PROFILENAME.1=BillsProfile
                          #    - PROFILENAME.2=JohnsProfile
                          #
                          # When PROFILENAME.0 is NULL or NOT found
                          #    - No profiles are updated.
                          #
                          # Severity Mappings represent the Monitors Severity
                          # used for TE/C Events Severity.
                          #
CRITICAL_SEVERITY = CRITICAL
SEVERE_SEVERITY   = CRITICAL
WARNING_SEVERITY  = WARNING
NORMAL_SEVERITY   = HARMLESS
ALWAYS_SEVERITY   = UNKNOWN
```

Figure 144. MSMAgent Configuration File 2/2

B.1.2 Command Service Configuration File - ihsshstc.cfg

This is our configuration file for the Command Service. We used TMEID2 as the NetView user ID.

```
*******************************************************************************
*
* Licensed Materials - Property of IBM
* 5697-B83
* (C) Copyright IBM Corp. 1997.  All rights reserved.
*
* US Government Users Restricted Rights - Use, duplication or
* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*
*******************************************************************************
*
* PRODUCT          (TME 10 GEM APM)
*
* MODULE_NAME      (IHSSHSTC.CFG)
* DESCRIPTIVE_NAME (Configuration file for Command Exit
*                   that sends comands to NV390)
*******************************************************************************
*
* Function:
*   Defines the NetView 390 Operator id and password that will execute the
*   NetView 390 command string passed as input arguments to command exit
*   IHSSHSTC.
*
*   Keywords supported are:
*
*     "OPER_ID="  Defines the NetView 390 Operator id
*     "OPER_PW="  Defines the NetView 390 Operator password (optional)
*
*   Notes:
*
*     - The password is optional, when the NetView 390 Operator id does
*       not require a password.  In this case, either the keyword can be
*       removed from this file, or by specifing nothing after the "=".
*     - There must NOT be a space between the keyword and the "=".
*     - There optionally can be spaces between the "=" and the value.
*     - Comment lines are supported, which must have an '*' in column 1.
*     - Blank lines are ignored.
*     - Lines with unrecongized keywords are ignored.
*     - When the same keyword is repeated on multiple lines, the last one is
*       the one used.  The others are ignored.
*
*******************************************************************************

OPER_ID=TMEID2
OPER_PW=TMEID2
```

Figure 145. Command Service Configuration File ihsshstc.cfg

B.1.3 Communications Server for Windows NT

This configuration file was used to configure an LU 6.2 session to TME 10 NetView from the NT machine using Communications Server for Windows NT Version 5.0. If you are familiar with CM/2 NDF files used for NGMF, you will see that this is very similar.

```
NODE=(
     ANYNET_SUPPORT=ANYNET_SUPPORTED
     CP_ALIAS=RA6015CP
     DEFAULT_PREFERENCE=NATIVE
     DISCOVERY_GROUP_NAME=<NONE>
     DISCOVERY_SUPPORT=NO
     FQ_CP_NAME=USIBMRA.RA6015CP
     NODE_ID=05D00000
     NODE_TYPE=END_NODE
     REGISTER_WITH_CDS=1
     REGISTER_WITH_NN=0
)
PORT=(
     PORT_NAME=LAN0_04
     DLC_DATA=00000000000004
     DLC_NAME=LAN
     IMPLICIT_CP_CP_SESS_SUPPORT=1
     IMPLICIT_DEACT_TIMER=0
     IMPLICIT_DSPU_SERVICES=NONE
     IMPLICIT_HPR_SUPPORT=1
     IMPLICIT_LIMITED_RESOURCE=NO
     LINK_STATION_ROLE=NEGOTIABLE
     MAX_IFRM_RCVD=8
     MAX_RCV_BTU_SIZE=65535
     PORT_TYPE=SATF
     PORT_LAN_SPECIFIC_DATA=(
          ACK_DELAY=100
          ACK_TIMEOUT=10000
          ADAPTER_NUMBER=0
          BUSY_STATE_TIMEOUT=15
          IDLE_STATE_TIMEOUT=30
          LOCAL_SAP=04
          OUTSTANDING_TRANSMITS=16
          POLL_TIMEOUT=8000
          POOL_SIZE=32
          REJECT_RESPONSE_TIMEOUT=10
          TEST_RETRY_INTERVAL=8
          TEST_RETRY_LIMIT=5
          XID_RETRY_INTERVAL=8
          XID_RETRY_LIMIT=5
     )
)
```

Figure 146. Communication Server Configuration File 1/5

```
LINK_STATION=(
     LS_NAME=LINK0000
     ACTIVATE_AT_STARTUP=1
     ADJACENT_NODE_TYPE=APPN_NODE
     AUTO_ACTIVATE_SUPPORT=0
     CP_CP_SESS_SUPPORT=0
     DEFAULT_NN_SERVER=0
     DEST_ADDRESS=40005200500004
     DISABLE_REMOTE_ACT=0
     DSPU_SERVICES=NONE
     HPR_SUPPORT=0
     LIMITED_RESOURCE=NO
     LINK_DEACT_TIMER=0
     LINK_STATION_ROLE=USE_ADAPTER_DEFAULTS
     MAX_IFRM_RCVD=0
     MAX_SEND_BTU_SIZE=65535
     NODE_ID=05D00000
     PORT_NAME=LAN0_04
     PU_NAME=APM
     SOLICIT_SSCP_SESSION=1
     TARGET_PACING_COUNT=1
     TG_NUMBER=0
     USE_DEFAULT_TG_CHARS=1
)
DLUR_DEFAULTS=(
     DEFAULT_PU_NAME=RA6015CP
     DLUS_RETRY_LIMIT=3
     DLUS_RETRY_TIMEOUT=5
)
LU_0_TO_3=(
     LU_NAME=LU2002
     LU_MODEL=3270_DISPLAY_MODEL_2
     NAU_ADDRESS=2
     POOL_NAME=PUBLIC
     PRIORITY=HIGH
     PU_NAME=APM
)
LU_0_TO_3=(
     LU_NAME=LU2003
     LU_MODEL=3270_DISPLAY_MODEL_2
     NAU_ADDRESS=3
     POOL_NAME=PUBLIC
     PRIORITY=HIGH
     PU_NAME=APM
)
LOCAL_LU=(
LU_NAME=RA600150
LU_ALIAS=EGVPC
     LU_SESSION_LIMIT=0
     NAU_ADDRESS=0
     ROUTE_TO_CLIENT=0
     SYNCPT_SUPPORT=0
)
```

Figure 147. Communication Server Configuration File 2/5

```
MODE=(
     MODE_NAME=BLANK
     AUTO_ACT=0
     COS_NAME=#CONNECT
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=1
     MAX_NEGOTIABLE_SESSION_LIMIT=8192
     MAX_RU_SIZE_UPPER_BOUND=1024
     MIN_CONWINNERS_SOURCE=4096
     PLU_MODE_SESSION_LIMIT=8192
     RECEIVE_PACING_WINDOW=3
)
MODE=(
     MODE_NAME=#BATCH
     AUTO_ACT=0
     COS_NAME=#BATCH
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=0
     MAX_NEGOTIABLE_SESSION_LIMIT=8192
     MAX_RU_SIZE_UPPER_BOUND=2048
     MIN_CONWINNERS_SOURCE=4096
     PLU_MODE_SESSION_LIMIT=8192
     RECEIVE_PACING_WINDOW=20
)
MODE=(
     MODE_NAME=#BATCHSC
     AUTO_ACT=0
     COS_NAME=#BATCHSC
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=1
     MAX_NEGOTIABLE_SESSION_LIMIT=8
     MAX_RU_SIZE_UPPER_BOUND=2048
     MIN_CONWINNERS_SOURCE=4
     PLU_MODE_SESSION_LIMIT=8
     RECEIVE_PACING_WINDOW=3
)
MODE=(
     MODE_NAME=#INTER
     AUTO_ACT=0
     COS_NAME=#INTER
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=1
     MAX_NEGOTIABLE_SESSION_LIMIT=8192
     MAX_RU_SIZE_UPPER_BOUND=4096
     MIN_CONWINNERS_SOURCE=4096
     PLU_MODE_SESSION_LIMIT=8192
     RECEIVE_PACING_WINDOW=20
)
```

Figure 148. Communication Server Configuration File 3/5

```
MODE=(
     MODE_NAME=#INTERSC
     AUTO_ACT=0
     COS_NAME=#INTERSC
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=1
     MAX_NEGOTIABLE_SESSION_LIMIT=8
     MAX_RU_SIZE_UPPER_BOUND=2048
     MIN_CONWINNERS_SOURCE=4
     PLU_MODE_SESSION_LIMIT=8
     RECEIVE_PACING_WINDOW=7
)
MODE=(
     MODE_NAME=QPCSUPP
     AUTO_ACT=0
     COS_NAME=#CONNECT
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=1
     MAX_NEGOTIABLE_SESSION_LIMIT=52
     MAX_RU_SIZE_UPPER_BOUND=1024
     MIN_CONWINNERS_SOURCE=26
     PLU_MODE_SESSION_LIMIT=52
     RECEIVE_PACING_WINDOW=2
)
MODE=(
     MODE_NAME=QSERVER
     AUTO_ACT=0
     COS_NAME=#CONNECT
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=1
     MAX_NEGOTIABLE_SESSION_LIMIT=64
     MAX_RU_SIZE_UPPER_BOUND=1024
     MIN_CONWINNERS_SOURCE=0
     PLU_MODE_SESSION_LIMIT=64
     RECEIVE_PACING_WINDOW=7
)
MODE=(
     MODE_NAME=SNASVCMG
     AUTO_ACT=0
     COS_NAME=SNASVCMG
     CRYPTOGRAPHY=NONE
     DEFAULT_RU_SIZE=0
     MAX_NEGOTIABLE_SESSION_LIMIT=2
     MAX_RU_SIZE_UPPER_BOUND=512
     MIN_CONWINNERS_SOURCE=1
     PLU_MODE_SESSION_LIMIT=2
     RECEIVE_PACING_WINDOW=1
)
PARTNER_LU=(
FQ_PLU_NAME=USIBMRA.RABAN
     CONV_SECURITY_VERIFICATION=0
     MAX_MC_LL_SEND_SIZE=32767
     PARALLEL_SESSION_SUPPORT=1
     PARTNER_LU_ALIAS=RABAN
     PREFERENCE=USE_DEFAULT_PREFERENCE
)
```

Figure 149. Communication Server Configuration File 4/5

```
TP=(
TP_NAME=30F0F4F4
     API_CLIENT_USE=0
     CONVERSATION_TYPE=EITHER
     DUPLEX_SUPPORT=EITHER_DUPLEX
     DYNAMIC_LOAD=0
     INCOMING_ALLOCATE_TIMEOUT=60
     LOAD_TYPE=0
PATHNAME=d:\Tivoli\bin\w32-ix86\GEM\ApmServer\BIN\ihsctp.exe
     PIP_ALLOWED=1
     QUEUED=1
     RECEIVE_ALLOCATE_TIMEOUT=0
     SECURITY_RQD=0
     SYNC_LEVEL=EITHER
     TP_INSTANCE_LIMIT=0
     TP_NAME_FORMAT=0
)
TP=(
     TP_NAME=APINGD
     API_CLIENT_USE=0
     CONVERSATION_TYPE=EITHER
     DUPLEX_SUPPORT=EITHER_DUPLEX
     DYNAMIC_LOAD=1
     INCOMING_ALLOCATE_TIMEOUT=30
     LOAD_TYPE=0
     PATHNAME=D:\IBMCS\apingd.exe
     PIP_ALLOWED=1
     QUEUED=0
     RECEIVE_ALLOCATE_TIMEOUT=3600
     SECURITY_RQD=0
     SYNC_LEVEL=EITHER
     TP_INSTANCE_LIMIT=0
     TP_NAME_FORMAT=0
)
TN3270E_DEF=(
     AUTO_LOGOFF=0
     FREQUENCY=60
     KEEPALIVE_TYPE=TN_NONE
     LOGOFF=30
     PORT=23
     TIMER=10
)
SPLIT_STACK=(
     POOL_NAME=<None>
     STARTUP=1
)
VERIFY=(
     CFG_MODIFICATION_LEVEL=12
     CFG_VERSION_LEVEL=1
)
```

Figure 150. Communication Server Configuration File 5/5

B.2 Rule Files

Rule files are provided for Global Enterprise Manager Integration Services. The Event/Automation service provides rules for the message adapter, alert adapter and event forwarding to TME 10 NetView for OS/390. The topology service provides a rule for forwarding of TME 10 Distributed Monitoring events.

B.2.1 Message Adapter Rule - tecad_nv390msg.rls

This is the default rule for the message adapter.

```
/*=====================================================================*/
/* Licensed Materials - Property of IBM                               */
/* 5697-B82                                                           */
/* (C) Copyright IBM Corp. 1997.  All rights reserved.                */
/*                                                                    */
/* US Government Users Restricted Rights - Use, duplication or        */
/* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.  */
/*********************************************************************/
/*                                                                    */
/* Description:                                                       */
/*   Default Rules Associated with the NetView/390 Message Adapter.   */
/*                                                                    */
/*=====================================================================*/
/*=====================================================================*/
/* This rules set utilizes the classes from the tecad_nv390msg.baroc  */
/* file.  This should only be used in conjunction with the NetView/390*/
/* Message Adapter.                                                   */
/*=====================================================================*/


/*********************************************************************/
/************************    RULES    ********************************/
/*********************************************************************/


/*=====================================================================*/
/* The nv390msg_dup_event rule determines if the current event is a   */
/* duplicate of an existing event, and if so the current event is     */
/* discarded.  This rule should be first to ensure no other rules are */
/* fired.                                                             */
/*=====================================================================*/

rule: nv390msg_dup_event:
  (
  event: _event of_class 'NV390MSG_Event',
  reception_action:
    (
    first_duplicate(_event,
      event: _nv390msg_dup_ev
        where
          [
          status: outside ['CLOSED']
          ],
        _event - 600 - 600
      ),
    commit_rule,
    add_to_repeat_count(_nv390msg_dup_ev, 1),
    drop_received_event
    )
  ).
```

B.2.2 SNA Alert Adapter Rules - tecad_snaevent.rls

These are the default rules for the alert adapter.

```
/*===================================================================*/
/* Licensed Materials - Property of IBM                             */
/* 5697-B82                                                         */
/* (C) Copyright IBM Corp. 1997.  All rights reserved.              */
/*                                                                  */
/* US Government Users Restricted Rights - Use, duplication or      */
/* disclosure restricted by GSA ADP Schedule Contract with IBM Corp. */
/*******************************************************************/
/*                                                                  */
/* Description:                                                     */
/*   Default Rules Associated with the Systems Network Architecture */
/*   (SNA) Alert Adapters.  This includes the NetView/390 Alert     */
/*   Adapter and AS/400 Alert Adapter.                              */
/*                                                                  */
/*===================================================================*/


/*===================================================================*/
/* This rules set utilizes the classes from the tecad_snaevent.baroc */
/* file.  This should only be used in conjunction with the Systems   */
/* Network Architecture (SNA) Alert Adapters, for example the        */
/* NetView/390 Alert Adapter and the AS/400 Alert Adapter.           */
/*===================================================================*/


/*******************************************************************/
/************************** RULES     ****************************/
/*******************************************************************/


/*===================================================================*/
/* The sna_dup_event rule determines if the current event is a       */
/* duplicate of an existing event, and if so the current event is    */
/* discarded.  This rule should be first to ensure no other rules are */
/* fired.                                                            */
/*===================================================================*/

rule: sna_dup_event:
 (
  event: _event of_class 'SNA_Event',
  reception_action:
    (
    first_duplicate(_event,
      event: _sna_dup_ev
        where
          [
          status: outside ['CLOSED']
          ],
        _event - 600 - 600
      ),
    commit_rule,
    add_to_repeat_count(_sna_dup_ev, 1),
    drop_received_event
    )
  ).


/*===================================================================*/
/* Notice that the "sna_dup_event" rule above detects a duplicate only*/
/* when all of the slots in the SNA_Event with "dup_detect=yes" match */
```

```
/* (see the tecad_snaevent.baroc file).  Some customers may prefer a   */
/* more general duplicate detection rule, for example two events can   */
/* be considered duplicates when they are both received within a       */
/* specified time interval and their SNA Hierarchies (in the origin    */
/* slot) match.  For example, if two events are received within one    */
/* minute of each other and both have an SNA Hierarchy of              */
/* "NCP01/COMC,LINE01/LINE,CTRL01/CTRL" then these events can be        */
/* considered duplicates and the most recent event can be discarded,   */
/* even if the msg slot differs for the two events.  Recall that the   */
/* msg slot contains a long error description-long probable cause      */
/* message.  You may wish to discard the second event even though its  */
/* msg slot differs because most likely both events pertain to the     */
/* same underlying problem since they have the same SNA hierarchy and  */
/* were received within one minute of each other.                      */
/*                                                                     */
/* The following commented-out rule detects duplicates based only on   */
/* matching the SNA Hierarchy within one minute.  Uncomment it and     */
/* delete the "sna_dup_event" rule above if you prefer this looser     */
/* duplicate definition.                                               */
/*                                                                     */
/* rule: sna_dup_event:                                                */
/*   (                                                                 */
/*   event: _event of_class 'SNA_Event'                                */
/*     where                                                           */
/*       [                                                             */
/*       origin: _origin                                               */
/*       ],                                                            */
/*   reception_action:                                                 */
/*     (                                                               */
/*     first_instance(                                                 */
/*       event: _sna_dup_ev of_class 'SNA_Event'                       */
/*         where                                                       */
/*           [                                                         */
/*           status: outside ['CLOSED'],                               */
/*           origin: equals _origin                                    */
/*           ],                                                        */
/*         _event - 60 - 60                                            */
/*       ),                                                            */
/*     commit_rule,                                                    */
/*     add_to_repeat_count(_sna_dup_ev, 1),                            */
/*     drop_received_event                                             */
/*     )                                                               */
/*   ).                                                                */
/*===================================================================*/


/*===================================================================*/
/* The sna_correlated_event rule determines if the current event is   */
/* correlated (read: resulted from the same underlying problem) to an */
/* existing SNA event, and if so the current event is discarded.  This*/
/* prevents two administrators from addressing two Events that         */
/* represent the same problem.  SNA alerts are correlated by MSU       */
/* Correlation subvector 47.  See the IBM "SNA Formats" book for more */
/* information concerning subvector 47.                                */
/*===================================================================*/

rule: sna_correlated_event:
  (
  event: _event of_class 'SNA_Event'
    where
```

```
              [
              event_correl: _event_correl outside [['N/A']]
              ],
        reception_action:
          (
          all_instances(
            event: _correlated_event of class 'SNA_Event'
              where
                [
                status: outside ['CLOSED'],
                event_correl: _corr_event_correl outside [['N/A']]
                ],
            _event - 600 - 600
            ),
          % Find all correlated events.
          intersect(_event_correl, _corr_event_correl),
          % If a correlated event is found, do the following.
          commit_rule,
          add_to_repeat_count(_correlated_event, 1),
          drop_received_event
          )
        ).


        /*====================================================================*/
        /* The following two rules correlate SNA Major Vector 0000 Generic    */
        /* Alerts and Major Vector 0002 Resolutions.  When a Resolution is     */
        /* received, CLOSE all the Generic Alerts which are resolved (i.e.     */
        /* fixed) by the Resolution.  Alerts and Resolutions are correlated by*/
        /* Incident Identification subvector 4A.  See the IBM "SNA Formats"    */
        /* book for more information.                                          */
        /*====================================================================*/

        rule: sna_Mv0002_Resolution:
          (
          event: _event of class 'SNA_Event'
            where
              [
              arch_type: equals 'GENERIC_RESOLUTION',
              incident_correl: _incident_correl outside [['N/A']]
              ],
          action:
            (
            all_instances(
              event: _resolved_event of class 'SNA_Event'
                where
                  [
                  incident_correl: _corr_incid_correl outside [['N/A']],
                  arch_type: equals 'GENERIC_ALERT',
                  status: outside ['CLOSED']
                  ],
              _event - 600 - 600
              ),
            % Find all Mv0000 alerts that are resolved by this Mv0002 resolution
            intersect(_incident_correl, _corr_incid_correl),
            % CLOSE all Mv0000 alerts that are resolved by the Mv0002.
            set_event_status(_resolved_event, 'CLOSED')
            )
          ).
```

```
/*=====================================================================*/
/* Sometimes the resolved Major Vector 0000 Generic Alert may be       */
/* received after the Major Vector 0002 Resolution.  For this case,    */
/* send the Mv0002 Resolution through the rules again, and this will   */
/* result in the Mv0000 Generic Alert being resolved.                  */
/*=====================================================================*/

rule: sna_Mv0000_Generic_Alert:
  (
  event: _event of_class 'SNA_Event'
    where
      [
      arch_type: equals 'GENERIC_ALERT',
      incident_correl: _incident_correl outside [['N/A']],
      status: outside ['CLOSED']
      ],
  reception_action:
    (
    all_instances(
      event: _resolution_ev of_class 'SNA_Event'
        where
          [
          arch_type: equals 'GENERIC_RESOLUTION',
          incident_correl: _corr_incid_correl outside [['N/A']]
          ],
      _event - 600 - 600
      ),
    % Find all Mv0002 resolutions that resolve the Mv0000 alert.
    intersect(_incident_correl, _corr_incid_correl),
    % Redo the Mv0002 resolutions, this causes the Mv0000 alerts to
    % be CLOSED.
    redo_analysis(_resolution_ev)
    )
  ).
```

B.2.3 Forwarding of Sentry Events to NetView/390 - forwardSentry.rls

This is the default forwardSentry rule as shipped with Global Enterprise
Manager.

```
rule: fwd_to_nv390:
  (
  event: _event of_class 'Sentry3_5_Base',
  reception_action:
    (
    forward_event(_event)
    )
  ).
change_rule: fwd_to_nv390_2:
  (
  event: _event of_class 'Sentry3_5_Base',
  slot: status set_to _status,
  action:
    (
    forward_event(_event)
    )
  ).
```

B.2.4 Forwarding of Events to NetView/390 - tecad_nv390fwd.rls

This is the default tecad_nv390fwd rule as shipped with Global Enterprise
Manager

```
/*======================================================================*/
/* Licensed Materials - Property of IBM                                 */
/* 5697-B82                                                             */
/* (C) Copyright IBM Corp. 1997.  All rights reserved.                  */
/*                                                                      */
/* US Government Users Restricted Rights - Use, duplication or          */
/* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.    */
/************************************************************************/
/*                                                                      */
/* Description:                                                         */
/*   Sample rules for forwarding events to NetView/390                  */
/*                                                                      */
/*======================================================================*/


/*======================================================================*/
/* To forward events you must:                                          */
/* 1) Edit the tec_forward.conf file and:                               */
/*   a) fill in ServerLocation                                          */
/*   b) comment out or delete the line: TestMode=yes                    */
/* 2) Update your rules base:                                           */
/*   a) import this .rls file                                           */
/*   b) recompile your rules base                                       */
/*   c) reload the rules base                                           */
/*   d) restart the event server                                        */
/*======================================================================*/


/************************************************************************/
/************************   RULES   ***************************/
/************************************************************************/


/*======================================================================*/
/* This rule will forward all 'CRITICAL' and 'FATAL' events that did    */
/* not come from the NetView/390 message adapter or alert adapter.      */
```

```
/*=====================================================================*/
rule: fwd_to_nv390:
  (
  event: _event of_class _ev_class
    where
      [
      source: outside ['NV390MSG', 'NV390ALT'],
      severity: within ['CRITICAL', 'FATAL']
      ],
  reception_action:
    (
    forward_event(_event)
    )
  ).

/*=====================================================================*/
/* This rule will forward 'CRITICAL' and 'FATAL' events that just      */
/* changed to 'CLOSED' if they did not come from the NetView/390       */
/* message adapter or alert adapter.                                   */
/*=====================================================================*/
change_rule: fwd_closed_to_nv390:
  (
  event: _event of_class _ev_class
    where
      [
      source: outside ['NV390MSG', 'NV390ALT'],
      severity: within ['CRITICAL', 'FATAL']
      ],
  slot: status set_to _status
    within ['CLOSED'],
  action:
    (
    forward_event(_event)
    )
  ).

/*=====================================================================*/
/* This rule will forward events that just changed to 'CRITICAL' or    */
/* 'FATAL' as long as the source was not the NetView/390 message       */
/* adapter or alert adapter.                                           */
/*=====================================================================*/
change_rule: fwd_severity_to_nv390:
  (
  event: _event of_class _ev_class
    where
      [
      source: outside ['NV390MSG', 'NV390ALT']
      ],
  slot: severity set_to _severity
    within ['CRITICAL', 'FATAL'],
  action:
    (
    forward_event(_event)
    )
  ).
```

B.2.5 ITSO Modified Rule - tecad_nv390fwd_itso.rls

This is our modified tecad_nv390fwd rule to ensure that we didn't get multiple alerts in NPDA. This modified rule will not forward events from the SENTRY source.

```
/*=====================================================================*/
/* Licensed Materials - Property of IBM                                */
/* 5697-B82                                                            */
/* (C) Copyright IBM Corp. 1997.  All rights reserved.                 */
/*                                                                     */
/* US Government Users Restricted Rights - Use, duplication or         */
/* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.   */
/***********************************************************************/
/*                                                                     */
/* Description:                                                        */
/*    Sample rules for forwarding events to NetView/390               */
/*                                                                     */
/*=====================================================================*/


/*=====================================================================*/
/* To forward events you must:                                         */
/* 1) Edit the tec_forward.conf file and:                              */
/*    a) fill in ServerLocation                                        */
/*    b) comment out or delete the line: TestMode=yes                  */
/* 2) Update your rules base:                                          */
/*    a) import this .rls file                                         */
/*    b) recompile your rules base                                     */
/*    c) reload the rules base                                         */
/*    d) restart the event server                                      */
/*=====================================================================*/


/***********************************************************************/
/************************    RULES     *********************************/
/***********************************************************************/


/*=====================================================================*/
/* This rule will forward all 'CRITICAL' and 'FATAL' events that did   */
/* not come from the NetView/390 message adapter or alert adapter.     */
/*=====================================================================*/
rule: fwd_to_nv390:
  (
  event: _event of_class _ev_class
    where
      [
source: outside ['NV390MSG', 'NV390ALT', 'SENTRY'],
      severity: within ['CRITICAL', 'FATAL']
      ],
  reception_action:
    (
    forward_event(_event)
    )
  ).


/*=====================================================================*/
/* This rule will forward 'CRITICAL' and 'FATAL' events that just      */
/* changed to 'CLOSED' if they did not come from the NetView/390       */
/* message adapter or alert adapter.                                   */
/*=====================================================================*/
change_rule: fwd_closed_to_nv390:
```

```
        (
      event: _event of_class _ev_class
        where
          [
source: outside ['NV390MSG', 'NV390ALT', 'SENTRY'],
          severity: within ['CRITICAL', 'FATAL']
          ],
      slot: status set_to _status
        within ['CLOSED'],
      action:
        (
        forward_event(_event)
        )
      ).


/*=================================================================*/
/* This rule will forward events that just changed to 'CRITICAL' or  */
/* 'FATAL' as long as the source was not the NetView/390 message    */
/* adapter or alert adapter.                                        */
/*=================================================================*/
change_rule: fwd_severity_to_nv390:
      (
      event: _event of_class _ev_class
        where
          [
          source: outside ['NV390MSG', 'NV390ALT']
          ],
      slot: severity set_to _severity
        within ['CRITICAL', 'FATAL'],
      action:
        (
        forward_event(_event)
        )
      ).
```

B.2.6 Sample T/EC Rule for MSM Used in ITSO - new_Sentry.rls

This is our new forwardSentry rule that we created in the advanced scenario
described earlier.

```
/**********************************************************************/
/*                                                                    */
/* Description:                                                       */
/*    ITSO Rules for Sentry and MSM                                   */
/*                                                                    */
/*====================================================================*/


/*====================================================================*/
/* This set of rules avoid sending repeated events to T/EC and Netview*/
/* You need to change the tecad_nv390fwd.rls to ensure that SENTRY    */
/* events are not forwarded to NetView.                               */
/* The header of the rule should look like this:                      */
/*    source: outside ['NV390MSG', 'NV390ALT', 'SENTRY'],             */
/* The default forward_Sentry.rls should not be loaded into your      */
/* rule base.                                                         */
/*====================================================================*/


/**********************************************************************/
/************************    RULES    *********************************/
/**********************************************************************/


/*====================================================================*/
/* The rule msm_topo_event is used to send an event from the MSMAgent */
/* class to the T/EC console and to Netview/390.                      */
/* It's necessary because a change of Topology can never can be       */
/* considered a duplicate event.   [A]                                */
/*====================================================================*/


/*====================================================================*/
/* The rule sever_dup_event has three actions:                        */
/* 1st Action - Drops any duplicated event (A duplicated event is     */
/*   any event for which the previous one is OPEN and has the         */
/*   same severity as the new one). Then exit from this rule set. [B] */
/* 2nd Action - Forward any event that has an OPEN duplicate          */
/*   event but has a different severity. Close the previous event     */
/*   but don't forward it. It's used when a monitor                   */
/*   changes its severity. Then exit from this rule set.        [C]   */
/* 3rd Action - Forward any event that does not fit any of the        */
/*   previous rules. For instance, if there are no previous OPEN      */
/*   events or if it's the first time the event arrives.        [D]   */
/*====================================================================*/


/**********************************************************************/
/************************    RULES    *********************************/
/**********************************************************************/


rule: msm_topo_event:                 [A]
  (
  event: _event of_class 'MSMAgent',
  action:
    (
    forward_event(_event),
    commit_rule
    )
```

```
               ).
        rule: sever_dup_event:
          (
         event: _event of_class 'Sentry3_5_Base'
            where
             [
              severity:_severity
             ],
         action:                                          B
            (
           first_duplicate(_event,
             event: _sever_dup_ev
               where
                  [
                  status: outside ['CLOSED'],
                  severity: equals _severity
                  ]
              ),
           drop_received_event,
           commit_rule
           ),
         reception_action:                                C
            (
           first_duplicate(_event,
             event: _sever_dup_ev
               where
                  [
                  status: outside ['CLOSED'],
                  severity: outside [_severity]
                  ]
              ),
             set_event_status(_sever_dup_ev,'CLOSED'),
             forward_event(_event),
             commit_rule
           ),
         action:                                          D
            (
             forward_event(_event),
             commit_rule
           )
          ).
```

B.3 Baroc Files

Some of the various baroc files shipped with Global Enterprise Manager and the TME 10 products are included here. These baroc files define the event classes to T/EC.

B.3.1 Class Definitions for Sentry - Sentry.baroc

This is the Sentry.baroc file shipped with TME 10 Distributed Monitoring.

```
TEC_CLASS :
        Sentry2_0_Base ISA EVENT
        DEFINES {
                distrib_admin:  STRING, dup_detect= no ;
                response_level: STRING, dup_detect= no ;
                probe_arg:      STRING, dup_detect= no ;
                prev_value:     STRING, dup_detect= no ;
                value:          STRING, dup_detect= no ;
                effective_value:STRING, dup_detect= no ;
                collection:     STRING, dup_detect= no ;
                info:           STRING, dup_detect= no ;
                monitor:        STRING, dup_detect= no ;
                units:          STRING, dup_detect= no ;
                relation:       STRING, dup_detect= no ;
                relation_delta: STRING, dup_detect= no ;
                msg:            STRING, dup_detect= no ;
                };
END

TEC_CLASS :
        Sentry3_5_Base ISA Sentry2_0_Base
        DEFINES {
                probe:          STRING, dup_detect= no ;
                tmr:            STRING, dup_detect= no ;
                dispatcher:     STRING, dup_detect= no ;
                };
END
```

B.3.2 Class Definitions for Message Adapter - tecad_nv390msg.baroc

This is the default baroc file for the message adapter. It defines the different event classes used by the meassage adapter.

```
#*======================================================================*
# Licensed Materials - Property of IBM
# 5697-B82
# (C) Copyright IBM Corp. 1997.  All rights reserved.
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#**********************************************************************
#
# Description:
#   Default set of T/EC message class definitions for IBM's NetView/390
#   Message Adapter.
#
# Note: The Rules (e.g. tecad_nv390msg.rls) in the T/EC Event Server
#    only fire against leaf class events.  So be careful if you decide
#    to subclass an existing class here in the tecad_nv390msg.baroc
#    file, because if you accidentally change a leaf class to a non-leaf
#    class by adding a new subclass, then the rules will no longer fire
#    against events received from the now non-leaf class.
#
#*======================================================================*
TEC_CLASS :
        NV390MSG_Event ISA EVENT
        DEFINES {
                source:            default="NV390MSG", dup_detect=yes;
                sub_source:        default="Unknown",  dup_detect=yes;
                origin:                                dup_detect=yes;
                sub_origin:        default="N/A";
                msg:                                   dup_detect=yes;
                msg_catalog:       default="none";
                msg_index:         default=0;
                repeat_count:      default=0;
                severity:          default=WARNING;
                adapter_host_snanode: STRING;
                msg_id: STRING;                        # E.g. DSI200I
                multiline_msg: LIST_OF STRING,
                                   default=["N/A"];
                jobname: STRING,   default="N/A";
                user1: STRING;
                user2: STRING;
                user3: STRING;
                user4: STRING;
                user5: STRING;
                };
        END

## The Message Adapter fills in the following values in the following
## slots by default (can be overridden in the FMT and CDS files):
##
## source:      NV390MSG
## sub_source:  Component Name ("Unknown" by default in BAROC file,
##              can be overridden in FMT or CDS file).
## origin:      Same as hostname (netid.domainid of NetView where
##              message originated).
## sub_origin:  The job number.  It is not always present, and when
```

```
##               it isn't its value is "N/A".
## hostname:    netid.domainid of NetView where msg originated
## adapter_host: The IP name of the host where the NetView/390 Message
##               Adapter resides.
## date:        In format: Oct 12, hh:mm:ss
## status:      Default of OPEN in root.baroc file.
## severity:    Inferred from last char of msg id, e.g. DSI200I gets
##               assigned to WARNING (rather than HARMLESS, because many
##               NetView "I" msgs are true problems and not HARMLESS).
## msg:         Includes msg_id as first token.
## msg_catalog: Default of "none" in tecad_nv390msg.baroc file.
## msg_index:   Default of "0" in tecad_nv390msg.baroc file.
## repeat_count: Default of "0" in tecad_nv390msg.baroc file.
##
## NV390MSG_Event specific slots:
## ------------------------------
## adapter_host_snanode: The netid.nau name of the SNA host that
##               contains the Adapter.  E.g. NETA.CNM01 for NetView.
## msg_id:       First token of message.
## multiline_msg: For multiline messages, this LIST_OF slot contains
##               the second and succeeding messages.  If a message is
##               longer than 255 chars, it is broken into multiple msgs.
## jobname:      The job name.  It is not always present, and when it
##               isn't its value is "N/A".
##


#
# NV390MSG_Event subclasses follow.
#

# Leaf class for MVS messages.
TEC_CLASS :
        NV390MSG_MVS ISA NV390MSG_Event;
END

# Leaf class for VTAM messages.
TEC_CLASS :
        NV390MSG_VTAM ISA NV390MSG_Event;
END

# Leaf class for RODM messages.
TEC_CLASS :
        NV390MSG_RODM ISA NV390MSG_Event;
END

# Leaf class for LAN Network Manager messages.
TEC_CLASS :
        NV390MSG_LAN_Network_Manager ISA NV390MSG_Event;
END

# Non-leaf class for all NetView messages.
TEC_CLASS :
        NV390MSG_NetView_All_Components ISA NV390MSG_Event;
END

# Leaf class for NetView NCCF component messages.
TEC_CLASS :
        NV390MSG_NetView_NCCF ISA NV390MSG_NetView_All_Components;
END
```

```
# Leaf class for NetView GMFHS component messages.
TEC_CLASS :
        NV390MSG_NetView_GMFHS ISA NV390MSG_NetView_All_Components;
END

# Non-leaf class for all NetView Hardware Monitor (NPDA) messages.
TEC_CLASS :
        NV390MSG_NetView_NPDA_All ISA NV390MSG_NetView_All_Components;
END

# Leaf class for a specific NetView Hardware Monitor problem.
TEC_CLASS :
        NV390MSG_NetView_NPDA_inactive ISA NV390MSG_NetView_NPDA_All;
END

# Leaf class for all remaining NetView Hardware Monitor (NPDA) messages.
TEC_CLASS :
        NV390MSG_NetView_NPDA ISA NV390MSG_NetView_NPDA_All;
END

# Leaf class for NetView Session Monitor (NLDM) component messages.
TEC_CLASS :
        NV390MSG_NetView_NLDM ISA NV390MSG_NetView_All_Components;
END

# Leaf class for NetView SNA Topology Manager component messages.
TEC_CLASS :
        NV390MSG_NetView_SNA_Topo_Mgr ISA
            NV390MSG_NetView_All_Components;
END

# Leaf class for all other messages sent from the NetView/390
# Message Adapter.
TEC_CLASS :
        NV390MSG_Default ISA NV390MSG_Event;
END
```

B.3.3 Class Definitions for SNA Alert Adapter - tecad_snaevent.baroc

This is the default baroc file for the alert adapter defines all the event classes for
SNA alerts.

```
#*=====================================================================*
# Licensed Materials - Property of IBM
# 5697-B82
# (C) Copyright IBM Corp. 1997.  All rights reserved.
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#*********************************************************************
#
# Description:
#   Default set of T/EC alarm class definitions for IBM's System Network
#   Architecture (SNA) Alert Adapters.  This includes the NetView/390
#   Alert Adapter and AS/400 Alert Adapter.
#
# Note: The Rules (e.g. snaevent.rls) in the T/EC Event Server only fire
#   against leaf class events.  So be careful if you decide to subclass
#   an existing class here in the snaevent.baroc file, because if you
#   accidentally change a leaf class to a non-leaf class by adding a
#   new subclass, then the rules will no longer fire against events
#   received from the now non-leaf class.
#
#*=====================================================================*


ENUMERATION ARCH_TYPE
        0       NONGENERIC_ALERT
        1       GENERIC_ALERT
        2       GENERIC_RESOLUTION
END

TEC_CLASS :
        SNA_Event ISA EVENT
        DEFINES {
                source:             default="NV390ALT";
                sub_source:         default="NET";
                origin:                                 dup_detect=yes;
                sub_origin:         default="N/A";
                adapter_host:       default="N/A";
                msg_catalog:        default="none";
                msg_index:          default=0;
                msg:                                    dup_detect=yes;
                repeat_count:       default=0;
                severity:           default=CRITICAL;
                adapter_host_snanode: STRING;
                event_type: STRING, default="PERMANENT", dup_detect=yes;
                arch_type: ARCH_TYPE, default=GENERIC_ALERT;
                product_id: STRING, default="N/A";
                alert_id: STRING,   default="N/A";
                block_id: STRING,   default="N/A";
                action_code: STRING, default="N/A";
                alert_cdpt: STRING;                                 # From Sv92
                self_def_msg: LIST_OF STRING, default=[N/A];  # From Sv31
                detailed_data: LIST_OF STRING, default=[N/A]; #From Sv98
                event_correl: LIST_OF STRING, default=[N/A];  # From Sv47
                incident_correl: LIST_OF STRING,                    # From Sv4A
                            default=[N/A],      dup_detect=yes;
```

```
                    adapter_correl: STRING;
                    causes: LIST_OF STRING, default=[N/A];
                                    # From Sv93, Sv94, Sv95, and Sv96
                    actions: LIST_OF STRING, default=[N/A];
                                    # From Sv94, Sv95, Sv96, and Sv97
                    user1: STRING;
                    user2: STRING;
                    user3: STRING;
                    user4: STRING;
                    user5: STRING;
                    };
END


## The Alert Adapter fills in the following values in the following
## slots by default (can be overridden in the CDS file):
##
## source:        Depends on the SNA Alert Adapter.  Is NV390ALT for the
##                NetView/390 Alert Adapter.
## sub_source:    Default of "NET" in snaevent.baroc file.
## origin:        SNA name/type hierarchy.
## sub_origin:    Last name/type pair of SNA hierarchy.
## hostname:      netid.nau (node name) of the node where the alert
##                originated.  For example, could be NETA.MYAS400 for an
##                AS/400 system.
## adapter_host: The IP name of the host where the Alert Adapter
##                resides.
## date:          E.g. in format: OCT 12 hh:mm:ss
## status:        Default of OPEN in ROOT.BAROC file.
## severity:      Default of CRITICAL in snaevent.baroc file.
## msg:           Long Error Description:Long Probable Cause
## msg_catalog:  Default of "none" in snaevent.baroc file.
## msg_index:    Default of "0" in snaevent.baroc file.
## repeat_count: Default of "0" in snaevent.baroc file.
##
## SNA_Event specific slots:
## -------------------------
## adapter_host_snanode: The netid.nau name of the SNA host that
##                contains the Adapter.  E.g. NETA.CNM01 for NetView/390.
## event_type:    "PERMANENT", "TEMPORARY", etc..
## arch_type:    "GENERIC_ALERT", "GENERIC_RESOLUTION", or
##                "NONGENERIC_ALERT".
## product_id:    For generic alerts (Mv0000 with Sv92, all Mv0002's),
##                contains the product id from Sv10.
## alert_id:      For Mv0000 generic alerts, contains the alert id from
##                Sv92.
## block_id:      For nongeneric alerts (including Mv0000 with Sv91)
##                contains the block id.
## action_code:  For nongeneric alerts, contains the action code from
##                Sv91.
## alert_cdpt:    Contains the alert description codepoint from Sv92.
##                Nongeneric alerts are assigned a Sv92 codepoint that
##                best fits.
## self_def_msg: Message(s) obtained from Self-Defining Text Message
##                Sv31.
## detailed_data: Data obtained from Detailed Data Sv98.  This is
##                product specific detailed data.
## event_correl: From Sv47 of Mv0000 or Mv0002.
## incident_correl: From Sv4A of Mv0000 or Mv0002.
## adapter_correl: A correlator that has meaning only to the sending SNA
```

```
##                Alert Adapter.
## causes:        A list of text strings that describe the possible
##                causes of the alert.  Obtained by decoding the
##                codepoints in Sv93, Sv94, Sv95, and Sv96.
## actions:       A list of text strings that describe actions to take to
##                resolve the alert.  Obtained by decoding the codepoints
##                in Sv94, Sv95, Sv96, and Sv97.
##

# The SNA_Event subclasses are defined below.  They are based on
# the Alert Description code point (bytes 5 and 6) in Subvector 92 of
# SNA MS Major Vector 0000 (the Generic Alert major vector).  Refer to
# the "IBM SNA Formats" book for more info.  The major breakdown is as
# follows:
#    X'0xxx'  Not presently used, reserved by IBM.
#    X'1xxx'  Hardware
#    X'2xxx'  Software
#    X'3xxx'  Communications
#    X'4xxx'  Performance
#    X'5xxx'  Congestion
#    X'6xxx'  Microcode
#    X'7xxx'  Operator
#    X'8xxx'  Specification
#    X'9xxx'  Intervention Required
#    X'Axxx'  Problem Resolved.  Now retired, but some products still
#             send it.
#    X'Bxxx'  Notification
#    X'Cxxx'  Security
#    X'Dxxx'  Not presently used, reserved by IBM.
#    X'Exxx'  Reserved for non-IBM products and customer written appls.
#    X'Fxxx'  Undetermined
# Not all alerts received by NetView/390 are Generic Alerts.
# Non-generic alerts will be mapped to the generic alert subclass that
# best fits, or, if no appropriate generic alert subclass can be
# determined, they will be mapped to subclass
# SNA_NonGeneric_Undetermined.

#
# Subclass SNA_1xxx_Hardware and its subclasses follow.
#

TEC_CLASS :
        SNA_1xxx_Hardware ISA SNA_Event;
END

TEC_CLASS :
        SNA_Equipment_Malfunction ISA SNA_1xxx_Hardware;
END

TEC_CLASS :
        SNA_Input_Device_Error ISA SNA_1xxx_Hardware;
END

TEC_CLASS :
        SNA_Output_Device_Error ISA SNA_1xxx_Hardware;
END

TEC_CLASS :
```

```
            SNA_Input_Output_Device_Error ISA SNA_1xxx_Hardware;
END

TEC_CLASS :
        SNA_Loss_Of_Electrical_Power ISA SNA_1xxx_Hardware;
END

TEC_CLASS :
        SNA_Loss_Of_Equipment_Cooling_Or_Heating
            ISA SNA_1xxx_Hardware;
END

TEC_CLASS :
        SNA_Subsystem_Failure ISA SNA_1xxx_Hardware;
END

# All other 1xxx codepoints are assigned to this catch-all
# SNA_Hardware (leaf) class.
TEC_CLASS :
        SNA_Hardware ISA SNA_1xxx_Hardware;
END

#
# Subclass SNA_2xxx_Software and its subclasses follow.
#
TEC_CLASS :
        SNA_2xxx_Software ISA SNA_Event;
END

TEC_CLASS :
        SNA_Software_Program_Abnormally_Terminated
            ISA SNA_2xxx_Software;
END

TEC_CLASS :
        SNA_Software_Program_Error ISA SNA_2xxx_Software;
END

TEC_CLASS :
        SNA_Software_Operation_Failure ISA SNA_2xxx_Software;
END

# All other 2xxx codepoints are assigned to this catch-all
# SNA_Software (leaf) class.
TEC_CLASS :
        SNA_Software ISA SNA_2xxx_Software;
END

#
# Subclass SNA_3xxx_Communications and its subclasses follow.
#
TEC_CLASS :
        SNA_3xxx_Communications ISA SNA_Event;
END

TEC_CLASS :
        SNA_Communication_Protocol_Error
            ISA SNA_3xxx_Communications;
END
```

```
TEC_CLASS :
        SNA_SNA_Protocol_Error ISA SNA_3xxx_Communications;
END

TEC_CLASS :
        SNA_LAN_Error ISA SNA_3xxx_Communications;
END

TEC_CLASS :
        SNA_Link_Error ISA SNA_3xxx_Communications;
END

TEC_CLASS :
        SNA_ISDN_Error ISA SNA_3xxx_Communications;
END

TEC_CLASS :
        SNA_Local_Connection_Error ISA SNA_3xxx_Communications;
END

TEC_CLASS :
        SNA_Link_Connection_Error ISA SNA_3xxx_Communications;
END

TEC_CLASS :
        SNA_BBNS_Communications_Error ISA SNA_3xxx_Communications;
END

# All other 3xxx codepoints are assigned to this catch-all
# SNA_Communications (leaf) class.
TEC_CLASS :
        SNA_Communications ISA SNA_3xxx_Communications;
END

#
# Subclass SNA_4xxx_Performance and its subclasses follow.
#
TEC_CLASS :
        SNA_4xxx_Performance ISA SNA_Event;
END

TEC_CLASS :
        SNA_Performance_Degraded ISA SNA_4xxx_Performance;
END

# All other 4xxx codepoints are assigned to this catch-all
# SNA_Performance (leaf) class.
TEC_CLASS :
        SNA_Performance ISA SNA_4xxx_Performance;
END

#
# Subclass SNA_5xxx_Congestion and its subclasses follow.
#
TEC_CLASS :
        SNA_5xxx_Congestion ISA SNA_Event;
END
```

```
TEC_CLASS :
        SNA_Congestion ISA SNA_5xxx_Congestion;
END

TEC_CLASS :
        SNA_Configurable_Capacity_Limit_Reached
            ISA SNA_5xxx_Congestion;
END

# All other 5xxx codepoints are assigned to this catch-all
# SNA_Congestion_Other (leaf) class.
TEC_CLASS :
        SNA_Congestion_Other ISA SNA_5xxx_Congestion;
END

#
# Subclass SNA_6xxx_Microcode and its subclasses follow.
#
TEC_CLASS :
        SNA_6xxx_Microcode ISA SNA_Event;
END

TEC_CLASS :
        SNA_Microcode_Program_Abnormally_Terminated
            ISA SNA_6xxx_Microcode;
END

TEC_CLASS :
        SNA_Microcode_Program_Error ISA SNA_6xxx_Microcode;
END

TEC_CLASS :
        SNA_Microcode_Program_Mismatch ISA SNA_6xxx_Microcode;
END

# All other 6xxx codepoints are assigned to this catch-all
# SNA_Microcode (leaf) class.
TEC_CLASS :
        SNA_Microcode ISA SNA_6xxx_Microcode;
END

#
# Subclass SNA_7xxx_Operator and its subclasses follow.
#
TEC_CLASS :
        SNA_7xxx_Operator ISA SNA_Event;
END

TEC_CLASS :
        SNA_Operator_Procedural_Error ISA SNA_7xxx_Operator;
END

# All other 7xxx codepoints are assigned to this catch-all
# SNA_Operator (leaf) class.
TEC_CLASS :
        SNA_Operator ISA SNA_7xxx_Operator;
END
```

```
                   #
                   # Subclass SNA_8xxx_Specification and its subclasses follow.
                   #
                   TEC_CLASS :
                          SNA_8xxx_Specification ISA SNA_Event;
                   END

                   TEC_CLASS :
                          SNA_Configuration_Or_Customization_Error
                             ISA SNA_8xxx_Specification;
                   END

                   # All other 8xxx codepoints are assigned to this catch-all
                   # SNA_Specification (leaf) class.
                   TEC_CLASS :
                          SNA_Specification ISA SNA_8xxx_Specification;
                   END

                   #
                   # Subclass SNA_9xxx_Intervention_Required and its subclasses follow.
                   #
                   TEC_CLASS :
                          SNA_9xxx_Intervention_Required ISA SNA_Event;
                   END

                   TEC_CLASS :
                          SNA_Operator_Intervention_Required
                             ISA SNA_9xxx_Intervention_Required;
                   END

                   TEC_CLASS :
                          SNA_Stock_Low ISA SNA_9xxx_Intervention_Required;
                   END

                   TEC_CLASS :
                          SNA_Stock_Exhausted ISA SNA_9xxx_Intervention_Required;
                   END

                   TEC_CLASS :
                          SNA_Depository_Full ISA SNA_9xxx_Intervention_Required;
                   END

                   # All other 9xxx codepoints are assigned to this catch-all
                   # SNA_Intervention_Required (leaf) class.
                   TEC_CLASS :
                          SNA_Intervention_Required ISA SNA_9xxx_Intervention_Required;
                   END

                   #
                   # Subclass SNA_Axxx_Problem_Resolved and its subclasses follow.  The
                   # Axxx codepoints have been retired, but some products still send them.
                   #
                   TEC_CLASS :
                          SNA_Axxx_Problem_Resolved ISA SNA_Event;
                   END

                   # All Axxx codepoints are assigned to this catch-all
                   # SNA_Problem_Resolved (leaf) class.
                   TEC_CLASS :
```

```
            SNA_Problem_Resolved ISA SNA_Axxx_Problem_Resolved;
END


#
# Subclass SNA_Bxxx_Notification and its subclasses follow.
#
TEC_CLASS :
        SNA_Bxxx_Notification ISA SNA_Event;
END

TEC_CLASS :
        SNA_Operator_Notification ISA SNA_Bxxx_Notification;
END

TEC_CLASS :
        SNA_Environmental_Problem ISA SNA_Bxxx_Notification;
END

TEC_CLASS :
        SNA_Resent_Alert_With_Updated_Information ISA
            SNA_Bxxx_Notification;
END

# All other Bxxx codepoints are assigned to this catch-all
# SNA_Notification (leaf) class.
TEC_CLASS :
        SNA_Notification ISA SNA_Bxxx_Notification;
END


#
# Subclass SNA_Cxxx_Security and its subclasses follow.
#
TEC_CLASS :
        SNA_Cxxx_Security ISA SNA_Event;
END

TEC_CLASS :
        SNA_Security_Event ISA SNA_Cxxx_Security;
END

# All other Cxxx codepoints are assigned to this catch-all
# SNA_Security (leaf) class.
TEC_CLASS :
        SNA_Security ISA SNA_Cxxx_Security;
END


#
# Subclass SNA_Exxx_Non_IBM_Codepoint follows.  It is a leaf class.
#
TEC_CLASS :
        SNA_Exxx_Non_IBM_Codepoint ISA SNA_Event;
END


#
# Subclass SNA_Fxxx_Undetermined follows.
#
TEC_CLASS :
        SNA_Fxxx_Undetermined ISA SNA_Event;
END
```

```
        TEC_CLASS :
                SNA_Undetermined_Error ISA SNA_Fxxx_Undetermined;
        END


        # All other Fxxx codepoints are assigned to this catch-all
        # SNA_Undetermined (leaf) class.
        TEC_CLASS :
                SNA_Undetermined ISA SNA_Fxxx_Undetermined;
        END


        #
        # Subclass SNA_NonGeneric_Undetermined follows.
        #
        # For non-generic alerts, the Alert Adapter attempts to map the block id
        # and action code to the alert description codepoint which best fits.
        # If the Alert Adapter can not determine which alert description
        # codepoint to map this non-generic alert to, then the Alert Adapter
        # maps the "FE00" UNDETERMINED ERROR alert description codepoint to the
        # non-generic alert.  The non-generic alert is then, by default,
        # assigned to class SNA_NonGeneric_Undetermined in the CDS file.
        #
        TEC_CLASS :
                SNA_NonGeneric_Undetermined ISA SNA_Event;
        END


        #
        # Subclass SNA_Reserved_By_IBM follows.  It is a leaf class.  As of
        # 12/03/96, no products should send alerts with an alert description
        # codepoint of "Oxxx" or "Dxxx", but if they (incorrectly) do, such
        # alerts are assigned to this class.
        #
        TEC_CLASS :
                SNA_Reserved_By_IBM ISA SNA_Event;
        END
```

B.4 Shell Scripts Used

Global Enterprise Manager provides a shell script to make installation and customization easier. We have included the one provided by the product and one that we modified to work in our environment.

B.4.1 Installation Shell Script - gemevent.sh

This shell scripts runs in several steps.

1 Prompts for name of the MVS host, the name of the rule base to append to, and the name of the APM server.

2 Creates a rule base with the name of GemEvent.

3 Copies the default rule base into the new one.

4 Imports classes and rules of different products.

5 Creates tec_forward.conf and APM server configuration files.

6 The apmseverity script adds three new severity levels. They are severe, informational and normal.
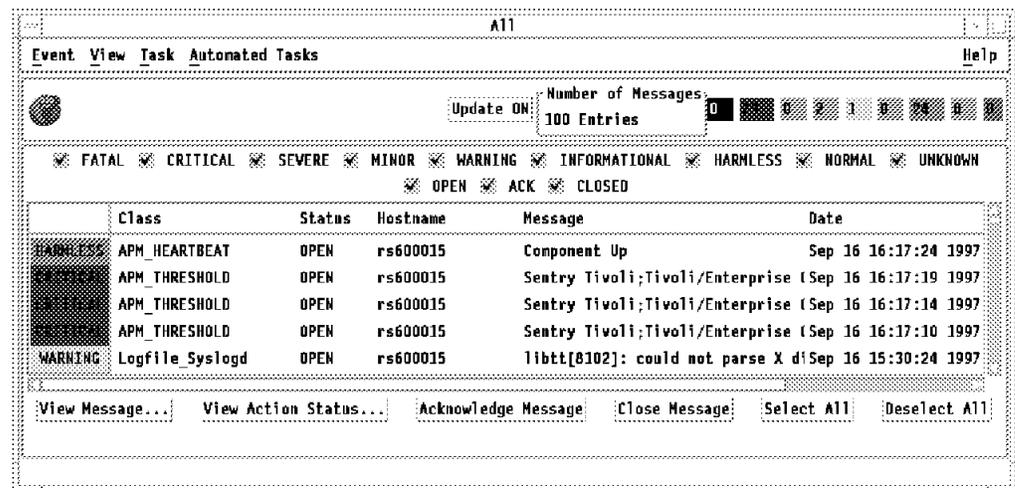
Figure 151. New severities on the T/EC console

7 Compiles and load the new rule base.

8 Creates event sources, and event groups with global filters.

9 Creates a UNIX and Tivoli user ID named GemAdmin and creates a console for that user with some event groups associated to it.

10 Stops and starts the event server.

```
#!/usr/bin/sh
###############################################################
#
# GEMEVENT: sample procedure for extending the Tivoli Event Console (TEC)
# and Tivoli Global Enterprose Manager (GEM) with a command line sample of:
#       -- installation of the object classes and rules for GEM.
#               --- Event Services
#               --- Storage Services
#               --- Topology Services
#               --- Application Policy Manager
#       -- creation of the event sources and event groups for GEM
#       -- definition of minimum filtering for the event groups
#       -- creation of a UNIX user and the associated Tivoli administrator
#       -- assignment of event groups to that administrator
#
# Before executing this procedure you must invoke the Tivoli setup.
# On Windows NT this procedure executes under the "bash" interpreter.
#
# for Windows_NT execute these two commands:
#
#           setup_env.cmd
#           bash $BINDIR/GEM/EventService/gemevent.sh
#
# for supported UNIX systems execute this command:
#
#           sh $BINDIR/GEM/EventService/gemevent.sh
###############################################################

####  CUSTOMER:  Enter the correct names on the following lines to
####  enable the creation of the APM and Event forwarding files.
####  These will be used as defaults when you are prompted later.
####  If you do not change the names below or at the prompt the files
####  tec_forward.conf and ihsttec.cnf will not be created.
####  If you do not wish to be prompted, set VariablePrompt=FALSE.
####  If you do not want sample user GemAdmin, set CreateUser=FALSE
####  or modify the create user section below.

###############################################################
#
# Set the default values for the prompting which will follow.
# If you are installing multiple TMRs with GEM you may want to
# customize these defaults for your enterprise.
#
###############################################################

#Enter the default OS/390 NetView or TEC master console IP hostname below:
OS390id="xxxxxxxx"

#Enter the default APM Server name below:
APMServer="xxxxxxxx"

#Enter the USERID for the sample GEM user below:
#The region name will be appended to this name to create the GEM Administrator ID
GemAdminID="GemAdmin"

#Enter the default name of the rules base to append to:
DefaultRulesBaseName="Default"
```

```
#Enable prompting for these values.
VariablePrompt=TRUE

#Enable creation of user and administrator for these values.
CreateUser=TRUE

#Import the file tecad_nv390fwd.rls
Importnv390fwdrls=N

#Import the file Sentry.baroc
ImportSentrybaroc=Y

#Import Tivoli Plus .baroc and .rls files
ImportTivoliPlus=Y

#Import Topology Services .baroc and .rls files
ImportTopologyServices=Y

#Import Application Policy Manager files (interapp.baroc and interapp.rls)
ImportAPMbaroc=Y
ImportAPMrls=N

##############################################################
#
# The following sets the environment so that we are sure that
# we can address all the needed Tivoli commands.  On Windows/NT
# the user has had to establish this environment and call "bash".
#
##############################################################

# get the operating system identity and set up for Tivoli commands
addr=${OS:=/usr/bin/uname}
if [ "$addr" != "Windows_NT" ]
  then . /etc/Tivoli/setup_env.sh  # setup for Tivoli environment
fi

if [ ! -d /tmp                 ]
  then mkdir /tmp                        # Temporary directory required
fi

##############################################################
#
# Prompt the user to pre-edit this file if needed.
#
##############################################################

echo '##############################################################'
echo 'Creating GemEvent rules data base and GemAdmin user.'
echo ' '
echo 'ATTENTION:'
echo "If you are installing multiple TMRs, have you updated"
echo "this file to include the defaults for your"
echo '  OS/390 NetView or TEC master server IP host?'
echo '  APM Server? '
echo '  Default Rules Base Name? '
echo '  Responses?'
echo ' '
echo 'Have you:'
echo '  reviewed the included rules and object files?'
```

```
echo '  commented out those you will not use?'
echo '  added any you need, especially other Sentry monitors? '
echo ' '
echo 'Have you modified the user creation portion for your system?'
echo ' '
echo 'The file name is ' $BINDIR/GEM/EventService/gemevent.sh
echo 'If you have not done so, please terminate now and reexecute.  '
echo '#########################################################'
echo '  Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
      exit 1
fi


#############################################################
#
# Prompt for the settings of the variables used to control the
# specific portions of the rules base included by the procedure.
#
#############################################################

# prevent prompting  by setting VariablePrompt to FALSE
# above in the default setting portion.
until [ "$VariablePrompt" = "FALSE" ]
do

echo ' '
echo 'Please enter the name of your NetView for OS/390'          1
echo 'or TEC Server IP host for forwarding events.'
echo 'If no name is entered we will use the name:' $OS390id
echo "If the name entered is 'xxxxxxxx', no setup will occur."
read UserReply
if [ q$UserReply != "q" ]
   then OS390id=$UserReply
fi

echo ' '
echo 'Please enter the name of your Application Policy Manager (APM) Server.'
echo 'If no name is entered we will use the name:' $APMServer
echo "If the name entered is 'xxxxxxxx', no setup will occur."
read UserReply
if [ q$UserReply != "q" ]
   then APMServer=$UserReply
fi


#############################################################
# Get the region name to build the administrator ID
#############################################################
IRO=wlookup InterRegion
IROname=idlattr -t -g $IRO name string
IROname=eval echo $IROname
GemAdminTivoli=$GemAdminID"_"$IROname

echo ' '
echo 'Please enter the USERID for the sample administrator to use.'
echo 'If no name is entered we will use the name' $GemAdminID
echo 'for the user and' $GemAdminTivoli 'for the administrator.'
echo "If the name entered is 'xxxxxxxx', no setup will occur."
```

```
read UserReply
if [ q$UserReply != "q" ]
   then GemAdminID=$UserReply
fi

echo ' '
echo 'Please enter the name of the TEC Rules Base to copy'
echo 'as a starting point for creating the GemEvent rules base.'
echo 'If no name is entered we will use the name:' $DefaultRulesBaseName
read UserReply
if [ q$UserReply != "q" ]
   then DefaultRulesBaseName=$UserReply
fi

#Import the file tecad_nv390fwd.rls
echo ' '
echo "Do you want to import the sample file tecad_nv390fwd.rls into"
echo "your rules base? These rules forward CRITICAL and FATAL events."
echo "If you use these rules you should make sure that no other rules"
echo "will cause duplicate events to be forwarded."
echo "Enter 'Y' to import this file."
echo "Enter 'N' to not import this file."
echo "The default is:" $Importnv390fwdrls
read UserReply
if [ q$UserReply != "q" ]
  then Importnv390fwdrls=$UserReply
fi


ImportSentrybaroc=$UserReply

#Import Tivoli Plus .baroc and .rls files
echo ' '
echo "Do you want to import Storage Services and Tivoli PLUS"
echo ".baroc and .rls files into your rules base?"
echo "Enter 'Y' to import these files."
echo "Enter 'N' to not import these files."
echo "The default is:" $ImportTivoliPlus
read UserReply
if [ q$UserReply != "q" ]
  then ImportTivoliPlus=$UserReply
fi

#Import Topology Services .baroc and .rls files
echo ' '
echo "Do you want to import the Topology Services"
echo ".baroc and .rls files into your rules base?"
echo "Enter 'Y' to import these files."
echo "Enter 'N' to not import these files."
echo "The default is:" $ImportTopologyServices
read UserReply
if [ q$UserReply != "q" ]
  then ImportTopologyServices=$UserReply
fi

if [ qImportTopologyServices != 'qn' -o qImportTopologyServices != 'qN\
        -o qImportTivoliPlus != 'qn' -o qImportTivoliPlus != 'qN' ]
  then ImportSentryBaroc='Y'
fi
```

```
#Import Application Policy Manager interapp.baroc file
echo ' '
echo "Do you want to import the Application Policy Manager"
echo "file interapp.baroc into your rules base?"
echo "Enter 'Y' to import this file."
echo "Enter 'N' to not import this file."
echo "The default is:" $ImportAPMbaroc
read UserReply
if [ q$UserReply != "q" ]
  then ImportAPMbaroc=$UserReply
fi


if [ qImportAPMbaroc != "qn" ] && [ qImportAPMbaroc !=   "qN" & rbrk.
   then
      #Import Application Policy Manager interapp.rls file
      echo ' '
      echo "Do you want this TEC to forward APM events to the GEM APM server?"
      echo "Enter 'Y' to import the interapp.rls file."
      echo "Enter 'N' if this is not the master TEC."
      echo "The default is:" $ImportAPMrls
      read UserReply
      if [ q$UserReply != "q" ]
        then ImportAPMrls=$UserReply
      fi
fi


VariablePrompt=FALSE       # prevent iteration of User prompting
done                 # end of UserReplying


##############################################################
#
# Create the rules base with the name of GemEvent,
# copy the default Tivoli rules base into it,
# import the distributed GEM object definitions and rules into the
# rules base, compile and load the rules base.
#
##############################################################

echo 'Create the GemEvent Rules Base'
set -x      #this causes the commands to display as they are executed

##############################################################
#
# erase the existing rules base in case there is one.
#
##############################################################
wdelrb GemEvent
rm -fr  $BINDIR/GEM/EventService/GemEvent_rb

##############################################################
#
# Create the directory and rules base structure.
# This creates a directory named GemEvent_rb and the three
# directories TEC_CLASSES, TEC_RULES and TEC_TEMPLATES
#
##############################################################
wcrtrb -d $BINDIR/GEM/EventService/GemEvent_rb GemEvent        2
```

```
##############################################################
#
# copy the existing system default rules base or the customer designated
# rules base (the three TEC_* directories to GemEvent_rb.
#
##############################################################
wcprb $DefaultRulesBaseName GemEvent                  3


##############################################################
# import the event services objects and rules to the rules base
# these handle events received from the NetView for OS/390 event adapters
##############################################################
wimprbclass   $BINDIR/GEM/EventService/tecad_nv390msg.baroc GemEvent   4
wimprbrules   $BINDIR/GEM/EventService/tecad_nv390msg.rls    GemEvent
set +x


##############################################################
# Import these two only if not in the default rules base.
# Starting with TEC 3.1 they will be shipped with the default rules
# base because they are needed for NetView for OS/390 and AS/400
# These files will not be present in the $BINDIR/GEM/EventService
# directory if TEC 3.1 or greater is present.
##############################################################
if [ ! -f $BINDIR/TME/TEC/default_rb/TEC_CLASSES/tecad_snaevent.baroc ]
  then
    set -x
    wimprbclass   $BINDIR/GEM/EventService/tecad_snaevent.baroc GemEvent    4
    set +x
fi
if [ ! -f $BINDIR/TME/TEC/default_rb/TEC_RULES/tecad_snaevent.rls ]
  then
    set -x
    wimprbrules   $BINDIR/GEM/EventService/tecad_snaevent.rls    GemEvent    4
    set +x
fi


##############################################################
# This is a sample set of rules to forward CRITICAL and FATAL events.
# If you use these rules you should make sure that no other rules will
# cause duplicate events to be forwarded.
##############################################################
if [ q$Importnv390fwdrls != "qn" ] && [ q$Importnv390fwdrls != "qN" ]
  then
    set -x
    wimprbrules   $BINDIR/GEM/EventService/tecad_nv390fwd.rls    GemEvent    4
    set +x
fi


##############################################################
# This is for the GEM Topology Services and Application Policy Manager
##############################################################
if [ q$ImportSentryBaroc != "qn" ]  && [ q$ImportSentryBaroc != "qN" ]
  then
    set -x
    wimprbclass   $BINDIR/../generic/SentryMonitors/Sentry.baroc GemEvent   4
    set +x
fi
```

```
##############################################################
# the following are for GEM Storage Services and Tivoli Plus
##############################################################
if [ q$ImportTivoliPlus != "qn" ] &&  [ q$ImportTivoliPlus != "qN" ]
  then
    set -x
    wimprbclass  $BINDIR/../generic_unix/TME/PLUS/ADSM/adsm.baroc       GemEvent       4
    wimprbrules  $BINDIR/../generic_unix/TME/PLUS/ADSM/adsm.rls          GemEvent
    set +x
    if [ q$ImportTopologyServices = "qn" ] ||[ q$ImportTopologyServices = "qN" ]
       then
         set -x
         wimprbclass  $BINDIR/../generic_unix/TME/PLUS/LINK/Sentry2.OMonitors.baroc    GemEvent      4
         set +x
    fi
    set -x
    wimprbrules  $BINDIR/../generic_unix/TME/PLUS/LINK/tivoli_plus.rls GemEvent       4
    set +x
fi

echo 'Terminate execution if any errors occured during these imports.'
echo '   Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
      exit 1
fi

##############################################################
# the following are for GEM Topology Services
##############################################################
if [ q$ImportTopologyServices != "qn" ]  && [ q$ImportTopologyServices != "qN" ]
  then
    set -x
    wimprbclass  $BINDIR/../generic/SNMP/Compaq.baroc                   GemEvent    4
    wimprbclass  $BINDIR/../generic/SNMP/UserSNMP.baroc                 GemEvent
    wimprbclass  $BINDIR/../generic/SNMP/rfc1213.baroc                  GemEvent
    wimprbclass  $BINDIR/../generic/SentryMonitors/tivoli.baroc         GemEvent
    wimprbclass  $BINDIR/../generic/SentryMonitors/universal.baroc      GemEvent
    wimprbclass  $BINDIR/../generic_unix/GEM/MSMAgent/MSMAgent.baroc  GemEvent
    wimprbrules  $BINDIR/../generic_unix/GEM/MSMAgent/forwardSentry.rls GemEvent
    set +x
    echo "WARNING: forwardSentry.rls has been imported into your rules base."
    echo "WARNING: This may cause duplicate event forwarding, please check"
    echo "WARNING: your rules to see of this may happen."
fi

##############################################################
# the following are for the GEM Application Policy Manager
##############################################################
if [ q$ImportAPMbaroc != "qn" ] && [ q$ImportAPMbaroc !=  "qN" ]
  then
    set -x
    wimprbclass $BINDIR/GEM/EventService/interapp.baroc         GemEvent
    set +x
fi

if [ q$ImportAPMrls != "qn" ] &&  [ q$ImportAPMrls != "qN" ]
  then
    set -x
```

```
        wimprbrules $BINDIR/GEM/EventService/interapp.rls        GemEvent
        set +x
fi


#############################################################
# Allow the user to cancel on the results of the rest of the imports.
#############################################################
echo ' '
echo 'Terminate execution if any errors occured during these imports.'
echo '  Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
        exit 1
fi


GEMDATE=date +"%c"
export GEMDATE
GEMFORWARD=$BINDIR/GEM/EventService/GemEvent_rb/TEC_RULES/tec_forward.conf
GEMAPMSERVER=$BINDIR/GEM/EventService/ihsttec.cfg
echo ' '
echo 'Create the tec_forward.conf file'    5
if [ $OS390id != "xxxxxxxx" ]
  then
    set -x
    echo "#TEC Server configuration file for forwarding events" >$GEMFORWARD
    echo "# Generated by TME 10 Global Enterprise Manager $GEMDATE"  >>$GEMFORWARD
    echo "ServerLocation=$OS390id" >>$GEMFORWARD
    echo "# TestMode=yes" >>$GEMFORWARD
    echo "EventMaxSize=4096" >>$GEMFORWARD
    echo "# ConnectionMode=connection_oriented"  >>$GEMFORWARD
    echo "# Filter:Class=Logfile_Base:"  >>$GEMFORWARD
    set +x
  else
    echo "tec_forward.conf was not created because a NetView for OS/390"
    echo "or TEC server IP hostname was not specified."
    echo ' '
fi


#############################################################
#
# Create the APM Server configuration file unless no good
# APM Server hostname was provided
#
#############################################################
if [ $APMServer != "xxxxxxxx" ]
  then
    echo 'Create the APM configuration file'  5
    set -x
    echo "#APM Server configuration file for forwarding events" >$GEMAPMSERVER
    echo "# Generated by TME 10 Global Enterprise Manager $GEMDATE" >>$GEMAPMSERVER
    echo "SERVER_HOST=$APMServer" >>$GEMAPMSERVER
    echo ' '
    set +x
  else
    echo "ihsttec.cfg was not created because an APM server name was not specified."
    echo ' '
fi
```

```
###############################################################
#
# Until the updated TEC root.baroc file is shipped, add severity
# levels to the root.baroc file to support all AMS severities
# Determine the system type and adjust the awk invocation appropriately
#
###############################################################
echo 'Updating TEC severity levels in root.baroc'
.$BINDIR/GEM/EventService/apmseverity.sh          6

###############################################################
#
# Rules bases must be compiled and loaded for the TEC Server and
# Event Services to function.  Compilation takes the baroc and rls
# files and creates PROLOG expansions.  Loading copies the rules base
# to the operational location $DBDIR/tec/rb_dir
#
###############################################################

echo ' '
echo 'Terminate execution if any errors occured during the previous step.'
echo 'The next step is to compile GemEvent rules base.'
echo '  Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
        exit 1
fi

echo 'Compiling GemEvent rules base'
set -x
wcomprules GemEvent       7
set +x

echo 'Terminate execution if any errors occured during the rules compile'
echo 'The next step is to Load the GemEvent rules base.'
echo '  Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
        exit 1
fi

set -x
wloadrb -u GemEvent     7
set +x

###############################################################
#
#  Event Sources and Groups must be created.  They may have been created while
#  installing TEC 3.1.  If so we will update the icons chosen.   8
#  If not we will create the appropriate event groups.
#
###############################################################

echo 'Terminate execution if any errors occured during the rules load.'
echo 'The next step is to Create Event Sources and Event Groups.'
echo '  Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
```

```
          exit 1
fi


##############################################################
#
#  Event sources are displayed in a horizontal window.
#  Event sources must exist to be clustered in classes.
#  If the TEC installation has created them we will select bitmaps
#  corresponding to the sources, otherwise we'll create them.
#
##############################################################

echo 'Creating Event Sources and Event Groups'
if [ wlssrc NV390ALT ] grep NV390ALT = 'NV390ALT' ]
  then
    set -x
    wsetsrc -b "genmainframe48" NV390ALT
    set +x
  else
    set -x
    wcrtsrc -b "genmainframe48" NV390ALT
    set +x
fi
if [ wlssrc NV390MSG] grep NV390MSG = 'NV390MSG' ]
  then
    set -x
    wsetsrc -b "genmainframe48" NV390MSG
    set +x
  else
    set -x
    wcrtsrc -b "genmainframe48" NV390MSG
    set +x
fi
if [ wlssrc LOGFILE ] grep LOGFILE = 'LOGFILE' ]
  then
    set -x
    wsetsrc -b "logf48"         LOGFILE
    set +x
  else
    set -x
    wcrtsrc -b "logf48"         LOGFILE
    set +x
fi
if [ wlssrc SENTRY ] grep SENTRY = 'SENTRY' ]
  then
    set -x
    wsetsrc -b "sentry48"       SENTRY
    set +x
  else
    set -x
    wcrtsrc -b "sentry48"       SENTRY
    set +x
fi
if [ wlssrc TEC ] grep TEC = 'TEC' ]
  then
    set -x
    wsetsrc -b "tecrsvr48"      TEC
    set +x
  else
```

```
       set -x
       wcrtsrc -b "tecrsvr48"        TEC
       set +x
fi


###############################################################
#
# Event groups are displayed in a vertical window.      [8]
# We have not duplicated the event sources but collected
# Alerts and Messages in one group and everything in another.
#
###############################################################

GEMTEST=wlseg NetView_390
if [ q$GEMTEST != "qNetView_390" ]
  then
    set -x
    wcrteg      -b "genmainframe48"          NetView_390
    waddegflt  -s "NV390ALT"                 NetView_390
    waddegflt  -s "NV390MSG"                 NetView_390
    set +x
fi
GEMTEST=wlseg All
if [ q$GEMTEST != "qAll" ]
  then
    set -x
    wcrteg      -b "collection"          All
    waddegflt                            All
    set +x
fi
GEMTEST=wlseg StorageServ
if [ q$GEMTEST != "qStorageServ" ]
  then
    set -x
    wcrteg      -b "db48"                     StorageServ
    waddegflt  -s "SENTRY"  -c "Sentry"      StorageServ
    waddegflt  -s "LOGFILE" -c "ADSM_BASE"  StorageServ
    set +x
fi
GEMTEST=wlseg TopologyServ
if [ q$GEMTEST != "qTopologyServ" ]
  then
    set -x
    wcrteg      -b "genapp48"                 TopologyServ
    waddegflt  -s "SENTRY"  -c "Sentry"      TopologyServ
    set +x
fi
GEMTEST=wlseg APM
if [ q$GEMTEST != "qAPM" ]
  then
    set -x
    wcrteg      -b "genapp48"                 APM
    waddegflt  -s "SENTRY"  -c "Sentry"      APM
    set +x
fi


###############################################################
#
# Prompt for the settings of the variables needed below.
```

```
# prevent by setting GEM Userid to "xxxxxxxx"
#
###############################################################
if [ q$GemAdminID != "qxxxxxxxx" ]
  then
    echo ' '
    echo 'The next step is to create a user named' $GemAdminID
    echo 'and an administrator named' $GemAdminTivoli
    echo 'with complete capabilities on a UNIX system.  Terminate'
    echo 'now if you do not want these names defined.  '
    echo '  Press Enter to continue or "q" to quit.'
    read UserReply
    if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
     exit 1
    fi

    ###############################################################
    #
    # the following prototype commands will create a UNIX userid and  9
    # a Tivoli administrator with the ID of GemAdmin.  It will then
    # assign the event groups to that id.
    #
    ###############################################################
    case $addr in
    SunOS ] HP-UX)
      set -x
      useradd     -g 'staff' -G 'sys,adm,other' \
                  -s '/bin/ksh' -d /home/$GemAdminID $GemAdminID
      set +x
      ;;
    AIX)
      set -x
      mkuser      -a pgrp='staff' groups='system,staff,usr' \
               shell='/bin/ksh' home=/home/$GemAdminID $GemAdminID
      set +x
      ;;
    Windows_NT)
      echo 'This is a Windows NT machine and you must create the GemAdmin'
      echo '    user manually (in another window) and proceed or terminate'
      echo '    the process. If you terminate you will have to create'
      echo '    TME 10 administrator desktops and event consoles manuallly'
      echo '    and assign the event groups to them.'
      echo 'You should also stop and start your event server.'
      echo ' '
      echo '  Press Enter to continue or "q" to quit.'
      read UserReply
      if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ]; then
       exit 1
      fi
      ;;
  esac

  set -x
  wcrtadmin -l $GemAdminID                             \
            -g staff                                   \
            -u $GemAdminTivoli                         \
            -n "Enterprise Console#"$IROname      \
            -r "@EventServer#"$IROname",super:senior:admin:user:RIM_update:RIM_view" $GemAdminTivoli
```

```
    wcrtconsole -l $GemAdminTivoli @$GemAdminTivoli

   wassigneg  @$GemAdminTivoli  NetView_390  senior admin user
   wassigneg  @$GemAdminTivoli  All          senior admin user
   wassigneg  @$GemAdminTivoli  StorageServ  senior admin user
   set +x

fi

#############################################################
#
# We need to stop and start the TEC Event Server  to activate the
# rules base we created, compiled and loaded earlier in this procedure
#
#############################################################

echo ' '
echo 'Stopping and restarting the Event Server'
echo '  Press Enter to continue or "q" to quit.'
read UserReply
if [ "$UserReply" = 'q' -o "$UserReply" = 'Q' ] ; then
 exit 1
fi

set -x
wstopesvr                 10
# Give the process time to complete before restarting.
sleep 10                  10
wstartesvr
set +x
```

B.4.2 Modified Installation Shell Script - itso_gem.sh

This is a mofified version of the gemevent.sh provided by Global Enterprise
Manager. We needed to change it to import and load the modified rules.

A new rule base called Roberto will be created, with the following changes:

1 All the baroc files for NT have been added.

2 The rule forwardSentry.rls has been commented.

3 The rule tecad_nv390fwd.rls has been commented.

Two rules have been added:

- **4** new_Sentry.rls: Imported first
- **5** tecad_nv390fwd_itso.rls

6 We create a filter for the Source SENTRY, and class MSMAgent.

```
#!/usr/bin/sh
#############################################################
#
# ITSO_GEM: procedure for extending the Tivoli Event Console (TEC) and
# Tivoli Global Enterprose Manager (GEM) with a command line sample of:
#      -- installation of the object classes and rules for GEM.
#           --- Event Services
#           --- Storage Services
```

```
#             --- Topology Services
#             --- Application Policy Manager
#     -- creation of the event sources and event groups for GEM
#     -- definition of minimum filtering for the event groups
#     -- creation of a UNIX user and the associated Tivoli administrator
#     -- assignment of event groups to that administrator
#
# Before executing this procedure you must invoke the Tivoli setup.
# On Windows NT this procedure executes under the "bash" interpreter.
#
# for Windows_NT execute these two commands:
#
#         setup_env.cmd
#         bash $BINDIR/GEM/EventService/gemevent.sh
#
# for supported UNIX systems execute this command:
#
#         sh $BINDIR/GEM/EventService/gemevent.sh
################################################################

####  CUSTOMER:  Enter the correct names on the following lines to
####  enable the creation of the APM and Event forwarding files.
####  These will be used as defaults when you are prompted later.
####  If you do not change the names below or at the prompt the files
####  tec_forward.conf and ihsttec.cnf will not be created.
####  If you do not wish to be prompted, set VariablePrompt=FALSE.
####  If you do not want sample user GemAdmin, set CreateUser=FALSE
####  or modify the create user section below.

#Enter the default OS/390 NetView IP host name below:
OS390id="MVS11"


#Enter the default APM Server name below:
APMServer="wtr05193"

#Enter the default name of the rules base to append to:
DefaultRulesBaseName="Default"

#Enable prompting for these values.
VariablePrompt=FALSE

#Enable creation of user and administrator for these values.
CreateUser=FALSE
################################################################

# get the operating system identity
addr=${OS:=/usr/bin/uname}

# Invoke the Tivoli setup procedure
if [ "$addr" != "Windows_NT" ]
  then . /etc/Tivoli/setup_env.sh  # setup for Tivoli environment
fi

if [ ! -d /tmp            ]
  then mkdir /tmp                   # Temporary directory required
fi
```

```
echo '###########################################################'
echo 'Creating GemEvent rules data base and GemAdmin user.'
echo ' '
echo 'ATTENTION:'
echo 'Have you updated this file to include the default names for your'
echo '  OS/390 NetView IP host? '
echo '  APM Server? '
echo '  Default Rules Base Name? '
echo ' '
echo 'Have you reviewed the included rules and object files and '
echo '  commented out those you will not use?'
echo '  added any you need, especially other Sentry monitors? '
echo ' '
echo 'Have you reviewed the user creation portion and modified it for'
echo '  your system or removed it and predefined user "GemAdmin"?'
echo ' '
echo 'The file name is ' $BINDIR/GEM/EventService/gemevent.sh
echo 'If you have not done so, please terminate now and reexecute.  '
echo '###########################################################'
echo '  Press Enter to proceed or Ctrl-C to terminate.'
read UserReply
############################################################
# Prompt for the settings of the variables needed below.
# prevent UserReplying by setting Prompt to FALSE
until [ "$VariablePrompt" = "FALSE" ]
do

echo 'Please enter the name of your OS/390 NetView IP host.'
echo 'If no name is entered we will use the name ' $OS390id
read UserReply
if [ q$UserReply != "q" ]
   then OS390id=$UserReply
fi
echo 'Please enter the name of your APM Server.'
echo 'If no name is entered we will use the name ' $APMServer
read UserReply
if [ q$UserReply != "q" ]
   then APMServer=$UserReply
fi

echo 'Please enter the name of the TEC Rules Base to copy'
echo 'as a starting point for creating the GemEvent rules base'
echo 'If no name is entered we will use the name ' $DefaultRulesBaseName
read UserReply
if [ q$UserReply != "q" ]
   then DefaultRulesBaseName=$UserReply
fi

VariablePrompt=FALSE        # prevent iteration of UserReplying
done                # end of UserReplying

############################################################
# Create the rules base with the name of GemEvent,
# copy the default Tivoli rules base into it,
# import the distributed GEM object definitions and rules into the rules
# base, compile and load the rules base.

set -x      #this causes the commands to display as executed
# erase the existing rules base in case there is one.
```

```
wdelrb Roberto
rm -fr  $BINDIR/GEM/EventService/Roberto_rb

# create the directory and rules base structure
wcrtrb -d $BINDIR/GEM/EventService/Roberto_rb Roberto

# copy the existing system default rules base to ours
wcprb $DefaultRulesBaseName Roberto
# import the Sentry rule for MSM use
wimprbrules  /usr/local/Tivoli_scripts/new_Sentry.rls Roberto                4

# import the event services objects and rules to the rules base
wimprbclass  $BINDIR/GEM/EventService/tecad_nv390msg.baroc Roberto
wimprbrules  $BINDIR/GEM/EventService/tecad_nv390msg.rls    Roberto
wimprbclass  $BINDIR/GEM/EventService/tecad_snaevent.baroc Roberto
wimprbrules  $BINDIR/GEM/EventService/tecad_snaevent.rls    Roberto
#wimprbrules  $BINDIR/GEM/EventService/tecad_nv390fwd.rls    Roberto         3
wimprbrules  /usr/local/Tivoli_scripts/tecad_nv390fwd_itso.rls    Roberto   5

# This is for the GEM Topology Services and Application Policy Manager
wimprbclass  $BINDIR/../generic/SentryMonitors/Sentry.baroc Roberto

# the following are for GEM Storage Services and Tivoli Plus
#wimprbclass  $BINDIR/../generic_unix/TME/PLUS/ADSM/adsm.baroc        Roberto
#wimprbrules  $BINDIR/../generic_unix/TME/PLUS/ADSM/adsm.rls          Roberto
#wimprbclass  $BINDIR/../generic_unix/TME/PLUS/LINK/Sentry2.0Monitors.baroc     Roberto
#wimprbrules  $BINDIR/../generic_unix/TME/PLUS/LINK/tivoli_plus.rls Roberto
set +x
echo 'Terminate execution  if any errors occured  during these imports.'
echo '  Press Enter to proceed or Ctrl-C to terminate.'
read UserReply
set -x
# the following are for GEM Topology Services
wimprbclass  $BINDIR/../generic/SNMP/Compaq.baroc                 Roberto
wimprbclass  $BINDIR/../generic/SNMP/UserSNMP.baroc               Roberto
wimprbclass  $BINDIR/../generic/SNMP/rfc1213.baroc               Roberto
wimprbclass  $BINDIR/../generic/SentryMonitors/tivoli.baroc       Roberto
wimprbclass  $BINDIR/../generic/SentryMonitors/universal.baroc     Roberto
#wimprbrules  $BINDIR/../generic_unix/GEM/MSMAgent/forwardSentry.rls Roberto  2
wimprbclass  $BINDIR/../generic_unix/GEM/MSMAgent/MSMAgent.baroc  Roberto

# Baroc files for NT Monitors     1

wimprbclass  $BINDIR/../generic/SentryMonitors/ntBrowser.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntCache.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntClientServerForNW.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntEventLog.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntICMP.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntIP.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntLogicalDisk.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntMemory.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNBTConnection.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNWLinkIPX.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNWLinkNetBIOS.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNetBEUI.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNetBEUIResource.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNetworkInterface.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntNetworkMonitor.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntObjects.baroc     GemEvent
```

```
wimprbclass  $BINDIR/../generic/SentryMonitors/ntPagingFile.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntPhysicalDisk.baroc    GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntProcess.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntProcessor.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntRedirector.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntServer.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntServerWorkQueue.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntSystem.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntTCP.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntThread.baroc     GemEvent
wimprbclass  $BINDIR/../generic/SentryMonitors/ntUDP.baroc     GemEvent

# the following are for the GEM Application Policy Manager
wimprbclass $BINDIR/GEM/EventService/interapp.baroc              Roberto
wimprbrules $BINDIR/GEM/EventService/interapp.rls              Roberto
set +x
echo 'Terminate execution  if any errors occured  during these imports.'
echo '  Press Enter to proceed or Ctrl-C to terminate.'
read UserReply


# Create the tec_forward.conf file
GEMDATE=date +"%c"
export GEMDATE
echo 'Create the tec_forward.conf file'
if [ $OS390id != "xxxxxxxx" ]
then
echo "#Config file for forwarding events" >$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
echo "# Generated by TME 10 GEM $GEMDATE"  >>$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
echo "ServerLocation=$OS390id" >>$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
echo "# TestMode=yes" >>$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
echo "EventMaxSize=4096" >>$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
echo "# ConnectionMode=connection_oriented"  >>$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
echo "# Filter:Class=Logfile_Base:"  >>$BINDIR/GEM/EventService/Roberto_rb/TEC_RULES/tec_forward.conf
fi

# Create the APM configuration file
if [ $APMServer != "xxxxxxxx" ]
then
echo 'Create the APM configuration file'
echo "#APM Server configuration file for forwarding events" >$BINDIR/GEM/EventService/ihsttec.cfg
echo "# Generated by TME 10 Global Enterprise Manager $GEMDATE" >>$BINDIR/GEM/EventService/ihsttec.cfg
echo "SERVER_HOST=$APMServer" >>$BINDIR/GEM/EventService/ihsttec.cfg
echo ' '
fi

echo 'Compiling Roberto rules base'
set -x
wcomprules -t Roberto
set +x
echo 'Terminate execution if any errors occured during the rules compile'
echo 'The next step is to Load the Roberto rules base and create groups.'
echo '  Press Enter to proceed or Ctrl-C to terminate.'
read UserReply

set -x
wloadrb -u Roberto
set +x
```

```
echo 'Creating Event Sources and Event Groups'
set -x
# Event sources are displayed in a horizontal window.
# Event sources must exist to be clustered in classes
wcrtsrc -l "NV390_Alert"   -b "genmainframe48" NV390ALT
wcrtsrc -l "NV390_Message" -b "genmainframe48" NV390MSG
wcrtsrc -l "Log_File"      -b "logf48"         LOGFILE
wcrtsrc -l "Sentry"        -b "sentry48"       SENTRY
wcrtsrc -l "TEC"           -b "tecsrvr48"      TEC

# Event groups are displayed in a vertical window.
# We have not duplicated the event sources but collected
# Alerts and Messages in one group and everything in another.
wcrteg     -b "genmainframe48"          NetView_390
waddegflt  -s "NV390ALT"                NetView_390
waddegflt  -s "NV390MSG"                NetView_390
wcrteg     -b "collection"              EveryThing
waddegflt                               EveryThing
wcrteg     -b "ADSM"                    StorageServ
waddegflt  -s "SENTRY"  -c "Sentry"     StorageServ
waddegflt  -s "LOGFILE" -c "ADSM_BASE"  StorageServ
wcrteg     -b "genapp48"                TopologyServ
waddegflt  -s "SENTRY"  -c "MSMAgent"   TopologyServ   6
wcrteg     -b "genapp48"                APM
waddegflt  -s "SENTRY"  -c "Sentry"     APM
set +x

##############################################################
# Prompt for the settings of the variables needed below.
# prevent by setting CreateUser to FALSE
until [ "$CreateUser" = "FALSE" ]
do
echo ' '
echo 'The next steps are to create a user named "GemAdmin"'
echo 'with complete capabilities on a UNIX system.  Terminate'
echo 'now if you do not want this user defined.  '
echo '  Press Enter to proceed or Ctrl-C to terminate.'
read UserReply
echo 'Creating GemAdmin as a UNIX user and a Tivoli administrator'
# the following prototype commands will create a UNIX userid and
# a Tivoli administrator with the ID of GemAdmin.  It will then
# assign the event groups to that id.

set -x
case $addr in
SunOS ] HP-UX)
useradd     -g 'staff' -G 'sys,adm,other' \
            -s '/bin/ksh' -d '/home/GemAdmin' GemAdmin;;
AIX)
mkuser      -a pgrp='staff' groups='system,staff,usr' \
            shell='/bin/ksh' home='/home/GemAdmin' GemAdmin;;
Windows_NT)
set +x
echo 'This is a Windows NT machine and you must create the GemAdmin'
echo '   user manually (in another window) and proceed or terminate'
echo '   the process. If you terminate you will have to create'
echo '   TME 10 administrator desktops and event consoles manuallly'
echo '   and assign the event groups to them.'
echo 'You should also stop and start your event server.'
```

```
echo ' '
echo '  Press Enter to proceed or Ctrl-C to terminate.'
set -x
read UserReply
;;
esac

wcrtadmin   -l GemAdmin -g staff -u GemAdmin -n "Enterprise Console" \
            -r "@EventServer,super:senior:admin:user:RIM_update:RIM_view" "GemAdmin"
wcrtconsole -l "GemAdmin" @GemAdmin
wassigneg   "@GemAdmin" NetView_390   senior admin user
wassigneg   "@GemAdmin" EveryThing    senior admin user
wassigneg   "@GemAdmin" StorageServ   senior admin user
set +x

CreateUser=FALSE
done
#################################################################
echo ' '
echo 'Stopping and restarting the Event Server'
echo '  Press Enter to proceed or Ctrl-C to terminate.'
read UserReply
set -x
wstopesvr
sleep 10
wstartesvr

set +x
```

Appendix C. Special Notices

This publication is intended to provide an introduction to TME 10 Global Enterprise Manger Topology, Event/Automation and Command Integration services and the TME 10 NetView for OS/390 Java Client. It describes how these functions should be installed, customized and examples of how they can be used. It can be used by customers and IBMers with an interest in network and systems management. The information in this publication is not intended as the specification of any programming interfaces that are provided by TME 10 Global Enterprise Manager. See the PUBLICATIONS section of the IBM Programming Announcement for TME 10 Global Enterprise Manager for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM′s product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

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, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

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

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX	APPN
AS/400	C/370
IBM	MVS/ESA
NetView	NetFinity
Nways	Operating System/2
OpenEdition	OS/390
OS/2	RACF
RS/6000	S/390
SP	System/390
VTAM	

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

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

Appendix D. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

D.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see "How to Get ITSO Redbooks" on page 199.

- *An Introduction to TME 10 NetView for OS/390*, SG24-4922

- *TME 10 Global Enterprise Manager Event/Automation and User Administration*, SG24-4921 (available December 1997)

- *Managing NetWare Environments from MVS Using NPM, MSM-NetWare*, SG24-4527

- *Managing IP Networks Using NetView MultiSystem Manager R2*, GG24-4337

- *Centralized Management of LNM and NetWare Networks Using NetView MultiSystem Manager MVS/ESA*, GG24-4181

D.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

D.3 Product Publications

- *TME 10 NetView for OS/390 Installation and Administration Guide*, SC31-8236

- *TME 10 NetView for OS/390 Command Reference*, SC31-8227

- *TME 10 NetView for OS/390 Data Model Reference*, SC31-8232

- *TME 10 NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*, SC31-8233

- *TME 10 Global Enterprise Manager Installation and User's Guide*, available with the product

- *TME 10 Global Enterprise Manager Topology Service User's Guide*, available with the product

- *Tivoli/Management Platform Documentation Kit*, SK2T-6058

- *Tivoli/Sentry Documentation Kit*, SK2T-6052

- *Tivoli Enterprise Console User's Guide*, GC31-8506

- *Tivoli Enterprise Console Adapters Guide*, SC31-8507
- *Tivoli Enterprise Console Rule Builder's Guide*, SC31-8508

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at `http://www.redbooks.ibm.com`.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type `GOPHER.WTSCPOK.ITSO.IBM.COM`
- **Tools disks**

    To get LIST3820s of redbooks, type one of the following commands:

    ```
    TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
    TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
    ```

    To get BookManager BOOKs of redbooks, type the following command:

    ```
    TOOLCAT REDBOOKS
    ```

    To get lists of redbooks, type one of the following commands:

    ```
    TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
    TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
    ```

    To register for information on workshops, residencies, and redbooks, type the following command:

    ```
    TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
    ```

    For a list of product area specialists in the ITSO: type the following command:

    ```
    TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
    ```

- **Redbooks Web Site on the World Wide Web**

    `http://w3.itso.ibm.com/redbooks`

- **IBM Direct Publications Catalog on the World Wide Web**

    `http://www.elink.ibmlink.ibm.com/pbl/pbl`

    IBM employees may obtain LIST3820s of redbooks from this page.

- **REDBOOKS category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

    With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to `announce@webster.ibmlink.ibm.com` with the keyword `subscribe` in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

Redpieces

For information so current it is still in the process of being written, look at ″Redpieces″ on the Redbooks Web Site (`http://www.redbooks.ibm.com/redpieces.htm`). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** — send orders to:

	IBMMAIL	Internet
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU

Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1)001-408-256-5422 (Outside USA)** — ask for:

  Index # 4421 Abstracts of new redbooks
  Index # 4422 IBM redbooks
  Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com

- **On the World Wide Web**

Redbooks Web Site	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Internet Listserver**

  With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (http://www.redbooks.ibm.com/redpieces.htm). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

First name _____ Last name _____

Company _____

Address _____

City _____ Postal code _____ Country _____

Telephone number _____ Telefax number _____ VAT number _____

- Invoice to customer number _____

- Credit card number _____

Credit card expiration date _____ Card issued to _____ Signature _____

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

List of Abbreviations

AIX	Advanced Interactive eXecutive		**MSM**	MultiSystem Manager
APF	Authorized Programming Facility		**NCCF**	Network Control Command Facility
AMS	Application Management Specification		**NGMF**	NetView Graphical Monitor Facility
APM	Application Policy Manager		**NPDA**	Network Problem Determination Application
CDS	Configuration Data Set		**OMVS**	OpenEdition for MVS
CNM	Communications Network Management		**OS/2**	Operating System/2
			PPI	Program to Program Interface
CSD	Corrective Service Diskette		**PSP**	Preventive Service Planning
DCE	Distributed Computing Environment		**PTF**	Program Temporary Fix
EIF	Event Integration Facility		**RACF**	Resource Access Control Facility
GMT	Greenwich Mean Time		**RDBMS**	Relational Data Base Management System
EAS	Event/Automation Service			
FMID	Function Management Identifier		**RODM**	Resource Object Data Manager
FTP	File Transfer Program		**SMP/E**	System Modification Program/Extended
GA	General Availability			
GEM	Global Enterprise Manager		**SNA**	System Network Architecture
GMFHS	Graphical Monitor Facility Host Subsystem		**SQL**	Structured Query Language
			TCP	Transmission Control Program
GTF	Generalized Trace Facility			
IBM	International Business Machines Corporation		**TPS**	TP Server
			T/EC	Tivoli/Enterprise Console
IP	Internet Protocol		**TEC**	Tivoli/Enterprise Console
ITSO	International Technical Support Organization		**TME**	Tivoli Management Environment
JCL	Job Control Language		**TMR**	Tivoli Management Region
LA	Limited Availability		**TSO**	Time Sharing Option
LCF	Lightweight Client Framework		**WTO**	Write To Operator

Index

ITSO Redbook Evaluation

TME 10 Global Enterprise Manager Topology Service and NetView 3270 Java Client
SG24-2121-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at http://www.redbooks.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@vnet.ibm.com

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs?   Yes____ No____

If no, please explain:

_____

_____

_____

_____

What other redbooks would you like to see published?

_____

_____

_____

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

_____

_____

_____

_____

_____

IBM ®

Printed in U.S.A.