HP OpenView Console User's Guide

HP 3000 MPE/iX Computer Systems

Edition 2

Manufacturing Part Number: B3118-90011
E0895

U.S.A. August 1995

Notice

The information contained in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for direct, indirect, special, incidental or consequential damages in connection with the furnishing or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013. Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19 (c) (1,2).

Acknowledgments

UNIX is a registered trademark of The Open Group.

Hewlett-Packard Company
3000 Hanover Street
Palo Alto, CA 94304 U.S.A.

Contents

Contents

Contents

Figures

Tables

Tables

Preface

Read this manual if you are responsible for the day-to-day operation of an HP 3000 system managed by an OpenView Console. You should have knowledge of HP 3000 systems and utilities of the MPE/iX operating system, including the commands required for operation.

Since HP OpenView Console (OVC delivers its functionality through Windows, you should be familiar with the Windows user interface. The operating system for your PC is MS-DOS so you should also be familiar with DOS concepts and utilities. For more information, refer to the manuals provided by Microsoft.

It is also important that you are familiar with PC-related hardware issues. For more information about PC concepts and troubleshooting, refer to the user's guide for your PC.

1 Introucton to Using HP OpenView Console

This chapter introduces the basic concepts and features of HP OpenView Console (OVC).

Overview

HP OpenView Console (referred to as OVC throughout this guide) is an HP application that provides a single integrated system management solution. Running on the MS Windows platform, HP OpenView provides a graphical user interface (GUI) that displays a *map* with network and system components represented by symbol icons. These symbols change color to reflect the state of the components being managed. Also OVC has unified command menu items, which are object specific, and invoke the registered OpenView applications. The OVC program is an HP OpenView application.

OVC is a subset of the HP OpenView System Manager product (referred to as SysMgr). OVC provides the GUI for a local console on a *single* MPE/iX system. Using SysMgr, you can manage, monitor, and control *multiple* HP 3000 systems from one central console in both Local Area Networks (LANs) and Wide Area Networks (WANs). You can easily upgrade from OVC to SysMgr if your system requirements change. (If you upgrade, you must obtain additional managed node software licenses.) For more information on how to upgrade to SysMgr, refer to the *HP OpenView Console Manager's Guide*.

The OVC solution integrates a number of features that provide you with the following capabilities:

- You can completely monitor and control an HP 3000 from your desk as a single, consolidated operation.

  This product allows you to consolidate operations for managing an HP 3000 system on a network. You have greater flexibility in defining the responsibilities of system control. Since OVC provides facilities for console consolidation, the problem of varying levels of staff expertise in your local operation is alleviated.

- You can *manage your system by exception*, meaning you are notified only when problems occur.

  You increase productivity when you manage your system by exception. This means that you are only notified when a problem occurs that requires operator action. OVC provides a mechanism to define important events and filter out the non-essential ones. This frees you from constantly watching a console, looking for problems. Once a problem is identified you can access the console. You only need to access the console to take a specific action.

OVC is not a single program running on a single HP 3000 or PC, but a number of programs that have specific functions. The integrated system uses the following components:

OVC PC Software — Software resident on the PC providing a Microsoft Windows GUI. You can have multiple PCs (up to 10) running the OVC PC software to communicate with the HP 3000 system. However, each PC requires a valid OpenView software license.

OVC HP 3000 Software — OVC HP 3000 software provides the ability to collect console events, perform automated responses, and deliver event messages to the OVC PC.

OVC uses a client/server design that involves many communication technologies. After you fully configure OVC, you can perform daily operations unaware of any complexity. This is due to the simple design of the HP OpenView interface. All OVC PC programs are fully integrated under the HP OpenView platform with the latest Visual OpenView technology.

OVC Features

When you configure OVC correctly, you can monitor, control, and diagnose any problem your HP 3000 may encounter. The OVC HP 3000 software monitors console events automatically. If a problem occurs, you are notified so that you can locate, analyze, and solve problems. The OVC features include:

System Event Notification

Agent software on the HP 3000 scans console messages for events. The OVC HP 3000 software collects and sends the event messages to the OVC PC. You are notified of the arrival of these events when the color of the objects on the OpenView map changes.

Event Message Review

When an event message is saved in the event message database on the HP 3000, you can review it from the OVC PC. By invoking the Multiple Document Interface (MDI) event browser, you can review the outstanding unacknowledged events. Using this message review process, you can annotate, respond to, acknowledge (remove), and export an event message.

Central Event Repository

All system events are stored in the Event Message Log (EML) database on the HP 3000. You can use the EML report generation feature to produce a report of the event messages, even if these messages are acknowledged (removed) from the event browser.

Monitor Console

With this feature you can monitor the output of the OVC console from a terminal session at the HP 3000. You can use the OVC NTC utility at a terminal session to display the console messages.

Control Console

This feature provides console access to the HP 3000 from the OVC PC via a physical connection to the HP 3000's console port. You can use a dial-up modem connection or a Data Communications and Terminal Controller (DTC) in a Local Area Network (LAN) environment.

Management Session Interface (MSI)

This feature provides control of normal HP 3000

operations while the system is running. From the OVC PC, the Management Session Interface (MSI) provides a special Virtual Terminal (VT) session for the HP 3000. During an MSI session you can execute commands that were previously restricted only to the system console. The MSI technology uses the existing network so no additional hardware is required. Block Mode programs can be executed via MSI.

Automated Response Service (ARS)

OVC's automated response service allows your data center to automate procedures by monitoring console messages and executing MPE Command Interpreter (CI) scripts without user intervention. As a customizable tool, you can configure ARS to execute any response.

Integrated OpenView Interface

All OVC PC programs are integrated with the HP OpenView/DOS platform. You can also integrate your network management tools with the same OpenView map. If you have the Workgroup Node Manager version of OpenView, you can link beeper notification with the OVC event posting function.

Task-Based Operations

OVC allows you to customize your network map to show only the system events that apply to an operator's job or task. This feature is especially useful in a data center environment where different operators are assigned to perform different tasks managing the system.

The OVC PC User Interface

Before starting an OVC session on your HP OpenView Console PC, you need to take certain steps. To start the OVC HP 3000 software or to install software on the HP 3000, refer to the first chapters of the *HP OpenView Console Manager's Guide*.

To start an OVC session, you should check that:

- The OVC HP 3000 software is installed and activated.
- All the required network and application software is installed and activated on your OVC PC.
- Your OpenView map is created and specifically tailored for your job assignment. (To learn how to assign a default map to your logon, or to modify your current map, refer to the *HP OpenView for Windows User's Guide* for more information.)

NOTE OVC is preconfigured with a user logon (MGR) and a password (HPSYSMGR). However, you can change the logon and password and add more users and passwords later.

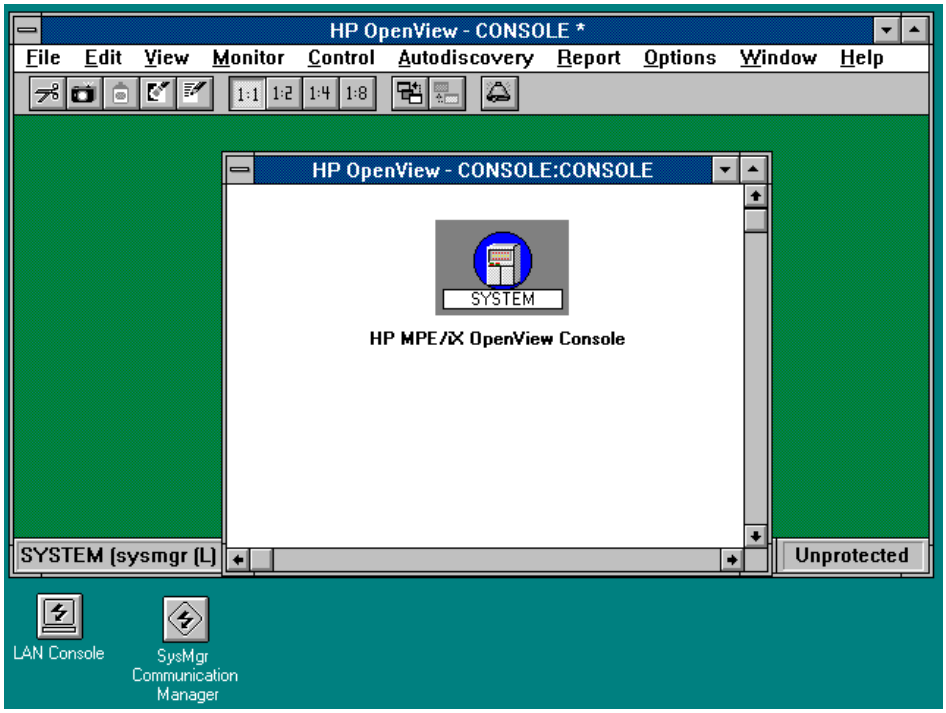
Starting HP OpenView

After completing the required installation and configuration, start HP OpenView:

- Double-click on the HP OpenView program icon.

  The OpenView logo is displayed. It takes some time to load the program. Then the OpenView map window is displayed along with two OVC program icons that you will use: Local Area Network (LAN) Console (LANCON) and SysMgr Communication Manager (COMMMGR), see Figure 1-1.

Figure 1-1 HP OpenView Window with OVC Program Icons



- LANCON coordinates the use of the COM port on the PC to provide console access.
- COMMMGR provides the network connection between the OVC PC and the HP 3000.

Both programs are in the C:\OV\SYSMGR directory. If one of the programs fails, you can start the program using the **Run** item from the Windows Program Manager window without restarting OpenView.

The first task for the COMMMGR program is to make a network connection with the HP 3000 using the NOCAP default user (more about NOCAP later). With this newly created link, COMMMGR requests the server at the HP 3000 to report the current state of all the

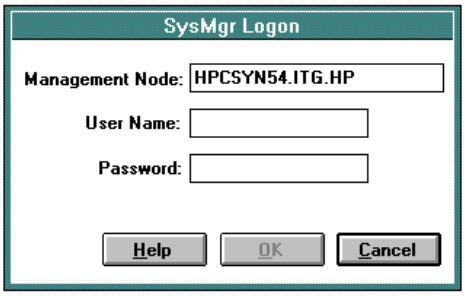
known map symbols. A status window is displayed. The Windows cursor changes to an hourglass (the *wait cursor*). All mouse and keyboard activity is suspended until the arrow cursor is displayed. Then each map symbol is displayed with one of these event colors indicating a specific condition.

- Red — a critical condition
- Yellow — a warning condition
- Magenta — an informational message
- Green — a normal condition
- Blue — an unknown condition (COMMMGR never received a color update for the symbol)

Logging On

The first time you select an OVC command (for example, the `Configure Mgmt Node...` submenu from the Customize HP SysMgr menu under the HP OpenView `Options` menu), the `SysMgr Logon` dialog box is displayed (see Figure 1-2).

Figure 1-2 SysMgr Logon Dialog Box



Follow these steps to log on:

1. Enter the name of the HP 3000 in the `Management Node:` text box.

   If your map symbols are receiving color update information, your current HP 3000 name is already displayed in the `Management Node:` text box. But if all your map symbols are blue, most likely you will see an empty text box. Without knowing what system is the designated HP 3000, the COMMMGR program cannot create the NOCAP user link to get the symbol color. Therefore, you must enter the HP 3000 name manually. Once a network link is created, this node name is remembered automatically.

   The NOCAP user is a special entry defined in the OVC user configuration database. The PC uses the NOCAP user to create a network link so that map symbols are updated although no real user logon exists. Once a real user logs on, the OVC HP 3000 drops the NOCAP link, and creates a new link for the PC that has the newly logged on user.

2. Use the **Tab** key to move the cursor to the `User Name:` text box (or click in the box with the mouse), and enter your assigned OVC user name.

3. Use the **Tab** key to move the cursor to the `Password:` text box (or click in the box with the mouse), and enter your password.

4. Either click on the **OK** button at the bottom of the dialog box, or press the **Enter** key to log on to the HP 3000.

   If you logged on successfully, the command that you selected is activated, and the associated window is displayed. At this point, you

have access to any functions that are configured for your OVC user name. Your OVC administrator has authority to change your OVC capabilities. For more information on these capabilities, refer to the *HP OpenView Console Manager's Guide.*

Follow this step to log off from your session with OVC:

1. From the `Control` menu, click on the `SysMgr Logoff` option.

   When you select this menu item, the COMMMGR program closes out all the links the HP 3000 associated with your logon, and creates a NOCAP link. Initially you see all the symbols on your map change to blue, but in a second all the symbols' colors are updated to their original color.

   Your HP OpenView map remains displayed on the OVC PC.

Where to Go from Here

If you successfully logged on for your first OVC session, your network is working properly. You are now ready to use OVC. First read Chapter 2 , “OVC Features,” which describes all the menu items you can use during an OVC session.

After reading Chapter 2 , “OVC Features,” you should have enough information to begin exploring OVC. Chapter 3 , “Using OVC: A Typical Scenario,” outlines how a data center typically uses OVC for monitoring, controlling, and maintaining its HP 3000 system. This should give you an idea of how to use OVC.

Chapter 4 , “Generating Reports.” describes how you create reports of event messages.

Chapter 5 , “Using Automated Response Service (ARS),” describes how you use the Automated Response Service to respond to system events.

The Appendices at the back of the guide explain the error messages for the OVC PC and the log file.

2 OVC Features

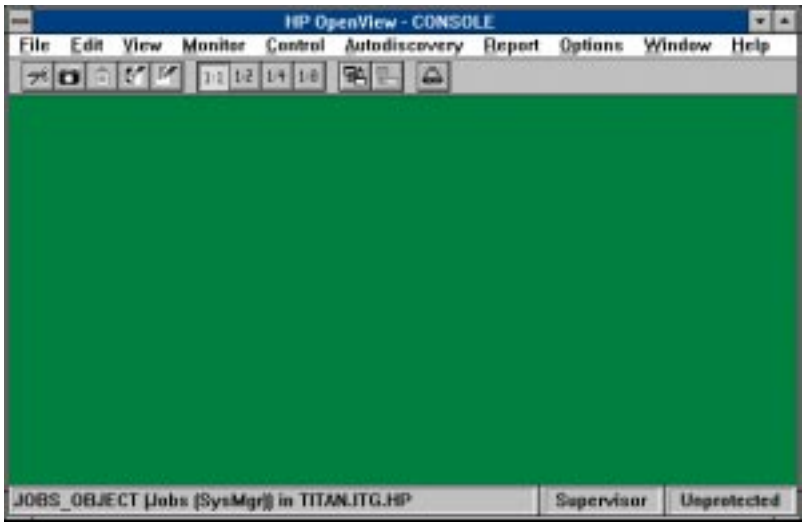
This chapter describes the OVC features that you can use to control the HP 3000 system from your PC.

Overview

OVC has a Windows graphical user interface so you can manipulate objects either directly (while designing your network map, for example), or by clicking on items, such as menus or buttons. When you click on one of the menus, you can select options that get system information or perform system operations during an OVC session. The OVC menus are the standard Windows type with alternate key combinations so that you do not have to use the mouse.

Across the top of the main HP OpenView window is a menu bar with pull-down menu options that you use to do tasks on your system (see Figure 2-1).

Figure 2-1 HP OpenView Menu Bar

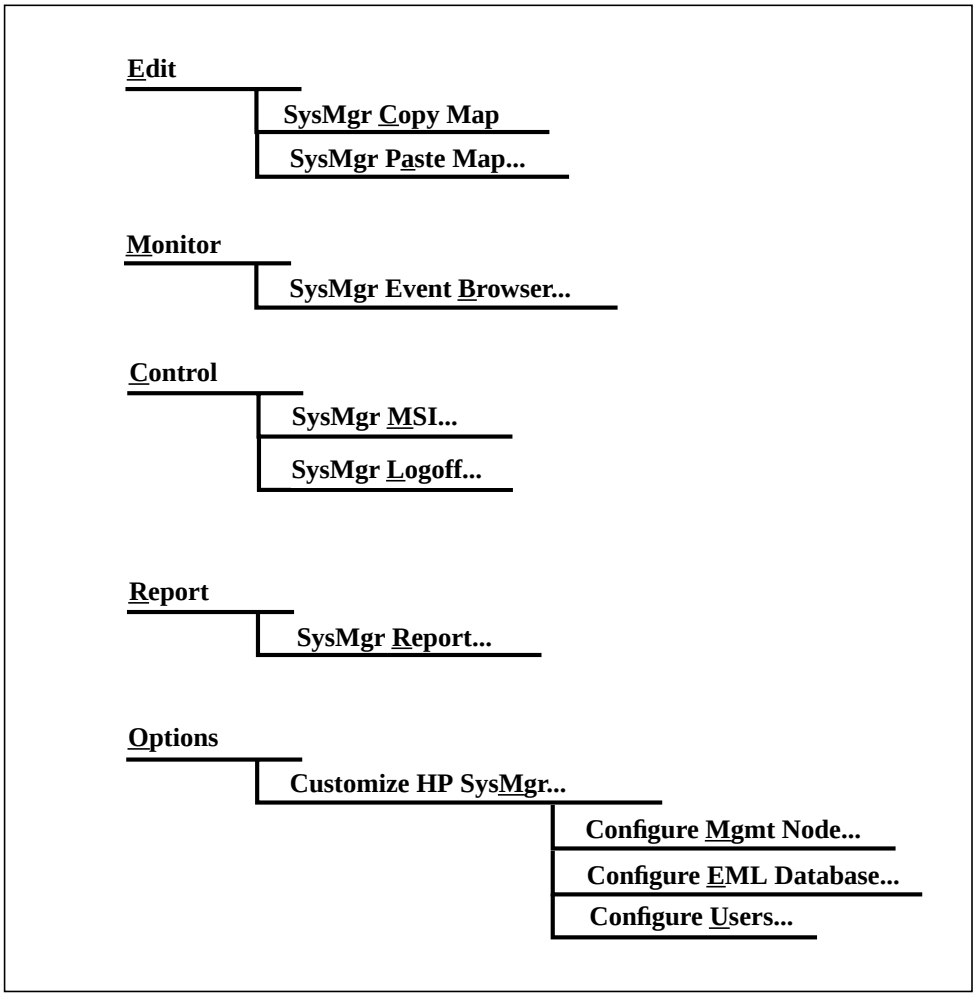


Menu Items of OVC

You do not need all the menu items associated with HP OpenView to use OVC. Figure 2-2 lists the menu items that you can use with OVC.

NOTE Since OVC is a subset of the full SysMgr system management solution, all the menu items are preceded by the name SysMgr.

Figure 2-2 OVC Menu Items



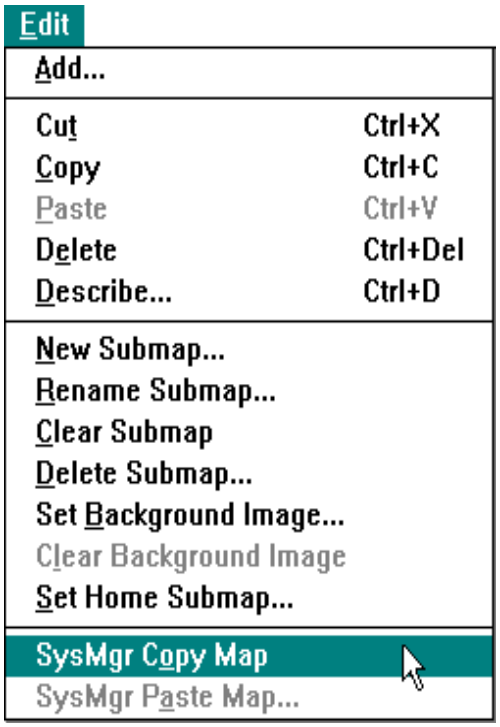
When you click on one of the OVC menu items in the HP OpenView main window, a window or dialog box is displayed so that you can do certain tasks.

SysMgr Copy Map

You use the SysMgr Copy Map menu item in the Edit menu to copy all the contents or just selected objects in a submap (see Figure 2-3).

NOTE With the HP OpenView Copy menu item (which is also found in the Edit menu), you can copy only one map symbol.

Figure 2-3 HP OpenView Edit Menu



To copy an entire submap, bring the submap window to the top by clicking on it. Make sure none of the objects in the submap are selected. When you click on the SysMgr Copy Map menu item in the Edit menu, all the map object symbols (except lines and text) are copied to local memory (not the clipboard). Then using the SysMgr Paste Map menu item, you can paste these objects to another submap.

If you just want to copy just a selected set of objects, select the objects by clicking each one while holding down the **Shift** key (or the **Ctrl** key). Then click on the SysMgr Copy Map... menu item in the Edit menu to copy your selected map symbols.

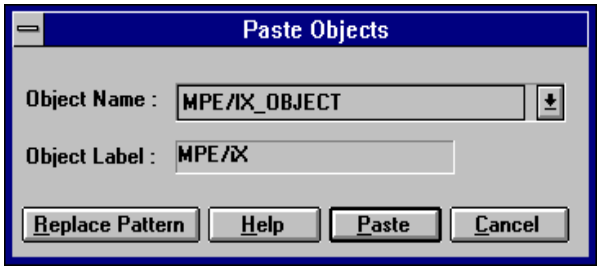
NOTE The locations of the objects are also copied so when the objects are later pasted to a new submap, they will be located in the same position as on the previous submap.

SysMgr Paste Map

You use the SysMgr Paste Map... menu item in the Edit menu to paste copied objects onto a submap. Before using this menu item, you must first use SysMgr Copy Map... to copy objects.

To paste the copied objects onto a new submap, bring the target submap to the top by clicking on it, and then click on the SysMgr Paste Map... menu item. The Paste Objects dialog box is displayed (see Figure 2-4).

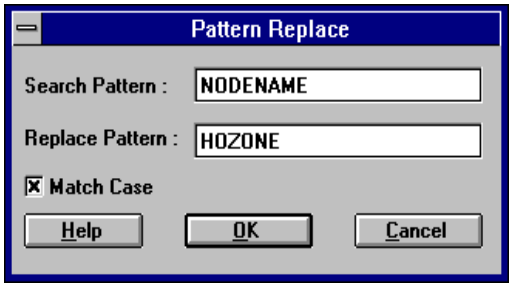
Figure 2-4 Paste Objects Dialog Box



You use the Paste Objects dialog box to verify the objects that were copied. You can also use this dialog box to perform global pattern replacement on the object names of the copied objects by clicking on the **Replace Pattern** button. Otherwise click on the **Paste** button, and the objects are pasted to the submap.

When you click on the **Replace Pattern** button, the Pattern Replace dialog box is displayed (see Figure 2-5).

Figure 2-5 Pattern Replace Dialog Box



You use the Pattern Replace dialog box to replace the pattern on object names. After you replace the pattern, click the **Paste** button in the Paste Objects dialog box to paste the objects to your new submap.

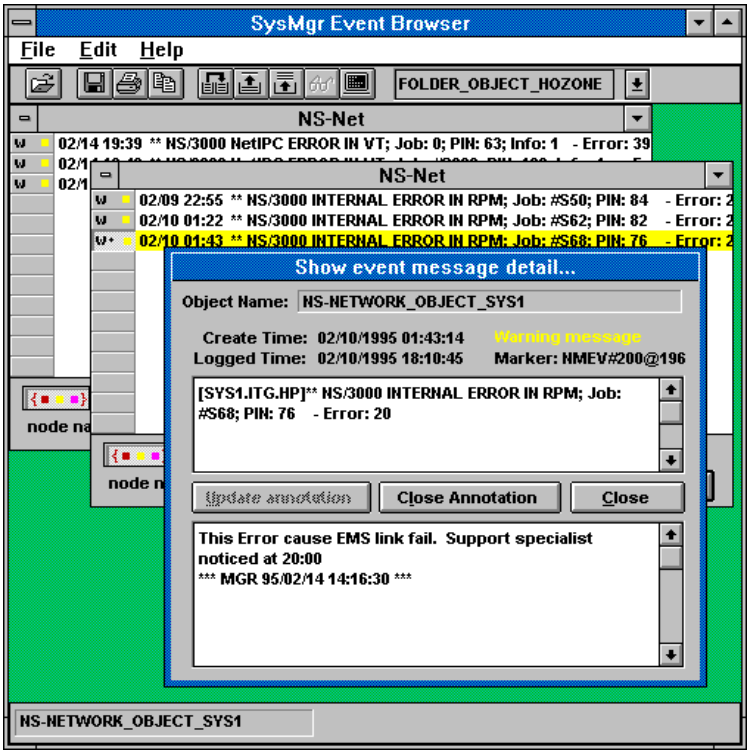
NOTE If you check the Match Case option, the Search Pattern is case sensitive. If you do not check the Match Case option, the Search Pattern is not case sensitive.

SysMgr Event Browser

You use the SysMgr Event Browser menu item in the Monitor menu to display outstanding events associated with the map symbols. In the dialog boxes associated with the SysMgr Event Browser, you can view, acknowledge (remove), or move events to a folder. Before clicking on this menu item, you must select an OVC map symbol. If you do not click on a map symbol first, only the SysMgr Event Browser dialog box is displayed. You can also click on the Open menu item from the File menu in the SysMgr Event Browser dialog box to create the event window for any known OVC object.

Figure 2-6 shows the SysMgr Event Browser dialog box.

Figure 2-6 SysMgr Event Browser Dialog Box with Detailed Message and Annotation



Each map symbol has its own event window which displays the outstanding events. You can click on the color buttons at the bottom of the window to select the severity of events you want displayed. Since the event window shows only the first 70 characters of an event message, you can display a detailed information window to see the entire message. You can invoke this window in three different ways:

- Double-click on the leading buttons at the left side of the message.
- Select the event, then click on the *glasses* button on the tool bar.

- Select the event, then pull down the `Edit` menu to choose the `Show Event Detail` menu item.

The detailed information window is shared by all the event windows, and its content is updated automatically when you select a new message. Each event window uses approximately 3% of the PC's resources, so do not open too many event windows at the same time.

The only way you can clear the symbol color is to acknowledge (remove) the event from the display. So if you want to keep the event for further investigation, move it into an event folder (a temporary holding place). You can add as many folder symbols as you want to your map.

CAUTION Folders do not provide long-term event storage. The folder symbol behaves as an ordinary object symbol. When the Event Message Log (EML) database containing the event expires, the event is removed automatically from the folder. To keep an event longer than the life of its related database, save it to a disk file or export it to a trouble-tracking system by using the Windows clipboard.

Click on the `Help` menu at the top of the `SysMgr Event Browser` dialog box to learn more about the SysMgr Event Browser features and uses.

SysMgr Logoff

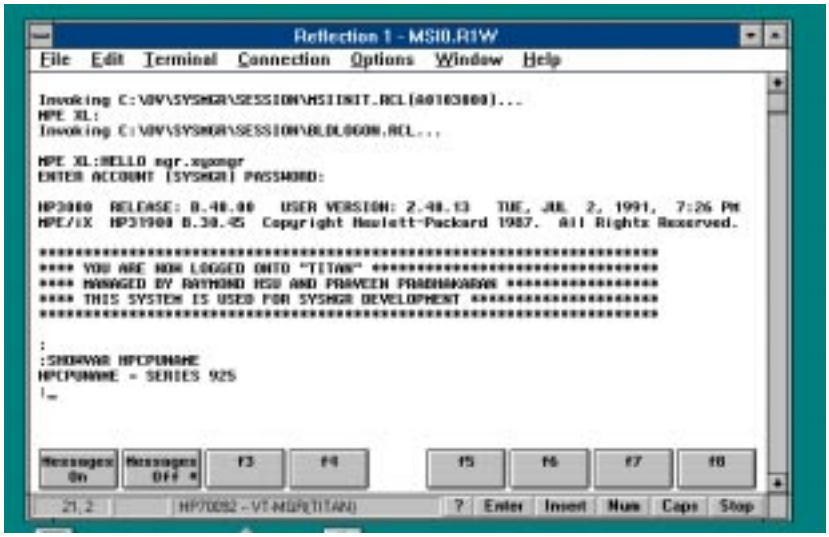
You use the `SysMgr Logoff` menu item in the `Control` menu to log off from OVC without exiting HP OpenView. The SysMgr COMMMGR program enables this menu item when you log on to SysMgr successfully. When you click on this menu item, COMMMGR closes all the currently operating OVC applications and returns the status colors of the map symbols to blue. Then COMMMGR creates the NOCAP user link with the HP 3000 to update the color status of all map symbols. This operation may take a few seconds. A status window may be displayed to indicate what is happening. User input is temporarily suspended.

SysMgr MSI

You use the `SysMgr MSI...` menu item in the `Control` menu to start the Management Session Interface (MSI). With MSI you can issue operator commands that control the HP 3000. Before clicking on this menu item, you must first select a computer symbol from your HP OpenView map. When you click on this menu item, a terminal emulator window is displayed, and a terminal script is executed automatically.

The first time you start `SysMgr MSI` you are prompted to specify the HP 3000 logon information needed to establish the VT session with the HP 3000. This logon information will be stored and used whenever you start MSI again. At the end of the script execution you see the colon (:) prompt (see Figure 2-7).

Figure 2-7 SysMgr MSI Terminal Window



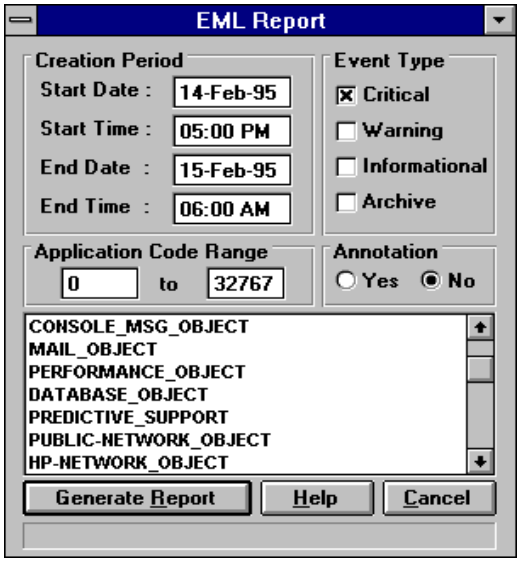
By running the NETALLOW.EMS.SYSMGR program you gain special user capabilities. Once you run this program, you can use the MPE commands which ordinarily are restricted to the local console port (ldev-20) logon user. Modify your personal HP 3000's logon User Defined Command (UDC) to include this program's execution for future connections.

You can also monitor console output and detailed automated response execution from this interface. Two function keys (**Message On** and **Message Off**) are enabled so that you can enter or exit the console message monitoring mode.

SysMgr Report

You use the SysMgr Report... menu item in the Report menu to create a report of events in the EML (Event Message Log) database. When you click on this menu item, the EML Report dialog box is displayed (see Figure 2-8).

Figure 2-8 EML Report Dialog Box



Typically you use this menu item when you want a hard copy of all events that occurred over a period of time. You can generate a report containing events from the EML databases in the current archive range.

NOTE The report includes events that are not in the SysMgr Event Browser since acknowledging (removing) events from the SysMgr Event Browser does not remove them from the EML database.

Generally a report is generated for week-end auditing, or to ensure that you still have access to event data even after EML databases go out of archive range.

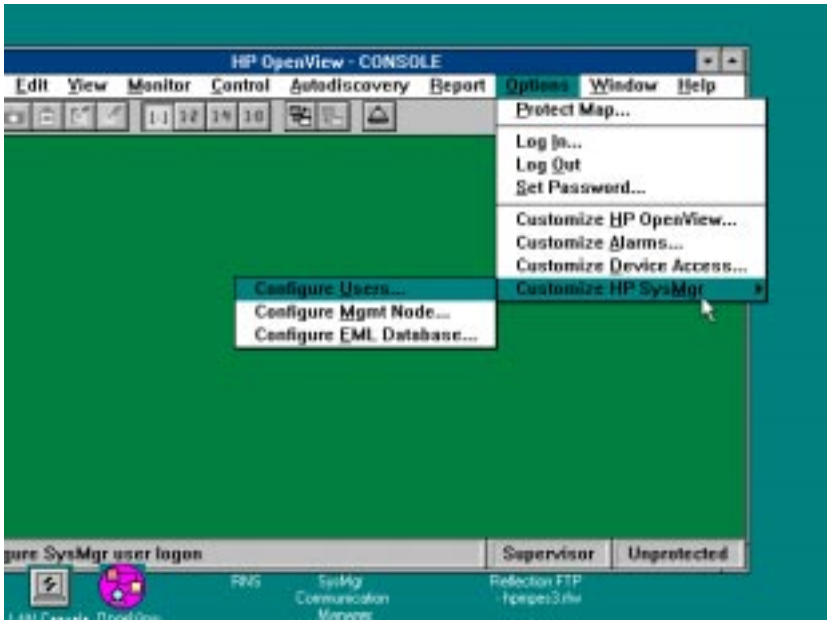
You can create a report either on the PC or the HP 3000. You can also redirect the report to the printer on the HP 3000. You can use report templates to produce reports in a specified format, defined at your site. For more information about creating report templates, refer to Chapter 4 , “Generating Reports.”

Click on the **Help** button in the EML Report dialog box to learn more about the SysMgr Report features and uses.

Customize HP SysMgr

You use the `Customize HP SysMgr...` menu item in the `Options` menu to customize your OVC setup. When you click on this menu item, a submenu of menu items is displayed (see Figure 2-9).

Figure 2-9 Customize HP SysMgr Submenu Items

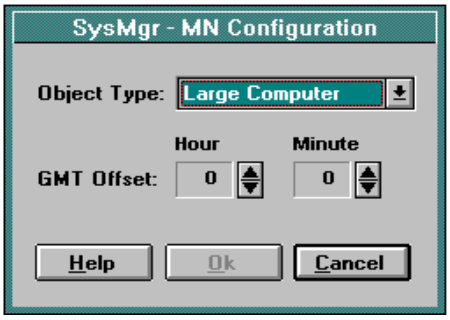


Most of these menu items require special OVC capabilities available only to the OVC administrator. Each menu item displays a dialog box with on-line help on how to use the command.

Configure Management Node

You use the `Configure Mgmt Node...` menu item to configure the HP 3000. Clicking on this menu item opens the `SysMgr - MN Configuration` dialog box (see Figure 2-10).

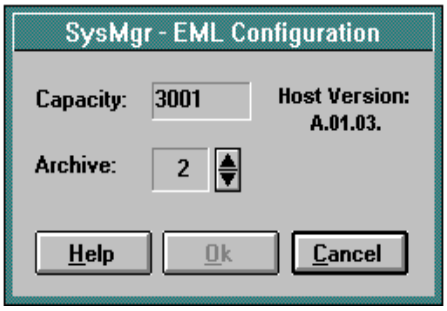
Figure 2-10 SysMgr - MN Configuration Dialog Box



Configure EML Database

You use the `Configure EML Database...` menu item to configure capacity and archive parameters. Clicking on this menu item opens the `SysMgr - EML Configuration` dialog box (see Figure 2-11).

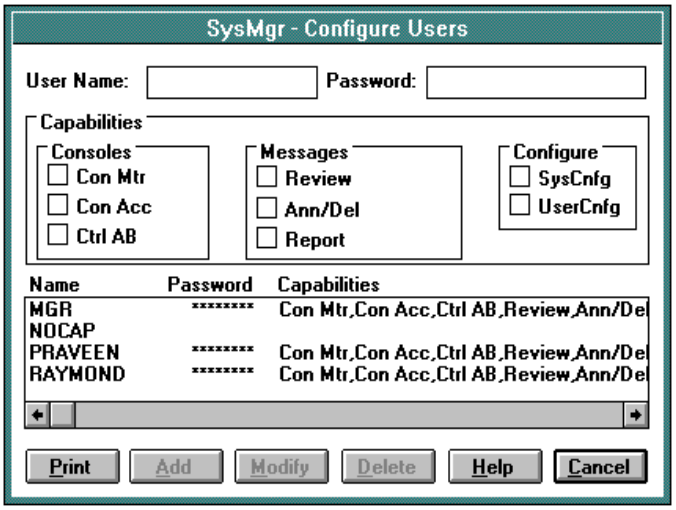
Figure 2-11 SysMgr - EML Configuration Dialog Box



Configure Users

You use the `Configure Users...` menu item to configure capabilities for each user. Clicking on this menu item opens the `SysMgr - Configure Users` dialog box (see Figure 2-12).

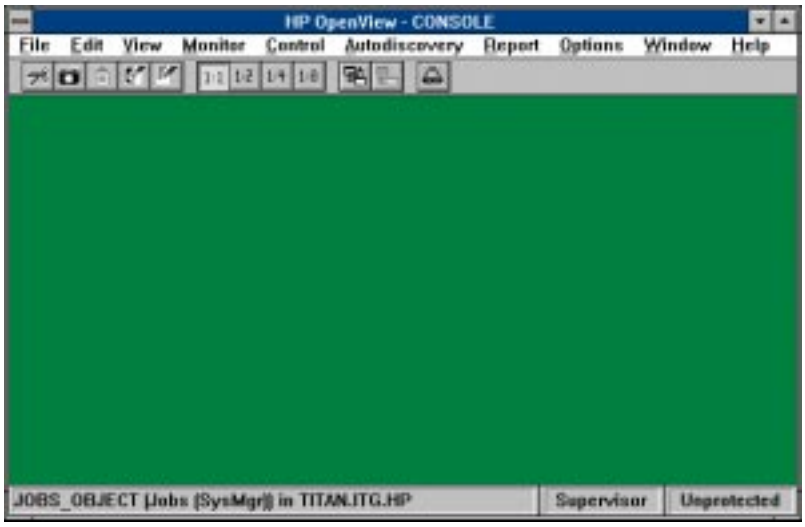
Figure 2-12 SysMgr - Configure Users Dialog Box



Using On-Line Help

To get information about HP OpenView, click on the HP OpenView `Help` menu item located on the menu bar of the HP OpenView main window (see Figure 2-13). An index of information about HP OpenView commands is displayed.

Figure 2-13 HP OpenView Help Menu Item

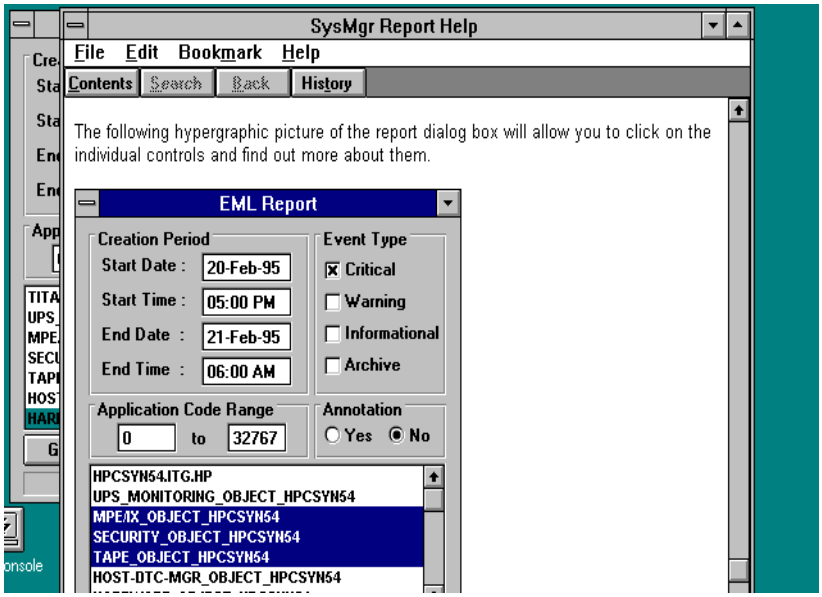


To get information about OVC, you can access OVC Help two ways:

- Most OVC dialog boxes have a **Help** button. By clicking on the **Help** button, you access context sensitive on-line help about that command.
- Some OVC dialog boxes contain Help as a menu name. By clicking on the `Help` menu name in the menu bar, you can access on-line help about that dialog box.

For example, if you select the `SysMgr Report` menu item from the `Report` menu, you see a **Help** button at the bottom of the `EML Report` dialog box. When you click on the **Help** button, the `SysMgr Report Help` dialog box is displayed with on-line help for the EML Report (see Figure 2-14).

Figure 2-14 SysMgr Report Help Dialog Box



The `SysMgr Report Help` dialog box contains information and appropriate *hypergraphic pictures* for using the dialog box. When you click on an item in the picture that you need information on, a pop-up window displays the required information for that item.

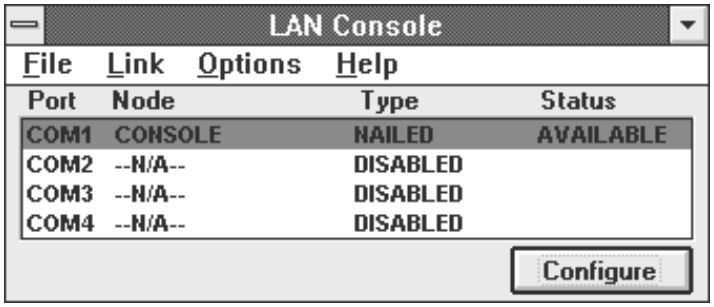
Some OVC Help dialog boxes also contain underlined *hypertext* which is linked (associated) to additional information in another Help dialog box. When you click on the hypertext, the information linked to the topic is displayed.

On-line help gives you more details about using OVC. This chapter describes all the OVC features, but you are encouraged to use on-line help to learn more about the OVC PC components.

LAN Console

LAN Console is the OVC application that you use to initiate a console connection to your HP 3000. When you install the OVC PC software, LAN Console is installed in the StartUp program group of the Windows Program Manager. When you start Windows, LAN Console starts up. By default, LAN Console is configured to use the COM1 port to initiate a console connection while all other ports are DISABLED (see Figure 2-15).

Figure 2-15 LAN Console Configured to Use COM1 Port



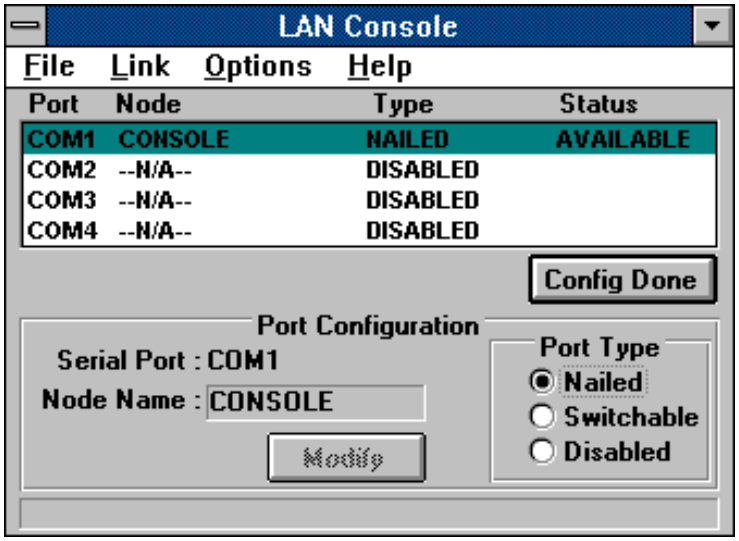
The OVC PC installation program also sets the LaunchNailedConsole entry in the `[Console]` section in the `c:\ov\sysmgr\sysmgr.ini` file to Yes. When LAN Console starts up, it reads this entry. If set to Yes, LAN Console starts console connections to all COM ports with NAILED connections. If you do not want the Reflection console window to open when you start Windows, set the LaunchNailedConsole entry to No.

If necessary, you can also configure LAN Console to use the remaining COM ports on your PC to connect to console ports on other HP 3000 systems. With LAN Console you can configure COM port console connections in three different ways:

- NAILED — specifies that the COM port is physically hardwired to connect to the console port of the HP 3000.
- SWITCHABLE — specifies that the COM port is connected to a device capable of switching connections (for example, modem or DTC).
- DISABLED — specifies that the COM port is not present or is reserved for another application.

To configure a COM port, click on the `Configure` button in the `LAN Console` dialog box. The Port Configuration frame is displayed at the bottom of the `LAN Console` dialog box (see Figure 2-16).

Figure 2-16 LAN Console Port Configuration Frame



To learn more about configuring and using LAN Console, click on the `Help` menu item in the `LAN Console` dialog box.

When you complete configuring your ports, click on the `Config Done` button. Then click on the `Save Settings F2` menu item from the `Edit` menu to save your settings.

NOTE

In SysMgr, the node name field identifies the HP 3000 to which the COM port is connected. In fact the user has to configure these entries and when the configuration is completed, appropriate script files (with file names identical to the node name) are created containing the connection information. In OVC, a dummy node name, “CONSOLE,” is preconfigured to use COM1 to connect to the HP 3000 console port. The user should avoid changing this COM1 configuration. If it is changed, then the console connection script must be rebuilt during the next console connection. For more information on console connection scripts, refer to Appendix A , “Console Automation Scripts,” in this guide.

Exiting OVC

You use the `Exit` menu item in the `File` menu to end your OVC session on the OVC PC. When you click on this menu item, a dialog box is displayed indicating that your OpenView session will end. Click on the **OK** button.

However, although you ended the OVC session, OVC is still running on the HP 3000. To terminate OVC on the HP 3000 system, refer to the *HP OpenView Console Manager's Guide*.

3 Using OVC: A Typical Scenario

This chapter shows an example of how you might use OVC for day-to-day operations.

Overview

This chapter briefly describes a scenario showing how an operator uses OVC. Hopefully, you will find the information useful in planning, organizing, and using the OVC tools. Also the information will help you understand how you can control and solve any problem you may encounter with the HP 3000 you manage from your desk. The scenario does not include details about configuring or setting up OVC. For more detailed information on those topics, refer to the *HP OpenView Console Manager's Guide*.

The scenario suggests a very likely series of tasks you might undertake, step by step, after finding a serious event on your HP 3000. Also included are suggestions for the steps you might use to locate and solve a problem on your HP 3000 system.

A Typical Scenario

Bill manages the HP 3000 system located in the Production Department. He uses OVC to manage the HP 3000 from his office. Bill's HP 3000 system is named TITAN and is on a LAN using Ethernet. Bill has ensured control console access to the HP 3000 from his OVC PC by connecting the OVC PC's COM1 port to the ldev-20 console port of the HP 3000.

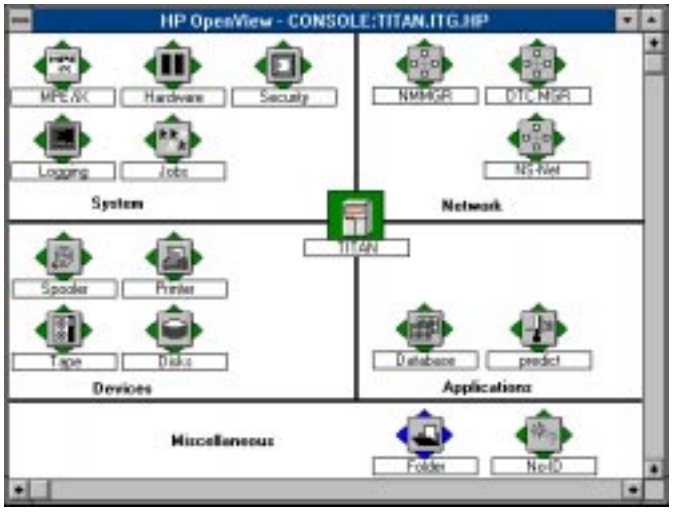
How Bill Organizes His Work

Bill's top-level HP OpenView map has a system symbol representing the HP 3000 (see Figure 3-1).

Figure 3-1 Bill's Top-Level Map

The HP 3000 system symbol explodes into a map of tasks (see Figure 3-2).

Figure 3-2 Bill's Submap of Tasks for the HP 3000



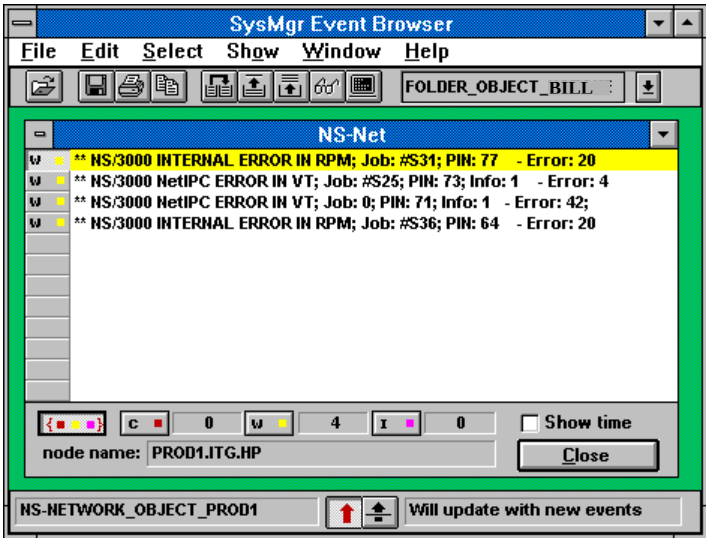
How Bill Responds to Events

Bill is primarily responsible for the Production Department system and monitors the HP 3000 map. The HP 3000 system symbol is green when there are no events against it. If the HP 3000 receives a new non-critical event, the symbol color changes, indicating an informational event (magenta), or a warning event (yellow). Bill double-clicks on the HP 3000 system submap symbol on the system map. Then when the task map opens, he clicks on the `Show SysMgr Events...` option from the pop-up menu by right clicking on the task symbol that changed color (see Figure 3-3).

Figure 3-3 Pop-Up Menu with Show SysMgr Events... Menu Item

Bill reads through the events listed in the `SysMgr Event Browser` to locate the event causing the color change (see Figure 3-4).

Figure 3-4 SysMgr Event Browser

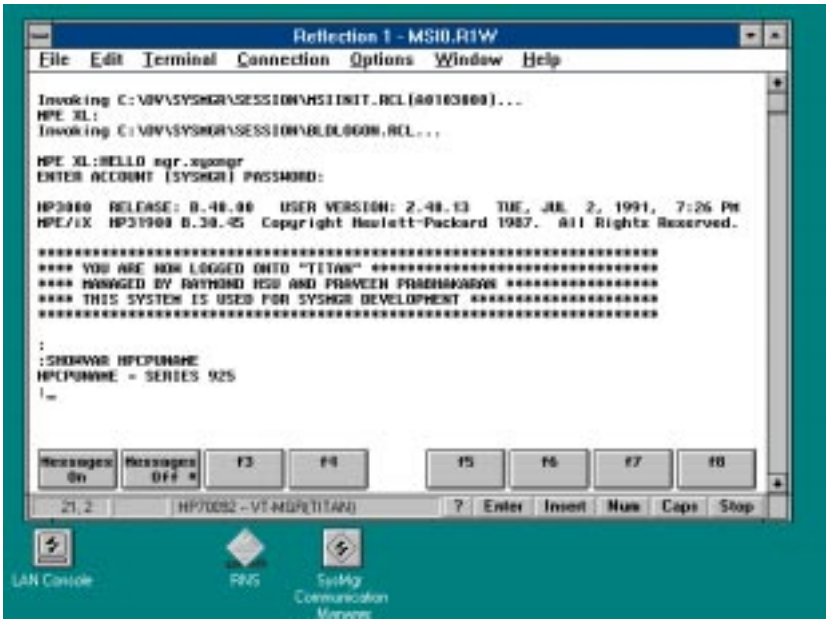


If the event does not require attention, Bill deletes the event from the `SysMgr Event Browser`, and the symbol color turns green. However, if the event requires immediate attention, Bill has several choices. If the event requires immediate attention, he begins the Management Session Interface (MSI) by clicking on the terminal icon on the tool bar of the `SysMgr Event Browser` (see Figure 3-5).

Figure 3-5 MSI Icon

When the MSI session is displayed, Bill has immediate access to the HP 3000 system to research the event (see Figure 3-6).

Figure 3-6 Bill's MSI Session



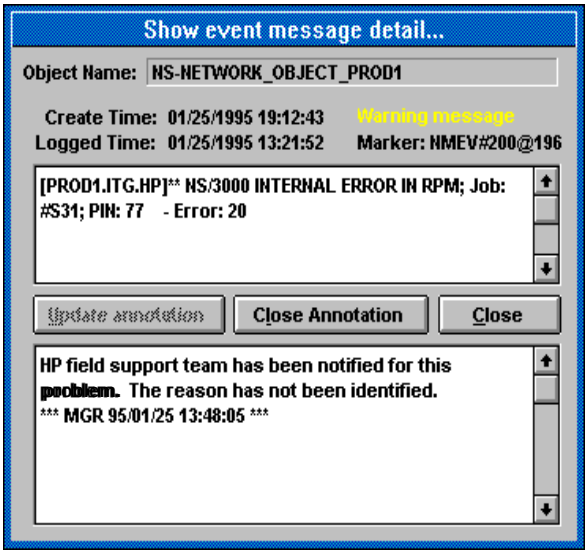
Bill used the NETALLOW.EMS.SYSMGR program from MSI to give himself access to privileged commands that allow him to do his work.

But if Bill decides that he cannot solve the problem immediately, he can use the `SysMgr Event Browser` to add annotation text to the event by clicking on the *glasses* icon on the tool bar (see Figure 3-7).

Figure 3-7 Detailed SysMgr Event Browser Icon

The `Show event message detail...` dialog box is displayed (see Figure 3-8).

Figure 3-8 Bill Adds Annotation Text



By entering this annotation, Bill can continue looking for other events or finish other work. Then by moving the event to an appropriate folder in the folder submap, the task symbol that contained the event turns to green.

If a task symbol turns red, it means that a critical event has occurred on the HP 3000. Such an event requires immediate attention. Bill uses the `SysMgr Event Browser` to look at the event and determines he needs to take immediate action. Then Bill uses MSI to solve the problem.

NOTE If a real console screen is required to solve the problem, then Bill would use the Reflection console window on the PC.

How Bill Uses the Event Folder

Bill knows that the folders on the map are not permanent storage objects. The lifetime of the event stored in a folder depends upon the EML database size and archive level. So, he makes annotations for each event he wants to research further using the `Add Annotation` feature in the `Show event message detail...` dialog box, then saves the information to a disk file. This prevents the events temporarily stored in a folder from being lost when a switch log takes place. An EML database is switched automatically to a new database when the existing one is filled to capacity. (For more detailed information about the EML database, refer to the *HP OpenView Console Manager's Guide*.) Bill can also pass the event information along to his trouble-tracking system via the clipboard.

When Bill has more time, he can work on the problem events that he stored in the folders on the folder submap.

4 Generating Reports

This chapter explains how you use SysMgr Report to generate reports of event messages.

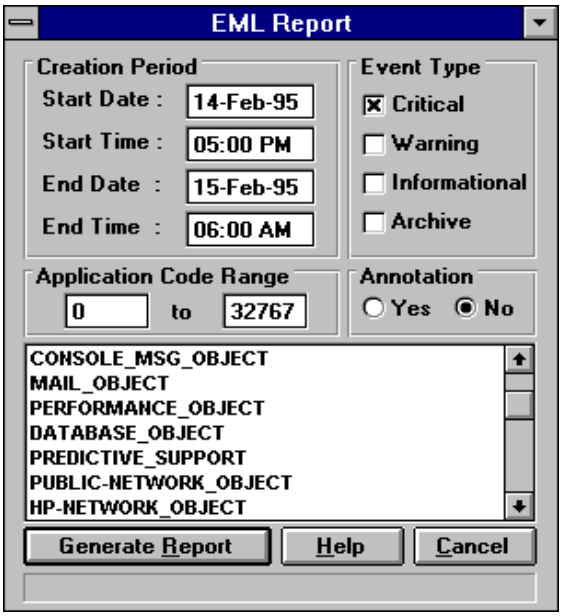
Overview

You use the SysMgr Report option from the Report menu to create customized reports of event messages from the Event Message Log (EML) database (see Figure 4-1).

Figure 4-1 HP OpenView Report Menu Option

The EML Report dialog box is displayed (see Figure 4-2). In this dialog box you can select the criteria used to generate the report.

Figure 4-2 EML Report Dialog Box

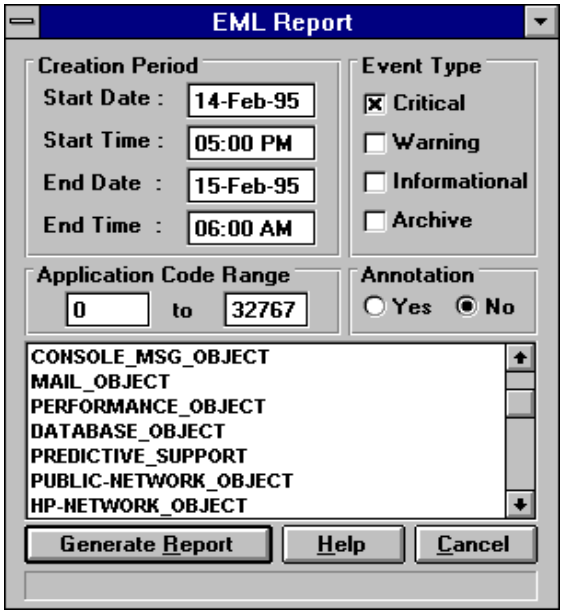


NOTE You cannot report on folder objects. Events which seem to belong to the folder object in the SysMgr Event Browser actually maintain links with the original source object. For example, an event which was moved from DISK_OBJECT to FOLDER_OBJECT still gets reported under DISK_OBJECT and does not get reported under FOLDER_OBJECT. Therefore, even though the object list box in the EML Report dialog box lists folder objects, they are not included in a report.

Creation Period Criteria

You use the four Creation Period text boxes to specify the date and time period of the events that you want in your report (see Figure 4-3).

Figure 4-3 Creation Period Text Boxes



Two times are associated with every event:

- Create time — specifies when the event was actually generated on the HP 3000
- Log time — specifies when the event was logged in the HP 3000 event EML database

Only the event's create time is used to generate a report.

Event Type

You use the Event Type check boxes to specify the type of event messages you want to include in your report. Each type of event corresponds to the HP 3000 map symbol color:

- Critical — Red
- Warning — Yellow
- Informational — Magenta
- Archive — Green

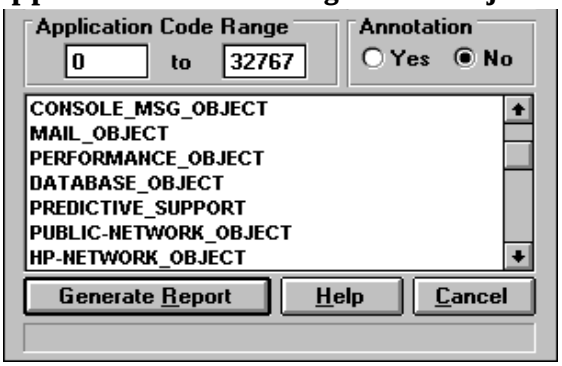
NOTE The Archive event message is not sent to the SysMgr Event Browser at all. So the report is the only place where you will see this type of event message.

Application Code Range

NOTE The Application Code Range is provided for users whose HP OpenView maps *do not* have task symbols, such as spooler, disk, security, etc.

If your map does contain task symbols, accept the default range (0-32767). Then use the object list box to select the objects you want to include in your report (see Figure 4-4).

Figure 4-4 Application Code Range with Object Name List Box



The Application Code Range fields are provided for backward compatibility with older versions of HP OpenView System Manager. Users are strongly encouraged to use the default values (0-32767).

Annotation

You select the **Yes** or **No** option button to specify whether you want to include annotation messages in your report.

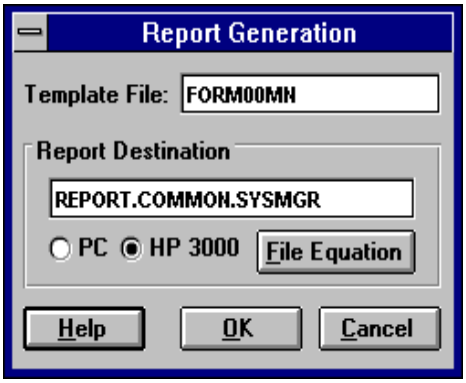
Object Name List Box

You use the object name list box to specify the objects you want to include in your report. The list box displayed in the EML Report dialog box contains all the object names known to the HP 3000 (see Figure 4-4). The list may include the names of objects not on your map and may exclude the names of objects on your map.

Report Generation

After you enter the appropriate information in the `EML Report` dialog box, click on the **Generate Report** button to specify the report format and destination. The `Report Generation` dialog box is displayed (see Figure 4-5).

Figure 4-5 Report Generation Dialog Box

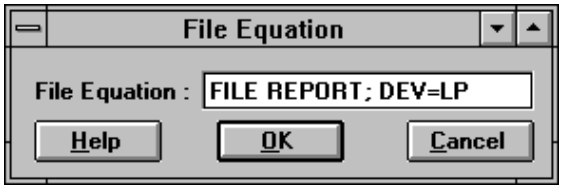


In the Template File text box you specify a report template file on the HP 3000. You will learn more about report templates in the next section, “Template Files.”

In the Report Destination dialog box you can specify the Report Destination as either the OVC PC or the HP 3000 by clicking on the PC or HP 3000 option button. Or you can send the report to a printer on the HP 3000 by following these steps:

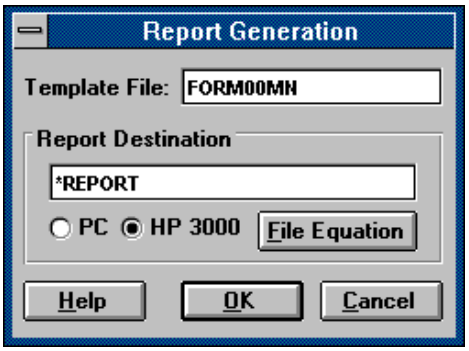
1. Click on the HP 3000 option button.
2. Click on the **File Equation** button. The `File Equation` dialog box is displayed (see Figure 4-6).

Figure 4-6 File Equation Dialog Box



3. Specify a file equation for your printer (for example, FILE REPORT;DEV=LP).
4. Click on the **OK** button.
5. In the Report Destination text box of the `Report Generation` dialog box, back reference the file descriptor specified in the `File Equation` dialog box. Using the example in, the entry would be *REPORT (see Figure 4-7).

Figure 4-7 Report Destination Showing Back-Referenced File Descriptor



Creating Template Files

You use report template s to customize reports. This section details how to create report templates to suit your needs. By using your own report templates, you can create reports that contain only the information you need in a format meaningful to you. Two default template files reside on the HP 3000 in the COMMON group of the SYSMGR account:

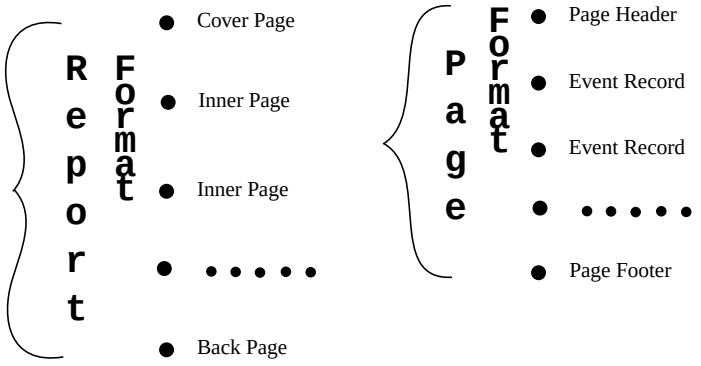
- FORM00PC — template file designed for report generation on the OVC PC
- FORM00MN — template file designed for report generation on the HP 3000

You can use one of these default templates or use your own template to create a report. In general, a report template contains three parts:

- Cover page
- Inner pages
- Back page

Figure 4-8 shows the general layout of a report.

Figure 4-8 EML Report Template



As the OVC administrator, you can define report templates for OVC users. Use an MPE text editor to make a template file which contains format keywords, user labels, and report variables that you need. The report variables serve as space holders which the report generator uses to insert the true values during report generation. Table 4-1 and Table 4-2 list the available variables:

Table 4-1 **Variables for Cover Page, Page Header, and Page Footer**

Variable	Description
<page>	: replace with page number
<page+>	: replace with page number, and increment it
<FF>	: replace with a form feed (ASCII value 12)
<report_date>	: current day/time
<management_node>	: HP 3000 name
<user_name>	: OVC user who generates the report
<rpt_min_logged_time>	: first msg logged time in this report
<rpt_max_logged_time>	: last msg logged time in this report
<rpt_min_create_time>	: first msg create time in this report
<rpt_max_create_time>	: last msg create time in this report
<rpt_min_appl_code>	: smallest application ID in this report
<rpt_max_appl_code>	: largest application ID in this report
<sel_min_logged_time>	: selection criteria — first msg logged time
<sel_max_logged_time>	: selection criteria — last msg logged time
<sel_min_create_time>	: selection criteria — first msg create time
<sel_max_create_time>	: selection criteria — last msg create time
<sel_min_appl_code>	: selection criteria — smallest application ID
<sel_max_appl_code>	: selection criteria — largest application ID
<sel_object_list>	: selection criteria — object list
<sel_msg_severity>	: selection criteria — message severities
<msg_total>	: total messages in this report
<msg_count_0>	: total archival messages in this report
<msg_count_1>	: total informational messages in this report
<msg_count_2>	: total warning messages in this report
<msg_count_3>	: total critical messages in this report
<template_name>	: template file name used
<page_width>	: page width of this report
<page_length>	: page length of this report
<page_margin>	:page margin of this report

Table 4-2 **Variables for Defining the Record**

Variable	Description
<object_name>	: message object name
<object_type>	: message symbol type
<application_name>	: application name
<application_code>	: application ID
<msg_severity>	: message severity
<create_gmt>	: GMT offset of message creation location
<msg_create_time>	: message creation time (adjusted to MN time)
<logged_gmt>	: GMT offset of message logged location
<msg_logged_time>	: message logged time (MN local time)
<message_text>	: message text
<msg_size>	: message text size
<annotation_text>	: annotation text
<db_number>	: (can be used to fetch the message)
<msg_number>	: (can be used to fetch the message)

Report Entries

The report template is an ordinary text file. Entries that start with a leading "!" character are comments. Entries that start with a "$" character are keyword lines. These keyword lines divide your report template into different sections, so you can define the layout of each report component. In the following example the keywords are defined:

```
$COVER_BEGIN
       < specify your cover page layout here>
$COVER_END

$PAGE_BEGIN
       < specify your inner page header here>
$RECORDS [(CROSS)]
       < specify your inner page footer here>
$PAGE_END

$RECORD_BEGIN
       < specify your event record layout here>
$RECORD_END

$FINAL_BEGIN
       < specify your back page layout>
$FINAL_END
```

There are limits to the number of lines you can specify for each report component:

```
Front/Back cover      : should not exceed the "page_length"
Page header/footer    : 6 lines
Record layout         : 12 lines
```

Even if a page layout declaration exceeds its size limit, the RPTEXE process does not fail. Instead, RPTEXE discards extra lines.

Report attributes such as "page_length" are defined by a set of keyword attributes. These keywords are:

$PAGE_WIDTH — Defines the page width in a range between 40 and 132 characters. The default value is 80 characters for a report on the PC and 132 characters for a report on the HP 3000, if this specification is omitted.

$PAGE_LENGTH — Defines the page length as a value that must be greater than 24 lines. The default value is 60, if this specification is omitted.

$PAGE_MARGIN — Defines the page margin. The difference between "page_width" and "page_margin" must be greater than 40 characters. The default value is 0, if this specification is omitted.

$PAGE_NUMBER — Defines the starting page number used for first <page>/<page+> variable substitution. Its value must be greater than or equal to 0. The default value is 1, if this specification is omitted.

$PAGE_FULL ON — Specifies that empty space on the page is filled with blank lines. If this keyword is missing, no blank "space" filling lines are added to a page. Use this keyword where the printer does not honor the FF (ASCII value 12) character.

Template Files

With the information given in the previous sections, you should now be able to read the FORM00PC and FORM00MN template files in the COMMON group. The FORM00PC report template is listed here for your review:

```
!
!             FORM: FORM00PC.COMMON.SYSMGR
!
!       This is an EML REPORT template file, which directs
!       the RPTEXE process to generate EML REPORTS on PC.
!
$PAGE_WIDTH 80
$PAGE_LENGTH 58
$PAGE_MARGIN 5
$PAGE_NUMBER 1
!
!       Define the front cover page
!
$COVER_BEGIN
<FF>
Event   Message Log Report <report_date> Page <page+> Management Node:
<management_node>

        *** EML Report Selection Information ****

        First msg logged date    : <sel_min_logged_time>
        Last msg logged date     : <sel_max_logged_time>
        First msg create date    : <sel_min_create_time>
        Last msg create date     : <sel_max_create_time>

        Included remote nodes    : <sel_object_list>

        Message severities       : <sel_msg_severity>
        Lowest application code  : <sel_min_appl_code>
        Highest application code : <sel_max_appl_code>

<FF>
$COVER_END
!
```

```
!
!       Define the page layout
!
$PAGE_BEGIN
Event Message  Log Report           <report_date>   Page <page>

$RECORDS

Event Message  Log Report           <report_date>   Page <page+>
<FF>
$PAGE_END
!
!      Define the record layout
!
$RECORD_BEGIN
Object        : <object_name>           Sev.: <msg_severity>
App. code     : <application_code>      Created: <msg_create_time>
App. name     : <application_name>      Logged : <msg_logged_time>
Msg text      : <message_text>
Cmt text      : <annotation_text>

$RECORD_END

!
!       Define the last page layout
!
$FINAL_BEGIN
Event Message Log Report <report_date> Page <page+> Management Node:
<management_node>


      **** EML Report Generation statistics ****

      First msg logged date      : <rpt_min_logged_time>
      Last msg logged date       : <rpt_max_logged_time>
      First msg create date      : <rpt_min_create_time>
      Last msg create date       : <rpt_max_create_time>

      Lowest application code    : <rpt_min_appl_code>
      Highest application code   : <rpt_max_appl_code>

      Archived msg count         : <msg_count_0>
      Informative msg count      : <msg_count_1>
      Warning msg count          : <msg_count_2>
      Critical msg count         : <msg_count_3>
      Total message count        : <msg_total>

      Report Generated by        : <user_name>
      Report Template used       : <template_name>

      END OF EML REPORT

$FINAL_END
```

The $RECORD_BEGIN and $RECORD_END keyword lines are required for each report. All other keyword lines are optional. For example, the most simple report template you can create has only three entries:

```
$RECORD_BEGIN
<message_text>
$RECORD_END
```

The event message report you generate from this example template has no cover page, no page header, and no page footer. It lists only message text for every event.

The default setting does not allow an event record to cross page boundaries. However, if you replace the “$RECORDS” keyword with the “$RECORDS (CROSS)” keyword, a record is allowed to cross a page boundary.

The rule of substitution used for report variables is *overwrite*. To make your report format correctly, you should be aware of how long a substitution is and what the page width is. If a formatted line is longer than the page width, characters that exceed the page width are not printed. It is also possible that a variable substitution may overwrite the succeeding variables on the same line. If this happens, the program does not fail, but the report is not formatted. Therefore, make sure that there is enough room for a variable substitution. For example, 28 characters for a time string and 50 characters for a NS/3000 node name. The “<message_text>,” “<annotation_text>,” and “<sel_object_list>” variables can do multiple line substitution. These variables remember the offset value when additional lines are required to represent the information.

5 Using Automated Response Service (ARS)

This chapter describes the Automated Response Service (ARS) and how you use ARS to respond to system events.

Overview

ARS is a fully integrated OVC feature to process events and monitor console messages. You can turn on ARS to respond to events on the HP 3000. No system overhead occurs if you do not use ARS.

When enabled on the HP 3000, ARS scans for OVC console events. That means that ARS checks only for console messages which have the NMEV# marker. The default setting for the message filter is ON. You can also set ARS to check for any text string. ARS does not use any additional processing time for the OVC event collector (the Event Message Gatherer or EMG) to provide this message filtering. ARS is a powerful tool with minimum CPU consumption: the major design guideline of ARS. However, without proper planning you can increase significantly the CPU's time consumption. By using the guidelines in this chapter, you will be able to use ARS wisely.

How ARS Works

When the OVC event collector (EMG) starts at the HP 3000, it looks for an ARS rule file. Once it finds this file, it launches ARS, which loads the rule data into memory. Since all the rules are memory resident, response trigger processing is very efficient. This rule file is used to define the triggering criteria for OVC events and response actions. It is a standard ASCII text file that can be prepared with an MPE text editor.

If an OVC event satisfies the ARS rule, ARS queues a response for it. All the events that require a response are handled by ARS, one by one. A response can be an MPE CI command, an MPE job file execution, or an MPE/iX CI script file execution. Using ARS does not require that you learn any new script language to implement response logic.

In this chapter you will learn how to write CI scripts and monitor their execution. Monitoring ARS execution is important, since to automate means that no human decision is involved once the response logic is reliable. You can monitor ARS response execution by using the monitoring console window from the OVC PC or by using the NTC00EXE program at the HP 3000. With the `Option List` command enabled for the CI script file, you can monitor line by line execution of a responding CI script.

For tracking purposes, ARS log files on the HP 3000 maintain detailed ARS response execution steps. For console messages identified as OVC event messages, the execution status is placed in the annotation part of the event message.

Customizing ARS

You can create your own ARS solutions by following some general steps and considering certain guidelines. This will help you implement smart and effective ARS solutions for your data center.

1. Identify the problem.

   First, use the right tool to do the right job. You must first define the problem. You may want to remove annoying messages, automate some task, or catch unexpected events. But you MUST keep in mind the importance of conserving CPU time. ARS is very powerful but use it as the last option for solving problems.

   The most effective way to remove unwanted OVC events is to drop them using logical object redirection. You can modify or add entries to the USER0DEF.EMS.SYSMGR file to void event messages. For information on customizing the task logical object, refer to Chapter 4, “Customizing Your Map and System Events” in the *HP OpenView Console Manager's Guide*.

   If you use ARS, the most effective rule to perform trigger checking is to identify an event from its application ID and message priority. With the help of the `SysMgr Event Browser`, find the NMEV# event marker from the detailed information window.

2. Add rules to trigger an action.

   The rule file contains the rules (matching patterns) that are compared to console messages during processing. The basic construction unit for this file is called the “decision block.” A rule file may contain just one or as many decision blocks as you need. Each decision block is formed by one or more IF commands and one DO command. The IF command is used to specify the triggering rules and the DO command is used to specify the response action. By allowing multiple IF commands in one decision block, it produces the effect of the AND operation. By allowing rule testing for every decision block in the rule file, it creates the effect of an OR operation. For example, two different decision blocks may have the same DO command, but each decision block has different IF commands.

   This design allows multiple responses for a single event, if an event satisfies multiple decision blocks. If multiple responses are desired, the response order is important. Use either CI variables or JCWs to coordinate the execution results among responses. This coordination is possible since all responses are done under the EMG process job context. Use the EMGINIT.EMS initialization file, which sets up the required job environment for ARS. If you want special capability, add the `:RUN NETALLOW` command to the file. If you want to define file equations, add JCWs or CI variables, this is the place to define them.

This also is the place to put the DSLINE and REMOTE command, if you want open VT links for this response execution environment.

The rule file is a standard MPE ASCII text file with the name EMG0AUTO. Creating and maintaining this file is your responsibility. The file equation that is specified to locate this file is not defined when OVC software is installed at the HP 3000. After the software installation, you will find a sample file in the SECURE group, but the ARS feature is disabled by default. When using the MPE EDITOR or TDP program to modify this file, remember to save this file as an unnumbered file. ARS rule file syntax for creating the rule file is described in the next section.

Try to use the IF$CODE command as often as possible. It is the most effective way to identify events. If you can respond with a single MPE command or UDC, do not write a CI script file. It may be difficult to set up and change the UDC, but it conserves CPU time. If the response is not time critical or if it requires a special job environment, you can stream a job file as the response. If none of these suggestions meet your needs, write an ARS response CI command file as a last resort.

3. Develop the ARS Response CI Command File.

   The ARS DO command can be used to execute an MPE/iX CI command file, also referred to as a command script. Writing the command script is the user's responsibility and should be considered a programming activity that requires an understanding of the MPE/iX CI Command File facility and script programming. For additional information on command files, refer to Chapter 3, "User Commands," in the *MPE/iX Commands Reference Manual*. Using command files provides a way for ARS to issue a sequence of MPE commands in response to a console message. OVC does not impose any restrictions on the use of MPE/iX command files for performing automated responses.

   Futhermore, OVC has added several CI variables. CI variables are used to assist in controlling the execution of, and in retrieving the results from, CI command file execution. These variables are accessible from within a command file. Refer to "SYSMGR CI Command File Variables and Processing" in this chapter for more information on CI command file variables and command file processing.

4. Safeguard the rule file, the CI scripts, and the response log file.

   ARS allows commands to be executed in a highly privileged command execution environment. For example, the "ARS Tutorial" section of this chapter shows you how to program ARS to execute a response when you issue an MPE TELLOP command. As a security measure, the logon ID of the user who triggers the response execution is included in the detailed ARS execution steps that are

written to the ARSLOG## log file.

Privileged information is likely to reside within the EMG0AUTO rule file, the ARSLOG## log files, and the CI command files used by ARS. As a security precaution, it is important that access to these files be restricted to guard against unauthorized access and modification. These files should NOT be placed within the EMS group of the SYSMGR account where many of the OVC files reside. EMS group access restrictions are set so that any user can read and execute files within that group.

OVC creates a new group, called SECURE, within the SYSMGR account. The EMG0AUTO rule file, the ARSLOG## log files, and the CI command files should be placed within this group. The SECURE group should have the following access capabilities:

```
ACCESS=(R,W,L,A,X,S:GU)
```

After doing this, modify the EMGINIT file (EMGINIT.EMS.SYSMGR) to include the following file equation:

```
FILE EMG0AUTO=EMG0AUTO.SECURE
```

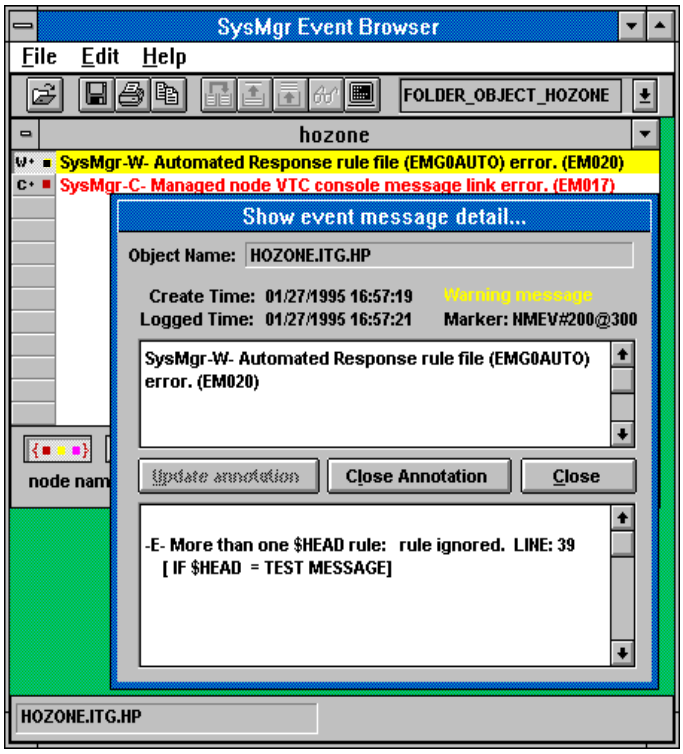
in the EMG0AUTO rule file with the proper MPE group name.

5. Enable ARS

   Automated response is enabled if the rule file, EMG0AUTO, can be found and opened at the time the EMG processes are initialized on the HP 3000. At this time, OVC reads the rule file, performs a syntax check, and stores the valid decision blocks into system memory.

   If any syntax errors were detected, a single WARNING OVC event message is forwarded to the Event Message Log (EML) on the HP 3000. The error text, the line number at which the error occurred, and the error entry for each detected error are passed as annotation text within this event message. Refer to later in this chapter for a list of syntax error messages. Figure 5-1 shows an example of a rule file syntax error EML entry. Since OVC reports only exception conditions, no event message is generated if all the decision blocks have correct syntax.

Figure 5-1 ARS Syntax Error EML Entry



Once OVC stores the valid rule file decision block into memory, the file EMG0AUTO is released by OVC and is available for modification. It should be noted that once the rule file is stored into memory, the ARS facility will not be aware of any modifications made to this file until the file is once again read and stored into memory by OVC.

OVC provides an alternative method, other than during OVC initialization, for reading the rule file and storing the valid decision block into memory. The MPE TELLOP command can be used to inform OVC to terminate the active ARS process and launch a new process that will read, and store into memory, the rule file. This is accomplished by specifying the event marker `NMEV#INIT` within the TELLOP command. For information on all the ARS-related TELLOP commands, refer to “Using the MPE TELLOP Command with ARS” later in this chapter.

6. Test and verify the installation of ARS.

   If you followed steps 1 through 5, you have the correct file equation for your rule file, defined in EMGINIT.EMS. Use either a freshly launched EMG job or a TELLOP NMEV#INIT command to enable the ARS engine. To verify that ARS is activated, issue a TELLOP NMEV#STAT command to receive the EMG’s process status. The returned TELL message indicates if ARS is enabled.

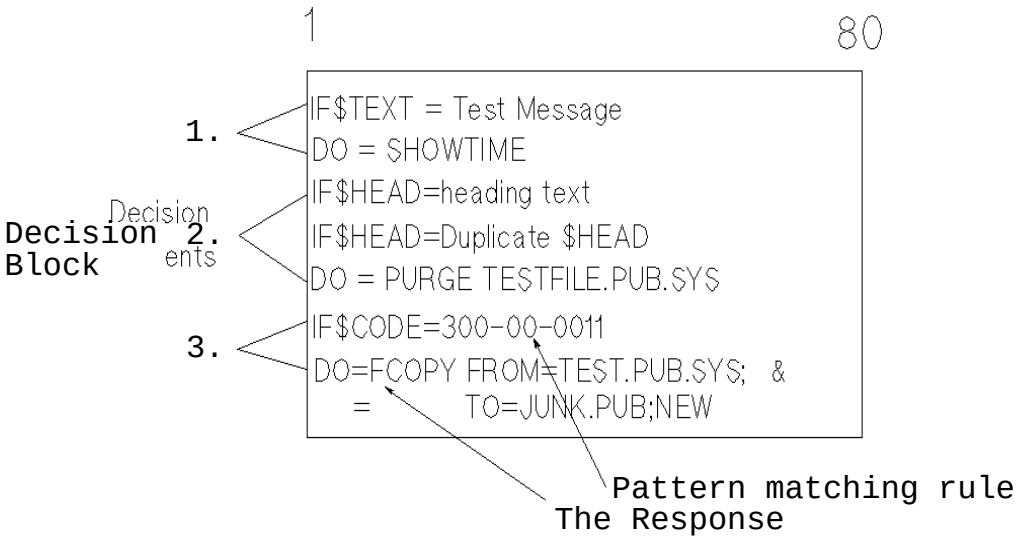
There are many ways to monitor response execution. You can monitor it using MSI. Turn **Message On** with the function key at the bottom of your terminal emulation window, so you can watch response execution in single node mode. Using the **Message On** function key activates a NTC00EXE.EMS program at your HP 3000 logon session. Run this program directly from a terminal connected to your HP 3000. For information about the NTC00EXE program, refer to Chapter 5, “OVC Utilities,” in the *HP OpenView Console Manager's Guide*.

You can find response detail information in the ARS log files. These log files reside on the HP 3000. Use the PRINT command to read the contents if ARS response is not set to the silent mode. OVC allows ARS to have silent response (which means that execution detail is not logged). You have no way to monitor a silent execution. Therefore, you must fully debug your response before setting ARS for silent response. However, it does not matter if ARS was set for silent execution or not. As long as an EML message is generated, you should be able to find the response result in the annotation text of the message.

ARS Rule File Syntax

The rule file is physically organized as a standard ASCII text file with 80 character, fixed-length records. It is logically organized as a series of decision blocks. Each decision block contains one or more pattern matching rules and one response command. The response command is executed only if all the pattern matching rules within that decision blocks are satisfied. Figure 5-2 shows an example rule file.

Figure 5-2 Example EMG0AUTO Rule File



Each match rule occupies one record in the rule file. Match rules are specified using the IF command. Refer to the IF command in “ARS Rule File Command Syntax” in the next section.

IMPORTANT

The DO command specifies the response portion of the decision block. Any valid MPE command can be specified as the object of the DO command. The DO command may occupy more than one record within the rule file. Refer to the DO command in the next section, “ARS Rule File Command Syntax.”

ARS Rule File Command Syntax

This section describes the ARS rule file syntax and correct usage for the IF and DO commands. Also at the end of the section Table 5-1 lists possible ARS syntax error messages.

IF

IF — Creates a rule statement within the ARS rule file EMG0AUTO. Allows the user to specify the search pattern to be matched within a console message.

SYNTAX — IF *keyword = pattern*

PARAMETERS

keyword — Must be one of the following:

$HEAD	Look for the matching pattern at the front of the console message.
$TAIL	Look for the matching pattern at the end of the console message.
$TEXT	Look for the matching pattern anywhere within the console message.
$CODE	Match on the application code contained within the console message.

pattern — The data being searched for within the console message. This is specified as a string of up to a maximum of 72 characters, or when used with the $CODE keyword, a valid SYSMGR application code, such as 300-00-0011, using the following syntax:

applicationID-classID-prioritymask

PARAMETERS

applicationID	A 3 digit decimal number assigned to a registered application. Refer to Chapter 4, “Customizing Your Map and System Events” in the *HP OpenView Console Manager's Guide* for more information.
classID	A 2 digit decimal value used to provide message grouping within a specific application.

prioritymask	A 4 digit binary value representing, from left to right within the mask, the 4 event message priorities of ARCHIVE, INFORMATIONAL, WARNING and CRITICAL, used within OVC. A value of 1 signifies that messages with that priority are an acceptable match; a value of 0 signifies an unacceptable match.

USE

This command can only be used within the ARS rule file EMG0AUTO.

OPERATION

Embedded spaces are allowed on either side of the keyword parameter and the "=" character. However, this will reduce the amount of characters remaining for specifying the pattern match data. Any record within this file preceeded by the "!" character is treated as a comment line and ignored by the rule file syntax checker. Spaces preceeding the "!" character are allowed.

Multiple IF$TEXT= match rule records are allowed per decision block. Only one IF$HEAD=, IF$TAIL=, and IF$CODE= match rule record is allowed per decision block. If multiple IF$HEAD= and IF$TAIL= match rule records are specified, only the first occurrence will be honored. For multiple IF$CODE= match rule records, only the last occurrence will be honored.

The pattern matching rule is case sensitive, so it is important that uppercase and lowercase be specified where necessary within the pattern matching data. Non-printing characters, such as Carriage Return, Line Feed, Space, etc., may be specified within pattern matching data using the <ddd> construct, where ddd is a decimal number with a valid range from 0 to 255. For example, a space is specified by placing <032> within the pattern matching data.

The placement of the dash character "-" within the pattern parameter is important when using the $CODE parameter. It is used as a parameter delimiter and helps define the wildcard conditions. If a parameter is not specified, then any value will constitute a valid match. The following examples will clarify its use.

IF$CODE= 300-00-0011 means match all console messages that have an application ID equal to 300 with either WARNING or CRITICAL priority. IF$CODE= --0011 means match all WARNING and CRITICAL console messages. And the match rule IF$CODE= -- means match all console messages. Finally, the match rule IF$CODE= 300--0000, with all 0s in the prioritymask field, would be an error since it means that no messages will satisfy the rule, and the decision is void.

It is important to remember that a response will be expected when all the rules for that decision have been successfully met. For those familiar with Boolean logic, this meets the criteria of the Boolean AND operator. Should you desire to execute a response based on Boolean OR logic, you can specify a decision based on a different set of rules and executing the same response.

DO or #DO

DO or #DO	Performs the MPE command or executes the MPE/iX command file. The execution result is inserted into the annotation text of the console message that is forwarded to the HP 3000. The DO command with the "#" preceeding is a silent DO command. This command is used where it is necessary to remove any trace of its execution.
SYNTAX	DO[/A]={}*{response}* [/D]
PARAMETERS	
/A	If the command is executed successfully, the priority of the console message is lowered to ARCHIVE.
/D	If the command is executed successfully, the priority of the message is set to UNKNOWN. This will cause the message to not be logged to the Event Message Log (EML) database on the HP 3000.
response	Can be any valid MPE/iX command, UDC, or MPE/iX CI command file.
USE	This command can only be used within the ARS rule file EMG0AUTO.
OPERATION	Since the maximum length of a CI command is 278 characters, and the record size of the rule file is 80 characters, the MPE standard continuation character "&" may be used to construct a DO command that requires multiple records to specify. When this is necessary, the line following the continuation character must begin with the "=" character. As with the IF command, embedded spaces are allowed on either side of the parameter and the "=" character to facilitate readability.

ARS Rule File Syntax Error Messages

Table 5-1 lists the ARS rule file syntax error messages and the system response to each message.

Table 5-1 **ARS Rule File Syntax Error Messages**

Rule File Syntax Error Message	System Response
Record syntax error:	record ignored
Invalid keyword found:	record ignored
Invalid application ID:	rule ignored
Invalid class ID value:	rule ignored
Invalid priority mask:	rule ignored
Invalid non-printable char:	rule ignored
No response before EOF:	rule ignored
More than one $HEAD rule:	rule ignored
More than one $TAIL rule:	rule ignored
Rule size exceeds 72 chars:	rule ignored
Exceeds max size (278):	response ignored
No valid match rule(s):	response ignored

ARS Rule File Processing

Console messages are checked sequentially against each decision block within the rule file. Checking is performed in the order in which the decision blocks are placed in the rule file. For each satisfied decision, a DO command (response) entry is placed within the execution queue file. Responses are executed in the order in which they are placed in the execution queue file.

A single console message can generate multiple entries within the queue by satisfying multiple decision blocks. It is important to understand that all queue entries relating to a single console message may not get executed.

This is the case when the DO/D= response execution changes the console priority to UNKNOWN. This causes OVC to discard the console message and ignore all remaining queue entries relating to this console message. With this in mind, the ordering of decision blocks within the rule file becomes important when using the DO/D= response command.

SYSMGR CI Command File Variables and Processing

OVC does not impose any restrictions for using MPE/iX CI command files to perform Automated Response execution. For information on using or programming CI command files, refer to Chapter 3, "User Commands," in the *MPE/iX Commands Reference Manual*. Also, for information about the SETVAR and SHOWVAR commands used to set and display CI variables, refer to the *MPE/iX Commands Reference Manual*.

Table 5-2 lists the OVC-added CI variables. The variables are used to help control the execution of and retrieve the results from CI command file execution. These variables are accessible from within the CI command file.

Table 5-2 CI Variables within CI Command

CI variable	Description
SM_OBJ_NAME	The object name of the PC map object. If object is the HP 3000, then object name is the NS node name in the format *nodename.domain.organization*. If object is a logical object (not a machine name), then the actual nodename can be retrieved from the variable SM_MSG_TEXT (see below). SM_OBJ_NAME is a string variable.
SM_OBJ_TYPE	Used to identify PC map object symbols, such as Large Computer Object and Medium Computer Object. The file MPE00DEF.EMS.SYSMGR, on the HP 3000, lists the types used by OVC. Event messages that are posted to the computer object symbol on your OpenView map have an SM_OBJ_TYPE value of 0.
SM_APPLCODE	Used to pass the application code to the CI command file. The application code consists of the 3 digit application ID and the 2 digit class ID. SM_APPLCODE is an integer variable.
SM_PRIORITY	Used to pass the event message priority to the CI command file. Valid values are -1, 0, 1, 2, 3 which, respectively, are UNKNOWN, ARCHIVE, INFORMATIONAL, WARNING and CRITICAL. SM_PRIORITY is an integer variable.
SM_MSG_TEXT	A string variable that contains the event message text.
SM_RESP_CNT	OVC maintains an internal counter associated with each DO response statement in the rule file. Each time a DO response statement is executed, the value of its associated counter is incremented by one. The ARS Response Handler program sets the value of SM_RESP_CNT to the value of this counter before executing the CI command file. When control is returned to the ARS Response Handler, after the CI command file has finished executing, the value of SM_RESP_CNT is saved for that DO response statement. SM_RESP_CNT can be used to control execution flow of the response CI script.

With the exception of the SM_MSG_TEXT variable, the value of each variable may be reset by the command file during execution.

Upon completing execution of the DO command, command file variables associated with the console message, except for SM_MSG_TEXT, are updated to reflect any changes to the variables that may have occurred during DO command execution. This enables the command file to change the values of these variables based on the outcome of its execution; this includes overriding the console message priority assigned as a result of specifying the /A= and /D= parameters.

Two additional CI variables, SM_STATUS and SM_RESULT, are provided. These can also be set by the CI command file during execution. SM_STATUS can be set to reflect the success of the command file execution. If this variable exists, its value will be inserted into the Automated Response result area within the console message annotation text. SM_RESULT is a string variable whose value, if it exists, will be passed as the Automated Response result.

Using the MPE TELLOP Command with ARS

This section describes the MPE TELLOP commands used with ARS. The part of the command following the "#" character can be either uppercase or lowercase.

NMEV#KILL Stops the ARS command execution program, ARS00EXE, and disables the ARS facility.

NMEV#INIT Starts a new ARS initialization and stops existing ARS execution, if the ARS process exists.

NMEV#FOFF Disables the NMEV# filter. This allows ARS to perform Automated Response for all console messages, not just those containing the event filter NMEV#. If the JCW variable EMGOPEN is defined within the EMGINIT file, then ARS is started with the filter disabled.

NOTE When the NMEV# filter is disabled, EMG passes all console messages to ARS. Operating in this mode is very expensive in CPU time, but it can trigger a response for messages without the NMEV# marker. Events without markers are dropped if no response raises the message severity.

NMEV#F_ON Enables the NMEV# filter. This is the default setting.

NMEV#STAT As it relates to ARS, reports the status of Automated Response and the NMEV# filter. The status is reported as ON or OFF.

ARS Log File

Both the detailed execution procedure and result, or status, of the Automated Response execution are accessible after processing is completed. The execution status is logged in the event message annotation area, and the detailed execution steps are logged to the log file ARSLOG##, where ## represents the log file index number whose valid range is from 1 to 99. You can read the log file, which is an ASCII text file, by using the FCOPY command, the PRINT command, or an ASCII text editor program.

When an ARS log file becomes full, OVC automatically performs a process known as the ARS Switch Log. This process deactivates the current ARS log file and creates a new log file in which the value of the numeric portion of the filename is incremented by 1. That is, a switch log creates log file ARSLOG05 once log file ARSLOG04 is full. If the log file already exists, the old copy is removed before creating the new copy. Once log file ARSLOG99 is reached, the next switch log creates ARS log file ARSLOG01.

The ARSLOGKEEP JCW variable is defined in the EMG job template file, EMG00TPL.EMS. This template file is used to build the EMG job file EMG00JOB.EMS. ARSLOGKEEP is used to specify the number of log files currently maintained on disk and, therefore, accessible for viewing. If ARSLOGKEEP is not defined, or is set to a value less than 2, OVC assigns a default value of 2. If the value specified is greater than 99, the value is set by OVC to 99. Using the above example with a default ARSLOGKEEP value of 2, ARSLOG03 would be deleted during the switch log to ARSLOG05, since an ARSLOGKEEP value of 2 specifies that only 2 log files are to exist at the same time. If the switch log process encounters a problem deleting an old copy of the log file, because the file is being read or is missing, the switch log process does not report an error message.

The ARSLOGSIZE JCW variable is also defined in the EMG00TPL job template file, as discussed above. ARSLOGSIZE is used to specify the number of records within the log file. The default value is 1024 records.

The switch log process generates an event message to signify that it has completed. The newly created log file name is inserted in the event message annotation text to inform the user of the name of the current ARS log file. Figure 5-3 shows an example of log file listing generated by ARS.

Figure 5-3 ARS Log File Listing

```
:FCOPY FROM=ARSLOG01;TO=
HP32212A.3.39 FILE COPIER (C) HEWLETT-PACKARD CO.1984


****[HANGTOWN.ITG.HP] 95/02/14-11:31:06(ARSLOG01)
*** AUTOMATED RESPONSE ***
SM_OBJ_NAME = HANGTOWN.ING.HP
SM_OBJ_TYPE = 0
SM_APPLCODE = 31000
SM_PRIORITY = 2
SM_MSG_TEXT = 11:31/#587/493/FROM/MGR.SYSMGR/Example Test
Message to verify Automated Response.

:showtime
TUE,FEB 14, 1995, 11:31 AM
```

ARS Tutorial

In this tutorial you will learn the basics for using OVC ARS so that you can further explore ARS on your own. The tutorial contains two exercises:

- **Exercise 1** — shows you how to program ARS so that specific users can execute MPE commands from within the EMS.SYSMGR job context.
- **Exercise 2** — shows you how to program ARS so that nonprivileged users at the HP 3000 can interactively receive MPE LOGON messages.

NOTE Before trying these exercises, it is strongly recommended that you first read this chapter.

In the exercises you will use three sample files:

- EMG0AUTO.SECURE — sample rule file (used in both exercises)
- EXECUTE.SECURE — sample CI command file
- NOTICE.SECURE — sample CI command file

With minor changes to reflect your operations environment, you can use these sample files to expand your ARS functionality. (Refer to the listings for these sample files at the end of this chapter.)

During execution of the OVC software installation script, the SECURE group is created within the SYSMGR account on the HP 3000.

Before doing the exercises, you need to follow some steps to enable ARS and verify that ARS is running on the HP 3000.

Follow these steps to enable and verify that ARS is running on the HP 3000:

1. Stop the JCOMMON job by issuing the UDC command:

   ```
   OVSTOP
   ```

2. Stop the EMG job (EMG00JOB) by issuing the following MPE command while logged on to the HP 3000:

   ```
   TELLOP NMEV#EXIT
   ```

3. Modify the MPE file EMGINIT to add the following file equation:

   ```
   : FILE EMG0AUTO=EMG0AUTO.SECURE
   ```

4. Restart OVC on the HP 3000 by typing the following UDC command at the MPE prompt:

   ```
   OVSTART
   ```

5. Verify the status of OVC by issuing the following command.

   ```
   TELLOP NMEV#STAT
   ```

This command should be issued *after* the EMG job shows up in an MPE SHOWJOB listing.

The status is displayed in an MPE TELL message such as:

```
FROM/J12 MGR.SYSMGR/Logical obj:OFF Auto-response:ON
Filter:ON Tracing:OFF
```

The two ARS-related status indicators are *Auto-response* and *Filter*. “Autoresponse:ON” indicates that ARS is enabled. “Filter:ON” indicates that the OVC event message filter is enabled. When this filter is enabled, ARS processes only those console messages that have the event marker string “NMEV#?” inserted within their text. The “?” in the event marker represents a value from 0 to 3 which indicates one of the OVC event message priority values.

Exercise 1

In this exercise you will learn how to program ARS so that specific users can execute MPE commands from within the EMS.SYSMGR (EMG00JOB) job context. Ordinarily, the MPE logon ID for these users would not be granted privileged capability, and these users would be unable to execute privileged MPE commands. However, by running the OVC utility program NETALLOW, the logon ID for the EMG00JOB can be granted privileged capability and can execute the privileged MPE commands for the users.

To begin this exercise, on the HP 3000 you must first create an account named TEST and the users MGR and USER, if they do not already exist. ARS (from within the ARS job context) authorizes the users MGR.TEST and USER.TEST to execute MPE commands.

Now that ARS is enabled, you will generate a few console messages to test the operation of ARS. Follow these steps:

1. At the MPE prompt, while logged on as MGR.SYSMGR, issue the following command:

   ```
   TELLOP NMEV#0 perform SHOWME
   ```

   After issuing this command, you receive a TELL message from the EMG job (EMG00JOB) such as:

   ```
   FROM/J12 MGR.SYSMGR/MGR.SYSMGR is not an authorized user
   ```

   If you have a map displayed on the OVC PC, you see the map symbol for the HP 3000 turn RED, indicating a critical message has been logged to the Event Message Log (EML).

2. Review the message using the `SysMgr Event Browser`. Review the annotation that is attached to the message.

To understand what happened when you issued the TELLOP command, look at both the rule file and the CI command file EXECUTE.SECURE. The TELLOP command satisfied the IF command matching pattern for the following decision block within the rule file:

```
IF $TEXT=perform

DO = EXECUTE.SECURE
```

Since a pattern match is found on the word *perform*, ARS executes the CI command file EXECUTE.SECURE. Looking at this CI command file listing, you see this sequence of events:

- The user that sent the TELLOP command is isolated within the CI variable SM_MSG_TEXT.
- A check is made to determine if the user is either USER.TEST or MGR.TEST.
- If the user is not USER.TEST or MGR.TEST, as is the case here, the CI variable SM_RESULT is assigned the *unauthorized user* error message and the CI variable SM_PRIORITY is assigned a value of 3. As a result, the HP 3000 map symbol turns RED, and a CRITICAL message is sent to the Event Message Log on the HP 3000.
- Send the execution result TELL message to the user.

In the remaining steps, you will test the case where the TELLOP command is issued by an authorized ARS user (MGR.TEST):

1. Log off as MGR.SYSMGR and log on as MGR.TEST.
2. Issue the following MPE TELLOP command:

   ```
   TELLOP NMEV#0 perform SHOWME
   ```

   This time you will receive a TELL message from the EMG job (EMG00JOB) such as:

   ```
   FROM/J12 MGR.SYSMGR/Execution completed successfully
   ```

   After receiving this message, you can verify the successful execution of ARS by printing out the contents of the ARS log file within the SECURE group of the SYSMGR account. This file should have a name such as ARSLOG01.

Again, since a pattern match is found on the word *perform*, ARS executes the CI command file EXECUTE.SECURE. Looking at this CI command file listing, you see this sequence of events. The initial sequence of events within the CI command file is the same as for the first TELLOP command that you issued. However, this time the sender of the TELLOP command matches one of the authorized users.

- The command portion of the TELLOP message (SHOWME) is isolated within the CI variable SM_MSG_TEXT.

- The command string SHOWME is echoed to the listing device.
- The command SHOWME is executed.
- The CI variable SM_RESULT is assigned the *Execution completed successfully* message and the CI variable SM_PRIORITY is assigned a value of 1. As a result, the HP 3000 map symbol turns MAGENTA, and an INFORMATIONAL message is sent to the Event Message Log on the HP 3000.
- Send the execution result TELL message to the user.

Exercise 2

NOTE This example is taken from the *HP OpenView System Manager User's Guide*. It shows how ARS can trigger an event at one HP 3000 and be responded to at another HP 3000. You cannot try this exercise if you have not upgraded your OVC to SysMgr.

In this exercise you will learn how to program ARS so that nonprivileged MPE users at the management HP 3000 can interactively receive MPE LOGON messages from a remote HP 3000. Immediately after a user logs on to the remote HP 3000, the user MGR.SYSMGR at the management HP 3000 receives two TELL messages indicating the logon has occurred. The first TELL message contains the HP 3000 name and MPE job number. The second TELL message contains the user name and account name.

Follow these steps to complete the exercise:

1. Log on to the remote HP 3000 as MGR.SYSMGR.
2. Remove the comment indicator (!) from the following decision block in the remote HP 3000 rule file EMG0AUTO.SECURE.

   ```
   IF $TEXT=LOGON FOR:

   DO = >>>
   ```

   This decision block redirects ARS execution from the remote HP 3000 to the management HP 3000.
3. Reinitialize ARS on the remote HP 3000 by issuing the command:

   ```
   TELLOP NMEV#init
   ```
4. Issue the following command to *disable* the NMEV#? filter. You must do this since the MPE LOGON message does not contain the OVC Event Marker "NMEV#?" where the "?" represents a value from 0 to 3.

   ```
   TELLOP NMEV#foff
   ```
5. Log on to the management HP 3000 as MGR.SYSMGR.

6. Remove the comment indicator (!) from the following decision block in the management HP 3000 rule file EMG0AUTO.SECURE.

   ```
   IF $TEXT=LOGON FOR:

   DO = NOTICE.SECURE
   ```

7. Reinitialize ARS on the management HP 3000 by issuing the command:

   ```
   TELLOP NMEV#init
   ```

8. Open a Reflections console session.

9. Log on as MGR.SYSMGR on the management HP 3000.

   This is the user who will receive the remote HP 3000 user LOGON messages. As users log on to the remote HP 3000, you should see the TELL messages indicating the logons on your display.

NOTE The map symbol for the remote HP 3000 where the logon occurred does not change color. Also no event message is added to the event message review. However, these logon messages are logged to the Event Message Log (EML) on the management HP 3000.

You need to look at both the rule file and the CI command file NOTICE.SECURE to understand how the user MGR.SYSMGR on the management HP 3000 received logon TELL messages from the remote HP 3000.

Removing the comment indicators from the second decision block in the remote HP 3000 rule file EMG0AUTO.SECURE resulted in redirecting ARS execution from the remote HP 3000 to ARS on the management HP 3000. This is necessary since you want the user MGR.SYSMGR on the management HP 3000 to receive the TELL messages.

Removing the comment indicators from the third decision block within the management HP 3000 rule file EMG0AUTO.SECURE enabled ARS to check on the management HP 3000 for the “LOGON FOR” string within console messages.

Each time a logon occurs on the remote HP 3000, since a match is found on the pattern `LOGON FOR:`, ARS executes the CI command file NOTICE.SECURE on the management HP 3000. Looking at this CI command file listing, you see this sequence of events:

- The TELL message containing the HP 3000 and job number is constructed and sent to MGR.SYSMGR.
- The second TELL message containing the user name and account name is constructed and sent to MGR.SYSMGR.
- The CI variable SM_PRIORITY is reassigned a value of 0, which means ARCHIVE. This is important since the default message priority of a console message without the NMEV#? event marker is 1

(INFORMATIONAL). As stated earlier, LOGON console messages do NOT have the event marker inserted in them.

If the value is not changed to zero, the map symbol for the remote HP 3000 receives an INFORMATIONAL (MAGENTA) message.

EMG0AUTO.SECURE Rule File Example

```
! File: EMG0AUTO.SECURE (A0100000)
!
! This is an example rule file for OVC Automated Response.
!
!
! example 1: (MPE-XL only)
!
! If user issues ":tellop NMEV#0 perform command", the MPE
! command will be executed by the ARS under the EMS.SYSMGR job
! context.
! For example:
!
! :TELLOP NMEV#0 perform SHOWJOB
!
! result will be logged in the ARSLOG??.SECURE.SYSMGR at
! the remote HP 3000.
!
! If user uses NMEV#1 instead of NMEV#0 in the :TELLOP command,
! the message is sent to the management HP 3000 and it can be
! reviewed from the OVC PC interface.
!
IF $TEXT = perform<032>
DO = EXECUTE.SECURE
!
!
! example 2: (applies to both MPE-V and MPE-iX remote node)
!
! This example shows that a user MGR.SYSMGR on the management
! HP 3000 can receive all the LOGON console messages from the
! remote HP3000.
!
! **** If this is remote HP 3000, remove the comment mark "!" for
! **** the following decision block.
!
!IF $TEXT =LOGON FOR:
!DO = >>>
!
! **** If this is management HP 3000, remove the comment mark "!"
! **** for the next decision block.
!
!IF $TEXT =LOGON FOR:
!DO = NOTICE.SECURE
!
! **** END OF EMG0AUTO.SECUR
```

EXECUTE.SECURE CI Command File

```
comment option list
comment
comment File: EXECUTE.SECURE (A0100000)
comment This is an example CI script file for ARS exercise one.
comment
comment This CI command file checks the user's credibility for
comment executing MPE commands within MGR.SYSMGR's job context.
comment Only MGR.TEST and USER.TEST are allowed.
comment
comment The valid syntax is
comment
comment :TELLOP NMEV#0 perform command
comment
comment for example:

comment :TELLOP NMEV#NMEV#0 perform LISTF
comment

setvar _status 1
setvar _start pos('/FROM/', '!sm_msg_text')
if !_start 0 then
      setvar _msg_text str('!sm_msg_text', (_start+6), len
                  ('!sm_msg_text'))
      setvar _start pos('/', '!_msg_text')
      if !_start 0 then
      setvar _user str('!_msg_text', 1, (_start-1))

comment
comment Only USER.TEST and MGR.TEST are allowed
comment

      if '!_user' = 'USER.TEST' then
             setvar _status 0
      endif
      if '!_user' = 'MGR.TEST' then
             setvar _status 0
      endif

comment
comment Fetch the command from SM_MSG_TEXT
comment

      if !_status = 0 then
             setvar _cmd_start pos('perform ', '!sm_msg_text')
             if !_cmd_start 0 then
             setvar _cmd_size (len('!sm_msg_text')
                          -(!_cmd_start + 8)) + 1
                    setvar _cmd_strg Rht
                           ('!sm_msg_text', !_cmd_size)

comment
comment Echo the MPE command and execute it
comment

      echo !_cmd_strg
      !_cmd_strg
      setvar sm_result 'Execution completed successfully'
      setvar sm_priority 1
      endif
      else
      setvar sm_result '!_user is not an authorized user'
```

```
            setvar sm_priority 3
      endif

comment
comment Return the result
comment

            tell !_user !sm_result
      endif
endif

comment *** END EXECUTE.SECURE script ***
```

NOTICE.SECURE CI Command File

```
comment option list
comment
comment File: NOTICE.SECURE (A0100000)
comment This an example CI script file for ARS exercise.
comment
comment This CI command file breaks a LOGON message such as
comment
comment 15:33/#S174/282/LOGON FOR: MANAGER.SYS,PUB
comment ON LDEV #585
comment
comment into two and sends each to MGR.SYSMGR user at
comment management HP 3000 via MPE TELL command.
comment

setvar _logon_pos pos('/LOGON', '!sm_msg_text')
setvar _logon_src str('!sm_msg_text', 1, (_logon_pos -1))
setvar _tell_text "["+"!sm_obj_name"+ " "+ "!_logon_src"+ "]"
tell MGR.SYSMGR !_tell_text<R>
setvar _logon_len len('!sm_msg_text')
setvar _logon_msg str('!sm_msg_text', (!_logon_pos + 1), &
(!_logon_len-!_logon_pos))
setvar _tell_text "["+ "!_logon_msg" + "]"
tell MGR.SYSMGR !_tell_text

comment
comment Next SETVAR command drops the message priority to
comment ARCHIVE; it will be inserted into the EML
comment database, but not notify the PC.
comment
comment You can modify the next line to
comment
comment setvar sm_priority -1 will discard the message
comment setvar sm_priority 2 will escalate it to a
Yellow message
comment setvar sm_priority 3 will escalate it to a Red
message
comment

setvar sm_priority 0
setvar sm_result 'TELL MGR.SYSMGR command done successfully'

comment *** END NOTICE.SECURE script ***
```

A Console Automation Scripts

This appendix describes the Reflection scripts used to automate console connections to HP 3000 systems.

Overview

In the OVC/SysMgr solution, you can access the console of HP 3000 systems in various ways. To obtain full console access capabilities (**Ctrl-A** and **Ctrl-B**) to an HP 3000, you must have access to its console port (ldev-20 or ldev-21). You can obtain this access several ways depending on the proximity of the HP 3000 system to the OVC SysMgr PC.

For HP 3000 Systems Close to an OVC/SysMgr PC

If the HP 3000 system is located close to the OVC/SysMgr PC:

1. Use a special HP cable specifically made for OVC (HP 8120-6415) to connect any PC COM port to the HP 3000's console port (ldev-20).
2. Use a local DTC as the switchbox.

   The PC's COM port is connected to a terminal port on the DTC, and a host port on the DTC is connected to ldev-21 of the HP 3000. The advantage of this solution is that you can use the same COM port to connect to different HP 3000s.

For HP 3000 Systems Far from an OVC/SysMgr PC

If the HP 3000 system is located far from the OVC/SysMgr PC:

1. Use modems to connect the OVC/SysMgr PC's COM port to the modem port (ldev-21) of the HP 3000.

   You need to use two modems, one at the PC end and one at the HP 3000 end.
2. Use a back-to-back DTC solution.

   This solution uses two DTCs: one next to the PC and the other next to the HP 3000. For more information on setting up DTCs, refer to the HP OpenView DTC Manager manuals.

SysMgr LAN Console

The SysMgr LAN Console program is used to automate the console connection once you have set up the hardware for either of the previously described solutions. Depending on the type of solution use, you must configure a PC COM port as either NAILED or SWITCHABLE. COM ports that are not used, or are not available, should be configured DISABLED. You configure the COM port as NAILED (to an HP 3000) in LAN Console if you think that the port is going to be used only to connect to the specified HP 3000 (as in the case of a direct serial connection). You configure the COM port to be SWITCHABLE if it is going to be used to connect to more than one HP 3000 (as in the case of using a DTC to switch between system consoles).

Choosing a specific configuration decides the contents of the configuration files that are created to automate the console connection. For every HP 3000 that you get console access to, a corresponding Reflection script file and a Reflection configuration file are created. These files reside in the `c:\ov\sysmgr\console` directory and have names identical to the node names of the HP 3000s that they represent. For HP 3000s that are NAILED to a COM port (that is, for HP 3000s that are connected using NAILED ports), the Reflection script files have the COM port number hardcoded.

For HP 3000 systems that can be connected using SWITCHABLE ports, the COM port number is not hardcoded, and at console connection time a free COM port is chosen from the list of available SWITCHABLE ports, and a connection is initiated. A disadvantage of this is if you have two SWITCHABLE COM ports, one using a modem to connect and the other using a DTC. Although both connections are capable of switching, both COM ports cannot be configured SWITCHABLE. The problem is obvious.

For example, assume that you configured COM1 and COM2 as SWITCHABLE with COM1 using a modem and COM2 using a DTC. You intended to connect to System A using the modem and System B using the DTC. But in the case when you try to create a console connection to System B while COM1 is free (you are not accessing System A's console), LAN Console chooses COM1 to connect to System B because it is the first SWITCHABLE port that is available. However, this won't work because System B is not accessible using a modem but by using a DTC. You can overcome this problem by configuring COM1 port as NAILED to System A. This way, COM1 is never used to initiate a connection to any other system except System A.

A separate Reflection script file is created for every HP 3000 that you want console access to. These Reflection scripts contain information related to your console connection (for example, baud rate, DTC

password, phone number, etc.). This information is collected the first time you create a console connection to the node. For subsequent connections, a console connection initiation script uses this connection information to establish the connection.

The next sections contain the names of the script files and configuration files that are used to create console connections with an explanation of what they do.

Script Files

build.rcl

This script executes whenever the user invokes LAN Console for a NAILED system that was not accessed previously and, therefore, has no configuration or script file. This script asks the user for the specifics of connecting to the system and builds a configuration file and a script file to carry out the access automatically for all subsequent operations.

builddtc.rcl

This script executes whenever the user uses LAN Console to establish a control console through a SWITCHABLE COM port to a system that has not been accessed previously and, therefore, has no configuration file or script file. This script asks the user for the specifics of connecting to the system and builds a configuration file and a script file to carry out the access automatically for all subsequent operations.

hiconsol.rcl

This script is used to connect to a remote system's console. The connection is either a MODEM, DTC, or SERIAL port. For modem connections, the "DIALHOST.RCL" script shipped with the Reflection product is used. This script is used only for NAILED connections.

hicondtc.rcl

This control console script is executed connect to a remote system's console using a SWITCHABLE COM port. The connection is either a MODEM, DTC, or SERIAL port. For modem connections, the "DIALHOST.RCL" script shipped with the Reflection product is used. This script is usually launched with the configuration file, <nodename>.r1w. If this is the first time connection to a node with a SWITCHABLE port then this script is called by BUILDDTC.RCL. This script is used only for SWITCHABLE connections.

byeconsol.rcl

This script terminates a remote console session. The connection is either a MODEM, DTC, or SERIAL port.

dialhost.rcl

This script file is included with Reflection 1 for Windows and is used to dial out using modems. This script is executed when the modem solution is used to get console access.

console.rcl

This is a dummy script file used with the CONSOLE.R1W configuration file for OV Console installations. This file is required by LAN Console to launch a connection on COM1.

<nodename>.rcl

This is a script file maintained for the HP 3000 system whose name is <nodename>. This file contains console connection information (connection type, connection speed, relevant passwords, etc.) specific to connecting to the system <nodename>.

Configuration Files

build.r1w

This Reflection configuration file sets up the initial configuration of the Reflection Terminal Emulator window which is invoked when a console script has to be built for the first time. In the initial configuration, the Reflection window does not have any active connection.

<nodename>.r1w

This Reflection configuration file sets up the initial configuration of the Reflection Terminal Emulator window which is invoked when you initiate a console connection to the system <nodename>. This configuration file sets up the initial connection to “NONE” and sets the window title to “<nodename>.r1w”.

console.r1w

This Reflection configuration file sets up the initial configuration of the Reflection Terminal Emulator window which is invoked when the default console connection is initiated on COM1 to the local HP 3000 system. This is identical to the <nodename>.r1w configuration file where <nodename> is replaced with “console.” Figure A-1 and Figure A-2 show the logic of two important script files, BUILDDTC.RCL and BUILD.RCL.

Figure A-1 Execution Logic to Connect via a SWITCHABLE Port

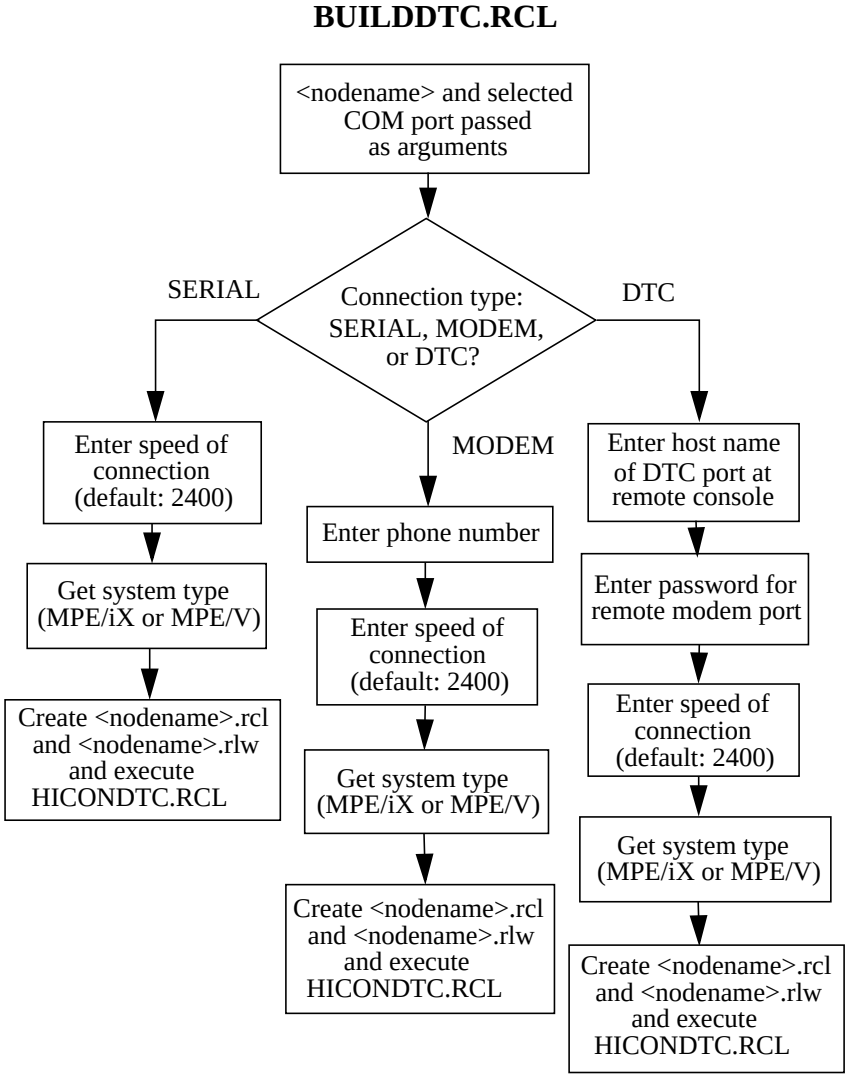
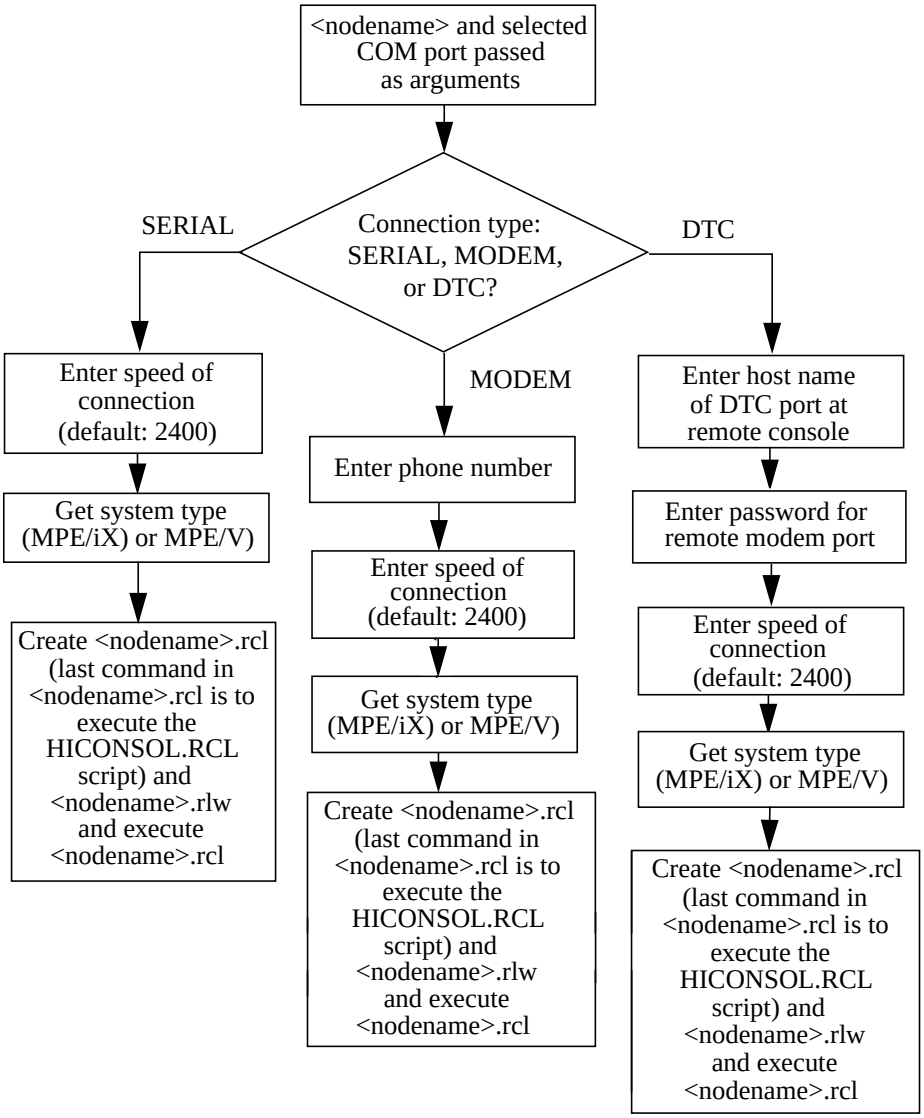


Figure A-2 Execution Logic to Connect via a NAILED Port



Once the <nodename>.rcl and <nodename>.rlw files are created, subsequent console connections do not execute BUILD.RCL or BUILDDTC.RCL. For NAILED connections, the <nodename>.rcl for the NAILED node is executed (this in turn executes HICONSOL.RCL which establishes the console connection).

For nodes connected using SWITCHABLE COM ports, LAN Console scans for an available port in the PC which is configured SWITCHABLE. If LAN Console finds a SWITCHABLE port, it passes it as an argument to HICONDTC.RCL. Then HICONDTC.RCL loads the appropriate <nodename>.rcl script file containing the connection information and establishes the connection. (The <nodename> is also passed as another argument to HICONDTC.RCL.)

B Error Messages

This appendix explains the error messages that you may see when using OVC. These messages may be displayed on the OVC PC screen, in an HP 3000 log file, or in the SysMgr Event Browser dialog box.

OVC PC Error Messages

The following error messages may be displayed in a dialog box on the OVC PC screen while you are running OVC.

IMFVAL ERR 3

MESSAGE: Unexpected SysMgr CommMgr Error: Error <error code> Program will terminate.

CAUSE: The SNA IMF path is missing from the configuration file.

ACTION: Restore the configuration file from a backup tape and retry the validation. Also, submit an SR.

P0-00

MESSAGE: Corrupt configuration file. Missing IMF path. (IMFVALERR 3)

CAUSE: Program defect

ACTION: Contact your HP representative

P0-02

MESSAGE: SysMgr CommMgr program missing! Program will terminate.

CAUSE: Someone closed the COMMMGR window or COMMMGR failed.

ACTION: Restart the c:\ov\sysmgr\commmgr.exe program using the "Run" menu item from the Windows Program Manager.

P0-11

MESSAGE: Network error happened. Program will terminate.

CAUSE: Some fatal network error happened.

ACTION: Some other message box should appear to report network error. Contact your local network administrator for assistance.

P0-20

MESSAGE: Error saving file <file name> <file error text>.

CAUSE: A disk (file) error occurred on your PC.

ACTION: Fix the problem.

P0-21

MESSAGE: SysMgr <application program name> is already loaded!

CAUSE: This is an unexpected error.

ACTION: Close the COMMMGR window and restart it using the "Run" menu item from the Windows Program Manager.

P0-22

MESSAGE: Logon user [<user name>] does not have required capability! See your SysMgr administrator for assistance.

CAUSE: User missing "SysCnfg" capability.

ACTION: Have your OVC administrator add the "SysCnfg" capability to your OVC logon.

P0-23

MESSAGE: CommMgr returned Error code : <errno>.

CAUSE: Program could not open a connection to the HP 3000.

ACTION: Make sure CommMgr is running.

P0-24 **MESSAGE: Unable to create window.**

CAUSE: The creation of the window or dialog box corresponding to the command or button you selected failed. Most likely, you are running out of memory.

ACTION: Close as many other Windows programs as possible.

P0-25 **MESSAGE: Action could not be completed due to a memory error.**

CAUSE: A failure occurred in accessing memory.

ACTION: Restart MS-Windows.

P0-26 **MESSAGE: Insufficient user capabilities.**

CAUSE: You don't have the capabilities necessary to access this command.

ACTION: Log off and log on as a user with the required capabilities, or select another command for which you have the required capabilities.

P0-27 **MESSAGE: Connection with the Management node has been lost.**

CAUSE: HP 3000 is down; network is down; or one or both the OVC jobs on the HP 3000 are down.

ACTION: Check to see if the OVC jobs are running on the HP 3000 and whether the network is up.

P0-28 **MESSAGE: Incompatibility detected between workstation and management node software.**

CAUSE: The OVC PC software and the OVC HP 3000 software are not version compatible.

ACTION: Reinstall the appropriate version of the PC/HP 3000 software.

P0-29 **MESSAGE: Configuration is already being accessed.**

CAUSE: Another user is accessing the same configuration command.

ACTION: Wait until the user is done.

P0-30 **MESSAGE: There are no window timers available to run this application.**

CAUSE: This is due either to a low memory condition or a software defect.

ACTION: Close as many other Windows programs as possible.

P1-01 **MESSAGE: The required management node job [<job name>] does not exist!**

CAUSE: The required job on the HP 3000 has either not been activated

or has failed.

ACTION: Use OVSTART UDC to launch the job if the job has not been started yet. In case of job failure, check the job spool file and the log file (OVSLOG00.COMMON.SYSMGR for JCOMMON) for the failure reason.

P1-02 **MESSAGE: Normal management node job [<job name>] shutdown detected!**

CAUSE: The required job has been terminated by someone on the HP 3000.

ACTION: If the automated link recovery is enabled, the map will receive a color change when the server job is restarted on the HP 3000. However, a new user logon is required.

P1-03 **MESSAGE: The entry (MgmtNode=<management node name>) in file <sysmgr.ini file> is not valid!**

CAUSE: The HP 3000 entry specified in the sysmgr.ini file is not a known node name.

ACTION: Check the hosts file on your PC, or contact your local network administrator for assistance.

P1-04 **MESSAGE: No Application is available for termination!**

CAUSE: You tried to abort the communication link between the COMMMGR and an OVC application. However, you did not select any application from the list box.

ACTION: Do nothing.

P1-05 **MESSAGE: Network response too slow to support this application!**

CAUSE: Your local network has a traffic jam. Either your network is running out of bandwidth, or your network has a bottleneck somewhere.

ACTION: Normalize COMMMGR and try to restart the connection. If the network still gives you trouble, contact your local network administrator for assistance.

P1-06 **MESSAGE: Fail to load WINSOCK.DLL\n (required version: %d.%d) <version found>.**

CAUSE: The WINSOCK.DLL on your PC does not meet the version requirement.

ACTION: Update the network software for your PC.

P1-07 **MESSAGE: Invalid/missing default user (nocap) in configuration database.**

CAUSE: Either you have an invalid NOCAP user in your user config database, or you do not have that entry.

ACTION: You need to add a NOCAP user entry in your user config database. This user should not be given any capability.

P1-09 **MESSAGE: Version mismatch between the PC and management node!**

CAUSE: Your HP 3000 software does not support this version of PC software.

ACTION: You need to upgrade your HP 3000 software.

P1-10 **MESSAGE: Unexpected fatal management node job [<job name>] abort!**

CAUSE: The <job name> failed on the HP 3000.

ACTION: This is caused by either a network failure or a job failure. If this is a job failure, check the job spool file and the log file (OVSLOG00.COMMON.SYSMGR for JCOMMON) for the failure reason.

P1-11 **MESSAGE: SysMgr Communication Manager already activated!**

CAUSE: Only one copy of COMMMGR is allowed.

ACTION: Do nothing.

P1-12 **MESSAGE: Network WINSOCK Error: <WINSOCK error text or error number>\n See your local network administrator for assistance.**

CAUSE: COMMMGR encountered a network problem.

ACTION: Contact your local network administrator for assistance.

P1-13 **MESSAGE: Failed to create file [<file name>]\nFileError: <error text>\nProgram will terminate!**

CAUSE: COMMMGR encountered a disk (file) problem.

ACTION: Fix the problem.

P1-14 **MESSAGE: Failed to allocate the required memory!**

CAUSE: You have run out of your PC resources. The COMMMGR program fails to allocate the buffer space to receive the incoming messages.

ACTION: Close some PC applications and restart OpenView.

P1-15 **MESSAGE: Requires an active Microsoft DBWIN.EXE window for this feature!**

CAUSE: The OVC PC software displays the tracing information via the DBWIN.EXE window, so you must have that program installed and activated.

ACTION: Install and run the DBWIN.EXE program.

P1-15 **MESSAGE: Requires an active Microsoft DBWIN.EXE window for this**

feature!

CAUSE: The OVC PC software displays the tracing information via the DBWIN.EXE window, so you must have that program installed and activated.

ACTION: Install and run the DBWIN.EXE program.

P4-01 **MESSAGE: The map object [<object name>] is unknown to the HP3000 server!**

CAUSE: The HP3000 server has not received any event information for this map symbol.

ACTION: You can manually issue a TELLOP message with the correct MNEV# marker so the HP3000 server can become aware of the existence of this map object.

P4-03 **MESSAGE: System resource is running low! .....No more event windows can be created!**

CAUSE: Current system resource is lower than the "low water mark." This parameter is defined in the C:\OV\SYSMGR\SYSMGR.INI file.

ACTION: Close some console windows or some applications to boost the resource level.

P4-04 **MESSAGE: Failed to fetch event annotation!**

CAUSE: Due to the reason specified in the error text, the required operation is not successful.

ACTION: Do nothing. Informational message.

P4-05 **MESSAGE: Failed to fetch detailed information!**

CAUSE: Due to the reason specified in the error text, the required operation is not successful.

ACTION: Do nothing. Informational message.

P4-06 **MESSAGE: Error Updating the event annotation!**

CAUSE: Due to the reason specified in the error text, the required operation is not successful.

ACTION: Do nothing. Informational message.

P4-08 **MESSAGE: Logon user [<user name>] does not have required capability! See your SysMgr administrator for assistance.**

CAUSE: User is missing "Review" capability.

ACTION: Get that capability from your OVC administrator.

P4-09 **MESSAGE: Logon user [<user name>] does not have sufficient capability to update the annotation! See your SysMgr administrator for assistance.**

CAUSE: User is missing “Ann/Del” capability.

ACTION: Get that capability from your OVC administrator.

P6-01 **MESSAGE: A Network error occurred.**

CAUSE: UsrCfg encountered an unexpected network error. This may occur due to abnormal network conditions.

ACTION: Check to see if your network is functioning properly.

P6-02 **MESSAGE: Logon user [<user name>] does not have required capability to configure users! See your SysMgr administrator for assistance.**

CAUSE: User does not have the UsrCnfg capability.

ACTION: Get that capability from your OVC administrator.

P6-03 **MESSAGE: Unexpected error happened in USRCFG.EXE.**

CAUSE: UsrCfg could not create the main dialog box. This could happen if the system is running low on resources.

ACTION: Close some applications to increase the resource level.

P6-04 **MESSAGE: Unable to print configuration data.**

CAUSE: The default printer may not be configured properly.

ACTION: Make sure that you can print to the default printer.

P6-05 **MESSAGE: Unable to create window.**

CAUSE: The creation of the window or dialog box corresponding to the command or button you selected failed. Most likely, you are running out of memory.

ACTION: Close some applications to increase the resource level.

P6-06 **MESSAGE: Menu item(s) could not be enabled.**

CAUSE: The application could not re-enable menu items that had been previously disabled. Most likely, you are running low on memory.

ACTION: Close some applications to increase the resource level.

P6-07 **MESSAGE: Cannot add <name>. Name already exists.**

CAUSE: You tried to add a user who is already configured.

ACTION: Change the name to make it unique.

P6-08 **MESSAGE: User <name> is not in the list.**

CAUSE: You tried to modify or delete users not in the list box.

ACTION: Select valid users from the list box to modify or delete them.

P6-09 **MESSAGE: Cannot remove user configuration capability for currently logged on user.**

CAUSE: You tried to remove your UserCnfg capability while using UsrCfg.

ACTION: Log off and log on again as a different user and repeat the action.

P6-10 **MESSAGE: Cannot remove currently logged on user.**

CAUSE: You tried to delete the user you were logged on as.

ACTION: Log off and log on again as a different user and repeat the action.

P6-11 **MESSAGE: Unable to add node to list box.**

CAUSE: An error was encountered while loading the UsrCfg list box. It is most likely that you are running low on memory.

ACTION: Close some applications to increase the resource level.

P6-12 **MESSAGE: Action could not be completed due to a memory error.**

CAUSE: A memory allocation error occurred.

ACTION: Restart MS-Windows.

pa-05 **MESSAGE: Logon user [<user name>] does not have sufficient capability to use MSI. See your SysMgr administrator for assistance.**

CAUSE: User did not have the Con Mtr and Con Acc capability.

ACTION: Get these capabilities from your OVC administrator.

pa-06 **MESSAGE: No computer object is selected.**

CAUSE: The system subpicture symbol you selected does not contain any computer symbol in it.

ACTION: To complete the operation, select a system subpicture symbol that has a computer symbol.

pa-07 **MESSAGE: Application is busy, wait for previous request to finish.**

CAUSE: The previous MSI request is being processed. No new requests can be processed at this time.

ACTION: Wait for the previous MSI request to be processed. This applies to the MSI button in the SysMgr Event Browser only. The user cannot use the SysMgr MSI menu item as it will be grayed out while the request is being processed.

pa-08 **MESSAGE: Config file: <config file name> does not exist.**

CAUSE: MSI could not locate the local terminal emulator configuration files.

ACTION: Make sure you have the <config file name> in the C:\OV\SYSMGR\SESSION directory.

pa-09 **MESSAGE: Could not launch <Terminal Emulator>.**

CAUSE: MSI could not find the terminal emulator specified in the [SMTerm] section in the C:\OV\SYSMGR\SYSMGR.INI file.

ACTION: Make sure the "TerminalEmulator=" entry in the [SMTerm] section in your \OV\SYSMGR\SYSMGR.INI file points to your local terminal emulator (Reflection1 for Windows).

pa-10 **MESSAGE: Operation failed (Network error), please try again.**

CAUSE: The PC encountered a network error while sending an MSI request packet to the HP 3000. This may occur under abnormal network conditions, such as the network receiving a burst of traffic.

ACTION: Check to see if the network is functioning properly. Retry the MSI operation at a later time.

pa-11 **MESSAGE: Unexpected network error <errnum> happened!**

CAUSE: MSI encountered an unexpected network error. This may occur due to abnormal network conditions.

ACTION: Check to see if the network is functioning properly.

pb-00 **MESSAGE: COMMMGR.OST is missing. Check if CommMgr is running.**

CAUSE: SysMgr Report could not find COMMMGR.OST. This file is created by CommMgr on initiating connection with the HP 3000.

ACTION: Check to see if CommMgr is running.

pb-01 **MESSAGE: Error loading from file COMMMGR.OST.**

CAUSE: File error was encountered while the COMMMGR.OST file was being accessed.

ACTION: Make sure the file is present and accessible.

pb-02 **MESSAGE: COMMMGR.TBL is missing!**

CAUSE: OV SysMgr was not installed properly.

ACTION: Copy the COMMMGR.TBL file from another PC with a proper OV Console installation, or reinstall OV Console on the PC.

pb-03 **MESSAGE: Error loading from file COMMMGR.TBL.**

CAUSE: File error was encountered while the COMMMGR.TBL file was being accessed.

ACTION: Make sure the file is present and accessible.

pb-04 **MESSAGE: Failed to create OVSEXE communication link! Program will terminate.**

CAUSE: Most probably the JCOMMON job on the HP 3000 has failed.

ACTION: Check for the existence of the JCOMMON job on the HP 3000. If it does not exist, launch it.

pb-05 **MESSAGE: Logon user [<user name>] does not have sufficient capability to generate a report. See your SysMgr administrator for assistance.**

CAUSE: User does not have the Report capability.

ACTION: OVC administrator can use the OVC "UsrCfg" module to grant this user the required capability.

pb-06 **MESSAGE: Error in getting Object List.**

CAUSE: Error encountered in getting object information on the OpenView map.

ACTION: Make sure a valid map is loaded in OpenView. If you have a valid map loaded in OpenView, try again. If this does not work, restart OpenView.

pb-07 **MESSAGE: Error in getting object selection.**

CAUSE: Error encountered while getting the selected object information from the OpenView map.

ACTION: Make sure a valid map is loaded in OpenView. If you have a valid map loaded in OpenView, try again. If this does not work, restart OpenView.

pb-08 **MESSAGE: Error in getting object Name.**

CAUSE: Error encountered while getting an object's name from the OpenView map.

ACTION: Make sure all selected objects on the current OpenView map have valid names.

pc-01 **MESSAGE: Error in MaxPort entry in Ini File. Using default of 4.**

CAUSE: There was an error in reading the "Maxport=" entry in the [Console] section of the SYSMGR.INI file.

ACTION: Check to see if the entry "MaxPort = n", where n is the maximum numbered COM port in your PC (usually 4), is present in the [Console] section of your C:\OV\SYSMGR\SYSMGR.INI file. If it is absent, add it.

pc-02 **MESSAGE: Error in ini file entry. Using default Configuration.**

CAUSE: The "COMn =<port type>" entries are incorrect in the [Console] section of the C:\OV\SYSMGR\SYSMGR.INI file.

ACTION: Check to see if the right-hand sides of these entries are one of the following types:
COMn = DISABLED
COMn = SWITCHABLE
COMn = NAILED, nodename
where n is the COM port number and nodename is the name of the HP 3000 system to which the COM port is connected.

pc-03 **MESSAGE: Map selection error.**

CAUSE: This error happens when LAN Console is unable to get the map selection from OpenView.

ACTION: Make sure you have a valid map loaded in OpenView. If the error persists, restart OpenView.

pc-04 **MESSAGE: Error in getting map object name.**

CAUSE: This error happens when LAN Console is unable to get the selected object's name from OpenView.

ACTION: Make sure the selected object on the map has a valid name.

pc-05 **MESSAGE: Selected object's name in unrecognizable format.**

CAUSE: This error happens when the computer object that was selected in the OpenView map does not conform to the NODENAME.DOMAIN.ORGANIZATION format.

ACTION: Correct the name of the selected computer object to conform to the NODENAME.DOMAIN.ORGANIZATION format.

pc-06 **MESSAGE: COMn not supported.**

CAUSE: LAN Console has encountered an unsupported COM port number.

ACTION: Make sure the "COMn=" entries in the [Console] section of the C:\OV\SYSMGR\SYSMGR.INI file are specified in such a way that n<= MaxPort entry in the same section.

pc-07 **MESSAGE: COMn is busy.**

CAUSE: This message is issued when LAN Console tries to use a COM port (because user initiated a console connection on that port) but finds that some other application has it open.

ACTION: Try using another COM port, or if you know which application has that port open, close that application and then repeat your action in LAN Console.

pc-08 **MESSAGE: Error encountered while accessing file_name. Do you want to try again?**

CAUSE: This message is issued when LAN Console tries to access one of the console script files and encounters a file error. This could be due to a corrupt disk, open drive, or an invalid path specified in the [SMTerm] section in the C:\OV\SYSMGR\SYSMGR.INI file.

ACTION: Check for any obvious disk problems. Also check to see if the entries in the [SMTerm] section in the C:\OV\SYSMGR\SYSMGR.INI file contain the correct path of the terminal emulator in the "TerminalEmulator=" entry.

pc-09 **MESSAGE: No COM port available for connection.**

CAUSE: This message is issued by LAN Console when it is not able to satisfy a user's request for a console connection due to unavailability of an appropriate COM port.

ACTION: Free COM ports by closing existing console connections.

Error Message Log Files

The following log files contain error messages for the HP 3000:

OVSLOG00.COMMON.SYSMGR

VRCLOG00.EMS.SYSMGR

Event Message Errors

The following Event Message error messages may be displayed in the SysMgr Event Browser while you are running OVC.

EM005 **MESSAGE: SysMgr-I- Workstation user interface link error. (EM005)**

CAUSE: An unexpected NetIPC 64 or 67 error was detected. Most likely the workstation was rebooted.

ACTION: Do nothing.

EM006 **MESSAGE: SysMgr-W- Workstation user interface link failed. (EM006)**

CAUSE: An unexpected NetIPC error was detected.

ACTION: View the annotation text and take the appropriate action.

EM010 **MESSAGE: SysMgr-W- STREAM EMG00JOB error <MPE error message>. (EM010)**

CAUSE: The EMS failed to stream EMG.

ACTION: Log on into the SYSMGR account on the HP 3000 and issue a STREAM EMG00JOB.EMS.SYSMGR directly at the MPE prompt. This will manually start up EMG. By doing this you will see the MPE failure that caused the error message.

EM011 **MESSAGE: SysMgr-W- EML report generation failed. (EM011)**

CAUSE: EML Report failed because of a problem on the HP 3000.

ACTION: Check the annotation text for a more detailed error message.

EM016 **MESSAGE: SysMgr-C- Unexpected internal process terminated. (EM016)**

CAUSE: Possible file system or system resource problem.

ACTION: Shut down SysMgr using OVSTOP. Read the log file VRCLOG00.SYSMGR.SYSMGR on the HP 3000 to tell you the job number that failed. Read that spool file on the HP 3000 to find out the reason for the error. Contact your system manager or your HP representative.

Index

Index