CiscoWorks Blue Maps and SNA View Mainframe Installation and Administration Guide

Release 3.0.1

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number: DOC-7811351=
Text Part Number: 78-11349-02

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with Cisco's installation instructions, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

Modifying the equipment without Cisco's written authorization may result in the equipment no longer complying with FCC requirements for Class A or Class B digital devices. In that event, your right to use the equipment may be limited by FCC regulations, and you may be required to correct any interference to radio or television communications at your own expense.

You can determine whether your equipment is causing interference by turning it off. If the interference stops, it was probably caused by the Cisco equipment or one of its peripheral devices. If the equipment causes interference to radio or television reception, try to correct the interference by using one or more of the following measures:

• Turn the television or radio antenna until the interference stops.

• Move the equipment to one side or the other of the television or radio.

• Move the equipment farther away from the television or radio.

• Plug the equipment into an outlet that is on a different circuit from the television or radio. (That is, make certain the equipment and the television or radio are on circuits controlled by different circuit breakers or fuses.)

Modifications to this product not authorized by Cisco Systems, Inc. could void the FCC approval and negate your authority to operate the product.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CONTENTS

About This Guide

This section describes the objectives, audience, organization, and conventions of the *CiscoWorks Blue Maps and SNA View Mainframe Installation and Administration Guide*. It also refers to related publications and other sources of information.

Document Contents

This guide contains information about the following topics:

- Introducing the CiscoWorks Blue Maps and SNA View Mainframe application
- Installing and configuring the mainframe application
- Updating VTAM and NetView or SOLVE:Netmaster
- Using the mainframe application
- Troubleshooting the mainframe application

For details not covered in this guide, and for last-minute changes and caveats, see the *Release Notes for CiscoWorks Blue Maps and SNA View Software Release 3.0.1*.

Audience

This guide is for the IBM mainframe system administrator or network technical support personnel who installs and configures the CiscoWorks Blue Maps and SNA View mainframe application.

The reader should be familiar with the following topics:

- Multiple Virtual Storage (MVS)
- Systems Network Architecture (SNA)
- Virtual Telecommunications Access Method (VTAM)
- Time Sharing Option Extensions (TSO/E)
- NetView or SOLVE:Netmaster
- Logical Unit 6.2 (LU 6.2) protocols (if used for connection to the workstation)
- Transmission Control Protocol /Internet Protocol (TCP/IP) stack on MVS (if used for connection to the workstation)

Document Organization

This guide contains the following chapters:

- Chapter 1, “Introducing the CiscoWorks Blue Maps and SNA View Mainframe Applications,” introduces the functions and commands of the mainframe application.
- Chapter 2, “Preparing to Install the Mainframe Application,” describes how to prepare to install the mainframe application.
- Chapter 3, “Installing the Mainframe Application,” describes the system requirements for the mainframe application, how to load mainframe data sets from the distribution tape, and additional steps to perform before the workstation can receive SNA management data from VTAM.
- Chapter 4, “Updating the Mainframe Application Software,” explains how to configure the mainframe application to use LU 6.2 or TCP/IP connectivity, perform MVS and VTAM updates, customize parameter cards, and update NetView or SOLVE:Netmaster.

- Chapter 5, “Using the Mainframe Application,” describes how to start and stop the mainframe tasks and how to use the mainframe commands.
- Chapter 6, “Troubleshooting the Mainframe Application,” describes how to detect and correct problems with the mainframe application.
- Appendix A, “Installing CiscoWorks Blue in a Different Zone,” describes how to install in a zone other than the MVS zone.
- Appendix B, “Mainframe and Workstation Installation Checklist,” is a checklist that you can use to communicate important installation information between the Maps or SNA View workstation installer and the mainframe installer.

Document Conventions

This guide uses basic conventions to represent text and table information.

Command descriptions use these conventions:

- Commands and keywords are in **boldface** font.
- Arguments for which you supply values are in *italic* font.
- Elements in square brackets ([ ]) are optional. Alternative but optional keywords are grouped in brackets ([ ]) and separated by a vertical bar (|).
- Alternative but required keywords are grouped in braces ({ }) and separated by a vertical bar (|).

Examples use these conventions:

- Terminal sessions and information that the system displays are printed in a `screen` font.
- Information that you enter is in **`boldface screen`** font. Variables you enter are printed in *`italic`* font.
- Nonprinting characters, such as passwords, are shown in angle brackets (< >).
- Information that the system displays is in `screen` font, with default responses in square brackets ([ ]).

This guide also uses the following conventions:

- Menu items and button names are in **boldface** font.

- A menu item that you are to select is indicated by the following convention: Select **Administer > CiscoWorks System > Process Mgr**.
- Directories and filenames are in *italic* font.

Note Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in the manual.

Related Documentation

The CiscoWorks Blue Maps and SNA View documentation set includes the following documentation:

- *Release Notes for CiscoWorks Blue Maps and SNA View Software Release 3.0*
- *CiscoWorks Blue Maps and SNA View Workstation Installation and Administration Guide*
- *CiscoWorks Blue Maps and SNA View User Guide*
- CiscoWorks Blue Maps Online Help System
- Installation Checklist for CiscoWorks Blue Maps and SNA View for UNIX
- Quick Start Guide for CiscoWorks Blue SNA View

Users who will be using Chapter 6, "Troubleshooting the Mainframe Application," should be familiar with the following IBM publications:

- *VTAM Resource Definition Reference*
- *MVS Message Reference*
- *VTAM Customization*

The following IBM publications are recommended reading for a broader understanding of the topics covered in this guide:

- *VTAM Operation Guide*
- *VTAM Programming*
- *VTAM Programming for LU 6.2*
- *TSO/E Operator Guide*
- *NetView Installation Guide*

- *NetView Installation and Administration Guide*
- *NetView Application Programming Guide*

The following SOLVE publication is recommended for use with SOLVE:Netmaster:

- *SOLVE:Netmaster Implementation and Administration Guide*

The following Cisco publication is recommended for configuring security:

- Cisco IOS Software *Security Configuration Guide*

Documentation CD-ROM

Cisco documentation and additional literature are available in a Cisco Documentation CD-ROM package, which may have shipped with your product. The Documentation CD-ROM is updated regularly and may be more current than printed documentation. The CD-ROM package is available as a single unit or through an annual or quarterly subscription.

Registered Cisco.com users can order a single Documentation CD-ROM (product number DOC-CONDOCCD=) through the Cisco Ordering tool:

http://www.cisco.com/en/US/partner/ordering/ordering_place_order_ordering_tool_launch.html

All users can order monthly or quarterly subscriptions through the online Subscription Store:

http://www.cisco.com/go/subscription

Ordering Documentation

You can find instructions for ordering documentation at this URL:

http://www.cisco.com/univercd/cc/td/doc/es_inpck/pdi.htm

You can order Cisco documentation in these ways:

- Registered Cisco.com users (Cisco direct customers) can order Cisco product documentation from the Networking Products MarketPlace:

  http://www.cisco.com/en/US/partner/ordering/index.shtml

- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco Systems Corporate Headquarters (California, U.S.A.) at 408 526-7208 or, elsewhere in North America, by calling 800 553-NETS (6387).

Documentation Feedback

You can submit comments electronically on Cisco.com. On the Cisco Documentation home page, click **Feedback** at the top of the page.

You can e-mail your comments to bug-doc@cisco.com.

You can submit comments by using the response card (if present) behind the front cover of your document or by writing to the following address:

Cisco Systems
Attn: Customer Document Ordering
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Obtaining Technical Assistance

Cisco provides Cisco.com, which includes the Cisco Technical Assistance Center (TAC) website, as a starting point for all technical assistance. Customers and partners can obtain online documentation, troubleshooting tips, and sample configurations from the Cisco TAC website. Cisco.com registered users have complete access to the technical support resources on the Cisco TAC website, including TAC tools and utilities.

Cisco.com

Cisco.com offers a suite of interactive, networked services that let you access Cisco information, networking solutions, services, programs, and resources at any time, from anywhere in the world.

Cisco.com provides a broad range of features and services to help you with these tasks:

- Streamline business processes and improve productivity
- Resolve technical issues with online support
- Download and test software packages
- Order Cisco learning materials and merchandise
- Register for online skill assessment, training, and certification programs

To obtain customized information and service, you can self-register on Cisco.com at this URL:

http://tools.cisco.com/RPF/register/register.do

Technical Assistance Center

The Cisco TAC is available to all customers who need technical assistance with a Cisco product, technology, or solution. Two types of support are available: the Cisco TAC website and the Cisco TAC Escalation Center. The type of support that you choose depends on the priority of the problem and the conditions stated in service contracts, when applicable.

We categorize Cisco TAC inquiries according to urgency:

- Priority level 4 (P4)—You need information or assistance concerning Cisco product capabilities, product installation, or basic product configuration. There is little or no impact to your business operations.
- Priority level 3 (P3)—Operational performance of the network is impaired, but most business operations remain functional. You and Cisco are willing to commit resources during normal business hours to restore service to satisfactory levels.
- Priority level 2 (P2)—Operation of an existing network is severely degraded, or significant aspects of your business operations are negatively impacted by inadequate performance of Cisco products. You and Cisco will commit full-time resources during normal business hours to resolve the situation.
- Priority level 1 (P1)—An existing network is “down,” or there is a critical impact to your business operations. You and Cisco will commit all necessary resources around the clock to resolve the situation.

Cisco TAC Website

The Cisco TAC website provides online documents and tools to help troubleshoot and resolve technical issues with Cisco products and technologies. To access the Cisco TAC website, go to this URL:

http://www.cisco.com/tac

All customers, partners, and resellers who have a valid Cisco service contract have complete access to the technical support resources on the Cisco TAC website. Some services on the Cisco TAC website require a Cisco.com login ID and password. If you have a valid service contract but do not have a login ID or password, go to this URL to register:

http://tools.cisco.com/RPF/register/register.do

If you are a Cisco.com registered user, and you cannot resolve your technical issues by using the Cisco TAC website, you can open a case online at this URL:

http://www.cisco.com/tac/caseopen

If you have Internet access, we recommend that you open P3 and P4 cases online so that you can fully describe the situation and attach any necessary files.

Cisco TAC Escalation Center

The Cisco TAC Escalation Center addresses priority level 1 or priority level 2 issues. These classifications are assigned when severe network degradation significantly impacts business operations. When you contact the TAC Escalation Center with a P1 or P2 problem, a Cisco TAC engineer automatically opens a case.

To obtain a directory of toll-free Cisco TAC telephone numbers for your country, go to this URL:

http://www.cisco.com/warp/public/687/Directory/DirTAC.shtml

Before calling, please check with your network operations center to determine the Cisco support services to which your company is entitled: for example, SMARTnet, SMARTnet Onsite, or Network Supported Accounts (NSA). When you call the center, please have available your service agreement number and your product serial number.

Obtaining Additional Publications and Information

Information about Cisco products, technologies, and network solutions is available from various online and printed sources.

- The *Cisco Product Catalog* describes the networking products offered by Cisco Systems, as well as ordering and customer support services. Access the *Cisco Product Catalog* at this URL:

  http://www.cisco.com/en/US/products/products_catalog_links_launch.html

- Cisco Press publishes a wide range of networking publications. Cisco suggests these titles for new and experienced users: *Internetworking Terms and Acronyms Dictionary*, *Internetworking Technology Handbook*, *Internetworking Troubleshooting Guide*, and the *Internetworking Design Guide*. For current Cisco Press titles and other information, go to Cisco Press online at this URL:

  http://www.ciscopress.com.

You can access the CiscoWorks Blue web page using the following URL:

http://www.cisco.com/go/cwblue

Note If you are a network administrator and need personal technical assistance with a Cisco product that is under warranty or covered by a maintenance contract, contact Cisco's Technical Assistance Center (TAC) at 800 553-2447, 408 526-7209, or tac@cisco.com. To obtain general information about Cisco Systems, Cisco products, or upgrades, contact 800 553-6387, 408 526-7208, or cs-rep@cisco.com.

Introducing the CiscoWorks Blue Maps and SNA View Mainframe Applications

This chapter introduces the features of the mainframe application. This chapter includes the following main sections:

- What is the CiscoWorks Blue Mainframe Application?, page 1-1
- What SNA Functions are Provided?, page 1-2
- The mainframe application includes a series of subtasks that run on the mainframe. These subtasks provide connections to the workstations and PU/LU discovery and monitoring. For more information, see the “Mainframe Subtasks” section on page 5-2 in Chapter 5, “Using the Mainframe Application.”, page 1-3

What is the CiscoWorks Blue Mainframe Application?

The CiscoWorks Blue mainframe application acts solely as a server to provide Systems Network Architecture (SNA) resource information to the CiscoWorks Blue Maps and SNA View applications on workstations. There are no end-user functions provided.

For information about how the SNA information is used at the workstation, see the *CiscoWorks Blue Maps and SNA View Workstation Installation and Administration Guide* or the *CiscoWorks Blue Maps and SNA View User Guide*.

What SNA Functions are Provided?

The CiscoWorks Blue Maps and SNA View products obtain SNA physical unit (PU) and logical unit (LU) resource information from VTAM at a mainframe computer and correlate that information with the Cisco routers that are associated with the PUs and LUs. Maps and SNA View consist of two components: a Multiple Virtual Storage (MVS) system mainframe application and a UNIX workstation application (for Maps and SNA View).

The mainframe component provides the SNA PU and LU information to the workstation program. It runs as a started task under OS/390 (MVS/Enterprise Systems Architecture).

The workstation component obtains the PU and LU information from the mainframe and stores it in a database at the workstation.

Note PUs connected via the remote source route bridging (RSRB) protocol and a CIP or 3172 connection to the mainframe will not benefit from the correlation feature. This means that the Maps and SNA View applications cannot determine on which routers these PUs are dependent for connectivity to the mainframe. However, the IBM APAR OW36070 is available that allows Maps and SNA View to support switched PUs using these types of connections.

The workstation program provides the user with PU and LU information. To provide these services to the workstation user, the mainframe application uses the interfaces described in the following sections.

VTAM XID Configuration Services Exit

When you install the mainframe application, you get a fully functional Virtual Telecommunications Access Method (VTAM) exchange identification (XID) configuration services exit routine. For more information, see the "Updating MVS and VTAM" section on page 4-9 in Chapter 4, "Updating the Mainframe Application Software."

Mainframe Parameter Cards

The mainframe application provides a set of initialization parameter cards with which you customize the mainframe application. For more information, see the "Updating the Mainframe Configuration File (NSPPARM)" section on page 4-21 in the Chapter 4, "Updating the Mainframe Application Software."

Mainframe Commands

The mainframe application provides a set of mainframe commands with which you configure and monitor the mainframe environment. You can start, stop, and display the status of the mainframe subtasks. For more information, see the "Issuing Mainframe Commands" section on page 5-3 in Chapter 5, "Using the Mainframe Application."

Mainframe Subtasks

The mainframe application includes a series of subtasks that run on the mainframe. These subtasks provide connections to the workstations and PU/LU discovery and monitoring. For more information, see the "Mainframe Subtasks" section on page 5-2 in Chapter 5, "Using the Mainframe Application."

CHAPTER 2

Preparing to Install the Mainframe Application

This chapter provides information on preparing to install the CiscoWorks Blue Maps and SNA View mainframe component. Before you install, configure, and validate this product, read the following sections in this chapter:

Mainframe Hardware Requirements

Your mainframe system must meet the following hardware requirements before you can install the mainframe application:

- The CD-ROM on your PC must have the ability to ftp data sets from the CD-ROM to the host system.
- Disk space—Approximately 50 cylinders direct access storage.

Mainframe Software Requirements

Your mainframe system must have the following mainframe software before you can use with the mainframe application:

- Operating system—MVS/ESA (OS/390) 5.5.2 or later, plus SMP/E, Release 7 or later
- Access method—VTAM 4.1 or later
- Protocol stack—One of the following communication protocols:
  - SNA LU 6.2
  - TCP/IP (Cisco IOS for S/390, IBM TCP/IP for MVS Version 3, Release 1 or later, or Interlink TCP/IP for MVS)

  If both TCP/IP and LU 6.2 are available, Cisco recommends using TCP/IP.

Workstation Software

All CiscoWorks Blue Maps or SNA View workstations must have either TCP/IP or LU 6.2 connectivity to the mainframe. TCP/IP connectivity is provided through the workstation operating system. LU 6.2 connectivity is provided by one of the following products:

- On AIX systems—IBM eNetwork Communications Server for AIX, Version 5
- On HP-UX systems—HP-UX SNAplus2, including SNAplus2 Link and SNAplus2 API
- On Solaris systems—Data Connection SNAP-IX V6.0.6 for SPARC Solaris (for information see the URL www.datacon.co.uk).

Optional Mainframe Software

The following mainframe software is optional:

- Network management system (NetView 2.3 or later, or SOLVE:Netmaster 2.2 or later). Use TME 10 NetView for OS/390 V1R2 if you want to connect the CiscoWorks Blue web server to NetView.
- Network Control Program (NCP) 4.3 or later if you use RSRB Maps and SNA View PU and LU correlation

Mainframe Installation Summary

This section summarizes the steps to follow for installing the mainframe application and to update the other mainframe software. Use this summary to ensure that you have performed all the mainframe installation tasks.

Step 1 Install the mainframe files from the CD-ROM.

Step 2 Configure connectivity from the workstation to the mainframe.

Step 3 Update MVS and VTAM. Change the MVS and VTAM system software.

Step 4 Update the Configuration Services XID exit routine.

Step 5 Do not filter out the following messages:

- IST093I
- IST619I
- IST590I
- IST1132I
- IST1416I
- IST1136I
- IST259I
- IST105I
- IST621I
- IST1133I

- IST1135I
- NSP2000I

Step 6 Update the NSPPARM configuration file by customizing the mainframe parameter cards.

Step 7 Update either TME:10 Netview for OS/390 (NetView) or SOLVE: Netmaster (Netmaster).

Installing the Mainframe Application

This chapter provides information on installing the mainframe application directly from the CD or using the System Modification Program Extended (SMP/E), Release 7 or later. The CD contains all the materials required to install the mainframe application.

This chapter includes the following main sections:

Using CiscoWorks Blue Release 3.0.1 Workstation Software with a Previous SNA View Release

If your workstation is running CiscoWorks Blue Release 3.0.1, the mainframe must be running CiscoWorks Blue of at least Release 2.1. Cisco strongly recommends that you upgrade both the workstation and mainframe to CiscoWorks Blue 3.0.1.

Installing the Mainframe Application

This section describes how to install the mainframe application. Two methods for installing the mainframe application from the CD-ROM are provided and are in the following subsections:

- Installing the Mainframe Application Directly, page 3-2.

  This method is the simplest installation method. Use this method if you are *not* going to use SMP/E to install the mainframe software.

- Installing the Mainframe Application Using SMP/E, page 3-4.

  Use this method if you are going to install the mainframe software using SMP/E.

Installing the Mainframe Application Directly

If you choose to directly install the application, use the procedures in the following subsections:

- Copying the Files from the Workstation to the Mainframe, page 3-2
- Receiving the Data Sets on the Mainframe, page 3-3

Copying the Files from the Workstation to the Mainframe

Use the following procedure to copy the required files from your workstation to the mainframe:

Step 1 Change the directory to the mount point for the CD-ROM.

Step 2 Select the **mainframe** subdirectory (cd mainframe) by entering the following command:

```
cd mainframe
```

Step 3 Select the **direct** subdirectory (cd direct).

Step 4 Using a valid user name and password, use ftp to transfer files to the mainframe.

Step 5 Enter **ftp** *target host* **> bin** for binary mode transfer.

Step 6 Enter the following commands to set the database attributes necessary for starting the data sets:

```
quote site lrecl=80 RECFM=FB CYLINDERS PRIMARY=1 SECONDARY=1
put NSPSCLST.SEQ
put NSPSLOAD.SEQ
put NSPSSAMP.SEQ
```

The three files now exist on the mainframe as the following sequential data sets:

- user.NSPSCLST.SEQ
- user.NSPSLOAD.SEQ
- user.NSPSSAMP.SEQ

Continue with the "Receiving the Data Sets on the Mainframe" section on page 3-3.

Receiving the Data Sets on the Mainframe

Use the following procedure to receive the data sets on the mainframe:

Step 1 For the following files, enter **RECEIVE INDS(/)** next to the name of the data set in the ISPF data set list.

RECEIVE INDS(/)

- user.NSPSCLST.SEQ
- user.NSPSLOAD.SEQ
- user.NSPSSAMP.SEQ

Step 2 Press **Enter** to receive the data sets as the following:

- user.NSPSCLST
- user.NSPSLOAD
- user.NSPSSAMP

Note You may have to press **Enter** more than once to receive the data sets and to clear the informational messages.

Step 3 Copy and then rename the data sets as desired. Rename the data sets to name that matches the values specified on the NSPOPEN procedure.

You have completed loading the mainframe application files from the CD-ROM. The next step is to configure the mainframe application software using the information in Chapter 4, "Updating the Mainframe Application Software."

Installing the Mainframe Application Using SMP/E

If you choose to install the mainframe application using SMP/E, use the following procedure:

Note All the files required to install the mainframe application are in the EBCDIC format and are located on the CD-ROM. To avoid character translation, perform all file transfers in binary mode. The installation libraries have been converted to the XMIT format using the TSO transmit facility.

Step 1 Change the directory to the mount point for the CD-ROM.

Step 2 Select the **mainframe** subdirectory (**cd mainframe**) by enetering the following command:

```
cd mainframe
```

Step 3 Select the **smpe** subdirectory (cd smpe).

Step 4 Using a valid user name and password, use FTP to transfer files to the mainframe.

Step 5 Enter **ftp** *target host* > **bin** for binary mode transfer.

Step 6 Use the following site options command to transfer the following six files to the MVS host.

quote site lrecl=80 RECFM=FB CYLINDERS PRIMARY=1 SECONDARY=2

These files should be located on the CD-ROM.

- NSPS301.F1.SEQ
- NSPS301.F2.SEQ
- NSPS301.F3.SEQ
- NSPS301.F4.SEQ
- NSPS301.NSPSINST.SEQ
- NSPS301.SMPMCS.SEQ

Remember to use binary mode and to set the MVS host FTP SITE options appropriately.

The following is a sample FTP session to send the files to the MVS host:

```
ftp mvshost.com
user:ibmuser
password: xxxxxxxx
bin

quote site lrecl=80 RECFM=FB CYLINDERS PRIMARY=1 SECONDARY=1
put NSPS301.F1.SEQ
put NSPS301.F2.SEQ
put NSPS301.F3.SEQ
put NSPS301.F4.SEQ
put NSPS301.NSPSINST.SEQ
put NSPS301.SMPMCS.SEQ
```

Step 7 The **ftp** command transfers files in sequence. After you transfer the files to the host, use TSO option 3.4 to list the six data sets. Your list should resemble the following list, except for the *userid* and *volume*:

```
DSLIST - Data Sets Matching xxxx.NSPS301      Row 1 of 6
Command ===>Scroll ===> PAGE
Command - Enter "/" to select action Message Volume
-------------------------------------------------------------------
xxxx.NSPS301.F1.SEQ                          SMSC1F
xxxx.NSPS301.F2.SEQ                          SMSC1F
xxxx.NSPS301.F3.SEQ                          SMSC1F
xxxx.NSPS301.F4.SEQ                          SMSC1F
xxxx.NSPS301.NSPSINST.SEQ                    SMSC1F
xxxx.NSPS301.SMPMCS.SEQ                      SMSC1F
*****************End of Data Set list ******************
```

Step 8 The TSO receive option can convert the sequential data sets you sent to the host into PDS format. You do not need to pre-allocate the data sets. From the displayed list, enter the following on the first line, as shown below:

RECEIVE INDS(/)

```
DSLIST - Data Sets Matching xxxx.NSPS301 Row 1 of 6
Command ===>Scroll ===> PAGE
Command - Enter "/" to select action Message Volume
-------------------------------------------------------------------
RECEIVE INDS(/) PS300.F1.SEQ                 SMSC1F
xxxx.NSPS301.F2.SEQ                          SMSC1F
xxxx.NSPS301.F3.SEQ                          SMSC1F
xxxx.NSPS301.F4.SEQ                          SMSC1F
xxxx.NSPS301.NSPSINST.SEQ                    SMSC1F
xxxx.NSPS301.SMPMCS.SEQ                      SMSC1F
*****************End of Data Set list ******************
```

Step 9 Tab to the next line, enter = and press **Enter**. Repeat this step until all data sets are restored.

Note You might have to press **Enter** more than once to receive the data sets and to clear the informational messages.

Step 10 Once all data sets are restored, press **PF3** and **Enter** to list all 12 data sets. The list contains the six sequential data sets and six PDS files.

```
DSLIST - Data Sets Matching xxxx.NSPS301      Row 1 of 6
Command ===>Scroll ===> PAGE
Command - Enter "/" to select action Message Volume
-------------------------------------------------------------
xxxx.NSPS301.F1                               SMSC1F
xxxx.NSPS301.F1.SEQ                           SMSC1F
xxxx.NSPS301.F2                               SMSC1F
xxxx.NSPS301.F2.SEQ                           SMSC1F
xxxx.NSPS301.F3.                              SMSC1F
xxxx.NSPS301.F3.SEQ                           SMSC1F
xxxx.NSPS301.F4                               SMSC1F
xxxx.NSPS301.F4.SEQ                           SMSC1F
xxxx.NSPS301.NSPSINST                         SMSC1F
xxxx.NSPS301.NSPSINST.SEQ                     SMSC1F
xxxx.NSPS301.SMPMCS                           SMSC1F
xxxx.NSPS301.SMPMCS.SEQ                       SMSC1F
*****************End of Data Set list *******************
```

Step 11 Use the sample SMP/E JCL named ALLOC, in *prefix.NSPS301.NSPSINST*, to allocate your target and distribution libraries.

Step 12 Use the sample SMP/E JCL name NSPSMPA in *prefix.NSPS301.NSPINST*, to allocate the SMP/E temporary Libraries needed if setting up a new global zone for SNAVIEW installation and maintenance.

Step 13 Use the sample SMP/E JCL name NSPCSI in *prefix.NSPS301.NSPINST*, to allocate and prime the VSAM Data Sets for Global, Target, and Distribution Zone CSIS. This job should be run if you want to have a separate CSIS for the target and distribution zones.

Step 14 Use the sample SMP/E JCL named RECNSP, in *prefix.NSPS301.NSPSINST*, for example, SMP/E receive job.

Step 15 Use the sample SMP/E JCL named APPNSP, in *prefix.NSPS301.NSPSINST*, for example, SMP/E apply job.

Step 16 Use the sample SMP/E JCL named ACCNSP, in *prefix.NSPS301.NSPSINST*, for example, SMP/E accept job.

You have completed loading the mainframe application files from the CD-ROM. The next step is to configure the mainframe application software using the information in Chapter 4, “Updating the Mainframe Application Software.”

CHAPTER 4

Updating the Mainframe Application Software

This chapter provides instructions for updating VTAM and MVS on the mainframe, as well as updating the input parameter cards to customize the mainframe application for your site's particular needs. You will use different sections of this chapter depending on whether your connection to the workstation uses LU 6.2 or TCP/IP.

This chapter includes the following main sections:

Use the installation checklist provided in Appendix B, "Mainframe and Workstation Installation Checklist," to coordinate the configuration of the mainframe and workstation components.

Configuring Connectivity

You must configure the mainframe to communicate with the CiscoWorks Blue workstation using either LU 6.2 or TCP/IP. See the section that pertains to your protocol.

- Configuring LU 6.2 Connectivity, page 4-2
- Configuring TCP/IP Connectivity, page 4-4

Configuring LU 6.2 Connectivity

This section describes how to modify VTAM data sets on a mainframe that is connected to the workstation using LU 6.2.

Before You Configure Connectivity

Before you start, you must configure the mainframe to allow an LU 6.2 session to flow from the workstation to the mainframe. You may need to change VTAM and workstation applications that support LU 6.2 sessions (for example, SNAplus2 or Communication Server for AIX). If the workstation is not directly connected to the mainframe running the mainframe application, but the session passes through one or more VTAMs before reaching the destination VTAM, the correct configuration might require changes to all VTAMs (and possibly the Network Control Programs [NCPs]) in the path. It is not the intent of this book to document all the steps necessary to set up the network. See the relevant VTAM and NCP publications.

If this LU 6.2 setup has not yet been done, delay the installation until the LU 6.2 configuration is complete. One way to determine whether there is LU 6.2 connectivity between the workstation and the mainframe is to issue the following VTAM command:

D NET,APING,ID=*NETID.RESOURCE*

Where the *NETID.RESOURCE* is the fully qualified name of the SNA workstation. Until the **APING** command returns a positive response, the mainframe application cannot connect to the workstation.

After the initial LU 6.2 configuration is complete, use the information in the following section, "Configuring LU 6.2 Connectivity" to complete the configuration.

Configuring LU 6.2 Connectivity

To configure LU 6.2 connectivity, perform the following steps:

Step 1 Ensure that your MODETAB table entry contains the MODENT entries (SNASVCMG and PARALLEL) shown in the following example.

LU 6.2 uses the following LOGMODE entries:

```
SNASVCMG MODEENT LOGMODE=SNASVCMG,FMPROF=X'13',TSPROF=X'07',  X
PRIPROT=X'B0',SECPROT=X'B0',COMPROT=X'D0B1',        X
RUSIZES=X'8585',PSERVIC=X'060200000000000000000300', X
ENCR=B'0000'
*
PARALLEL MODEENT LOGMODE=PARALLEL,FMPROF=X'13',TSPROF=X'07',X
PRIPROT=X'B0',SECPROT=X'B0',COMPROT=X'50B1',TYPE=X'00',     X
RUSIZES=X'8787',PSERVIC=X'06020000000000000002F00'
*
```

If your MODETAB table lacks these entries, use a text editor to add them before you reassemble and link-edit the MODETAB table. The text for these table entries is available in *prefix.NSPS301.NSPSSAMP(MODEENT)*. A sample of assembly and link-edit JCL is available in *prefix.NSPS301.NSPSSAMP(MODEJCL)*.

Note If you use Parallel Mode, the RUSIZE values must be at least hex 8787.

The changes to the MODETAB table take effect when VTAM is restarted. You can load the changes immediately with the following system console command:

MODIFY NET,TABLE,NEWTAB=modetab,OPTION=LOAD

Step 2 Create a PU definition for the workstation that meets the following requirements:

- The PU is defined under a major node type that accepts independent LUs.
- The PU is defined with a MODETAB featuring the MODEENT entries listed in Step 1.
- The PU has DYNLU=YES coded.
- The PU has PUTYPE=2 coded.

Additionally, if the PU is defined under an NCP major node, the NCP definition must contain the LUDRPOOL statement for the configuration of at least three independent LUs.

A sample PU definition (defined under a switched major node) is available in *prefix.NSPS301.NSPSSAMP(SWMNILU)*.

Step 3 Define an independent LU under a cross-domain resource (CDRSC) major node, associating the LU with an existing PU. A sample CDRSC definition is available in *prefix.NSPS301.NSPSSAMP(NSPCDRSC)*. Alternatively, you can define an independent LU under an existing PU definition by coding LOCADDR=0.

You can modify the resource names in the sample major node members to your site's naming conventions for network resources, but the same modifications must be made to the parameter cards (See the "Updating the Mainframe Configuration File (NSPPARM)" section on page 4-21).

Configuring TCP/IP Connectivity

This section describes how to modify your mainframe TCP/IP installation. TCP/IP connects through Cisco IOS for S/390, Interlink TCP/IP, or IBM TCP/IP.

Configuring the Mainframe TCP/IP Stacks

Use of Cisco IOS for S/390 or Interlink TCP/IP for MVS

To use either of these TCP/IP stacks in MVS, make the following changes to the NSPOPEN procedure (located in *prefix.NSPS301.NSPSSAMP*) or JCL.

- Add the Cisco IOS for S/390 or Interlink TCP/IP for MVS link library, and then add the Cisco IOS for S/390 or Interlink TCP/IP load library to the STEPLIB and NSPCLIB ahead of the SNA View load library.

  This task is necessary because SAS/C libraries are shipped in the SNA View load library, and the Cisco IOS stack and Interlink stack ship their own SAS/C copy of the LSCNCOM module, which replaces the copy shipped with the SAS/C library.

- If you want the NSPOPEN proc to use the Cisco IOS stack or an Interlink stack that does **not** use the default subsystem name of ACSS, you will need to apply a fix to the NSPSLOAD data set. You can retrieve the fix using the Cisco CCO. Contact the Cisco TAC for details.

  If you are using the Interlink stack, you must also supply the 4-character name of the Interlink stack in the NSPOPEN EXEC card in the NSPOPEN proc:

  ```
  //NSPOPEN EXEC PGM=NSPOPEN,PARM='=ICS_SUBSYS=SUBS',
  //        TIME=1440,REGION=&RGN,
  ```

  Where *SUBS* is the 4-character subsystem name for the Interlink TCP/IP stack.

- If you are using Cisco IOS 390 Release 1 or TCP Access 4.1 (or earlier), and you are not using the default subsystem name of ACSS, you must apply PTF TP06511 from Interlink/Sterling. PTF TP06511 requires the following other PTFs:

  – TP04545

  – TP04823

  – TP06256

  – TP06308

  You can obtain all these PTFs from *ftp.interlink.com/pub/ptf410*. PTFs TP04545 and TP04823 are in the 9712 Cumulative.

If you are using Cisco IOS 390 Release 2, or TCP Access 5.2, you do not need these PTFs.

PTFs Required for the Mainframe TCP/IP Stacks

If you are using Cisco IOS 390 Release 1 or TCP Access 4.1 (or earlier), and you are not using the default subsystem name of ACSS, you must apply PTF TP06511 from Interlink/Sterling. PTF TP06511 requires the following other PTFs:

- TP04545
- TP04823
- TP06256
- TP06308

You can obtain all these PTFs from *ftp.interlink.com/pub/ptf410*. PTFs TP04545 and TP04823 are in the 9712 Cumulative.

If you are using Cisco IOS 390 Release 2, or TCP Access 5.2, you do not need these PTFs.

Configuring IBM TCP/IP Connectivity for MVS

Complete the following steps to configure systems that use IBM TCP/IP for MVS.

Step 1 Reserve port numbers in the *PROFILE.TCPIP* file.

This step is optional. If you reserve specific port numbers, this reservation will flag these port numbers for exclusive use by the mainframe application. Consequently, other products on the mainframe will not use them.

For each workstation, choose two consecutive, available port numbers and add the following two lines to the list of PORT values in your *PROFILE.TCPIP* file. The default port values are 6104 and 6105, as shown in the following example TCP lines. If you use other port numbers, change the TCP lines in the *PROFILE.TCPIP* file.

```
6104 TCP NSPOPEN
6105 TCP NSPOPEN
```

Note The TCP port numbers on the mainframe must match the port numbers on the workstation.

Step 2 If the TCP/IP address space is not named TCPIP, edit the *TCPIP.TCPIP.DATA* data set, and verify that the TCPIPJOBNAME or TCPIPUSERID parameter is set correctly to the name of the TCP/IP address space.

Step 3 Ensure that the TCPIP prefix is set correctly in the NSPOPEN procedure (located in *prefix*.*NSPS301*.*NSPSSAMP*) by using one of the following procedures:

- Use SYSTCPD if the data file is a member of a partitioned data set, but is not a partitioned data set itself:

  ```
  //SYSTCPD DD DISP=SHR,DSN=HLQ.NAME(TCPDATA)
  ```

- Change the characters TCPHLL in PARM='=TCPIP_PREFIX=TCPHLL', to your installation-defined high-level qualifier for TCP/IP. For example, if the profile DD statement in the TCP/IP PROC is TCPMVS2.TCPIP1.PROFILE.TCPIP, then the high-level qualifier is TCPMVS2.TCPIP1 and the PARM would be specified as follows:

  ```
  PARM='=TCPIP_PREFIX=TCPMVS2.TCPIP1',
  ```

Step 4 If you are running IBM TCP/IP V3R4 or later, edit the *NSPPARM* parameter file and change the TCP_LEVEL card to **IV3R4**, as described in the "TCP_LEVEL" section on page 4-35.

Step 5 After you make any changes, restart the mainframe application.

Step 6 Verify the TCP/IP connectivity for MVS.

OS/390 Changes

Any TCP/IP client application that is running on OS/390 V2R5 or later, or on TCPIP V3R3, must be defined to the security manager (RACF) with an OMVS segment. If you start the Maps or SNA View mainframe application from the console, first you must add an entry for the procedure in the STARTED class with an Open MVS (OMVS) segment. This segment must assign the OMVS group and userid to the Maps or SNA View mainframe application to connect as a client to TCP/IP OpenEdition.

If you do not add the entry, the following message results:

```
NSP150  TCP/IP communications: socket() for workstation message agent
failed with error 14
```

The mainframe application also fails if ports are defined in the *PARM* data set reserved for BINDs to port 0 in the applicable *BPXPRMxx* member in the operating system *PARMLIB*. The following example shows where the ports are defined:

```
NETWORK DOMAINNAME(AF_INET) DOMAINNUMBER(2) MAXSOCKETS(1000)
TYPE(CINET)  INADDRANYPORT(5000)  INADDRANYCOUNT(4000)
```

The second (TYPE) statement reserves ports 5000 through 8999 for BINDs to port 0 and cannot be used in the mainframe application's *PARM* data set. An attempt to bind to a port in this range will result in the following message:

```
NSP150  TCP/IP communications: bind() for workstation message agent
failed with error 37
```

Because CiscoWorks Blue defaults to use ports 6104 and 6105 for communication with the workstation, you should free those ports. One way to do this is by changing the INADDRANYCOUNT value (in the previous example) to 1000, as follows:

```
NETWORK DOMAINNAME(AF_INET) DOMAINNUMBER(2) MAXSOCKETS(1000)
TYPE(CINET)  INADDRANYPORT(5000)  INADDRANYCOUNT(1000)
```

Or, you can choose other available ports. If you choose other ports, make the similar port changes at the workstation.

Configuring TCP/IP Connectivity to Multiple Domains

If you configure TCP/IP connectivity from multiple UNIX workstations, each set of SVMF_HCI_AGENT_PORT and SVMF_CMDS_AGENT_PORT parameters for each mainframe connection must have corresponding TCP parameter cards in the mainframe. (TCP parameter cards are described in the "Updating the Mainframe Configuration File (NSPPARM)" section on page 4-21.) For example, set the workstation parameters on workstation A as follows in the file */etc/svopen_config_DOMAIN*:

```
SVMF_HCI_AGENT_PORT 6104
SVMF_CMDS_AGENT_PORT 6105
```

You would set the workstation parameters on workstation B as follows in the file */etc/svopen_config_DOMAIN*:

```
SVMF_HCI_AGENT_PORT 6124
SVMF_CMDS_AGENT_PORT 6125
```

The host configuration files for workstations A and B each must have a TCP card that specifies the correct port configurations:

- For workstation A:

  ```
  TCP 6104 6105
  ```

- For workstation B:

  ```
  TCP 6124 6125
  ```

Note The TCP port numbers must match the port numbers on the workstation.

Data transferred between the mainframe and workstation components is not encrypted, but probably will be secure if transferred over a private intranet. If the workstation-to-host connection traverses the Internet, or if additional security is desired over an intranet, you can use the Network Data Encryption with Router Authentication feature. This feature is provided with Cisco routers to encrypt the data that flows between the router nearest to the workstation and the router nearest to the host.

More information about encrypted connections can be found in the Cisco IOS software Security Configuration Guide.

Updating MVS and VTAM

This section describes changes that you must make in your system's MVS and VTAM data sets. These changes are necessary regardless of the method used to connect the workstation to the mainframe. Notify your system programmer of the changes that must be made to the members in the *SYS1.PARMLIB* data set.

Step 1 Authorize LOADLIB (*prefix.NSPS301.NSPSLOAD)* by adding the data set *prefix.NSPS301.NSPSLOAD* and its direct access storage device (DASD) volume name to your list of authorized program facility (APF) authorized data sets in *SYS1.PARMLIB(IEAAPFxx)* or *SYS1.PARMLIB(PROGxx)*. Doing this lets the mainframe application process some authorized commands and perform security checks.

If necessary, reload (re-IPL) MVS. If your system is set up to use dynamic APF services, you can avoid reloading MVS by using the **SETPROG** command to update the APF list dynamically. See the *Initialization and Tuning Reference* manual for your MVS/ESA system for more information about authorizing data sets.

Step 2 Set the performance group by adding a TRXNAME parameter for the Maps and SNA View mainframe application to the started task control (STC) subsystem definition of *SYS1.PARMLIB(IEAICSxx)*. In the TRXNAME line, specify the same performance group used by NetView or other high-priority application programs to ensure that the mainframe application receives enough CPU time to avoid a backlog of network information processing.

The default application name for the mainframe application startup job is **NSPOPEN**.

If NetView is running in performance group 8, and you want the mainframe application also to run in performance group 8, specify the following TRXNAME parameter:

TRXNAME=NSPOPEN,PGN=8

After you add a new entry, you can use the following MVS command to dynamically reload the installation control specification (ICS) file:

SET ICS=*xx*

Where:

xx is the two-digit suffix of the member that was edited.

Step 3 Add an entry to the program properties table in *SYS1.PARMLIB(SCHEDxx)*:

PPT PGMNAME(NSPOPEN)

NOSWAP

SYST

After you add the new entry, you can reload the program properties table immediately with the following MVS command:

SET SCH=*xx*

Where:

xx is the two-digit suffix of the member that was edited.

Step 4 Add the VTAM parameter **PPOLOG=YES** to your VTAM startup options in the *SYS1.VTAMLST(ATCSTRxx)* file to ensure that messages issued by VTAM in response to console commands are sent to the primary program operator.

If the PPOLOG parameter has not been set in the currently running VTAM, you can add it dynamically with the following command:

MODIFY vtamproc,PPOLOG=YES

Step 5 Copy and modify the *prefix.NSPS301.NSPSSAMP(NSPAPPL)* data set.

- Copy the application major node definition *prefix.NSPS301.NSPSSAMP(NSPAPPL)* into your VTAMLST library.
- Add NSPAPPL to your list of major nodes in *SYS1.VTAMLST(ATCCONxx)*.

Step 6 Activate the major node and verify that the APPL definitions are active.

You can modify the APPL resource names in the definition to suit your site's naming conventions for network resources, but the same modifications must be made to the parameter cards described in the "Updating the Mainframe Configuration File (NSPPARM)" section on page 4-21.

Step 7 Install the VTAM exit routine, and allocate the VSAM database, as described in the "Using the Configuration Services XID Exit Routine" section on page 4-12.

Using the Configuration Services XID Exit Routine

This section describes the Configuration Services exit routine and includes the following subsections:

XID Configuration Services Exit Manager

Maps and SNA View provide a functional XID Configuration Services exit manager that can invoke multiple exit routines. Two instances of the exit manager are built in the NSPSLOAD data set. One instance invokes the Maps and SNA View version of the exit and is named NSPESNAV. The other instance, named NSPESNIS, invokes both the Maps and SNA View version as well as the Internetwork Status Monitor (ISM) version of the exit. Also, you can generate your own NSPEMGR exit to invoke the Cisco-supplied exits as well as an existing exits used at your site. The exit manager lets your ISTEXCCS exit routine coexist with the CiscoWorks Blue exit routines without requiring modifications to your exit routine's source code.

If you do not have a Configuration Services exit in use (see the "Determining ISTEXCCS Exit Presence" section on page 4-13), install the Maps and SNA View version as described in the "Installing the Maps and SNA View XID Configuration Services Exit Routine" section on page 4-15.

If you have previously installed ISM, and the ISM exit is currently the only exit being handled by the exit manager (you do not have another exit that must be invoked), install the combined ISM and Maps and SNA View version as described in the "Installing the Combined ISM, Maps, and SNA View XID Configuration Services Exit Routine" section on page 4-16.

If you have your own ISTEXCCS exit routine, follow the procedures as described in the "Adding Your Own ISTEXCCS Exit Routine to NSPEMGR" section on page 4-16.

SNA View, ISM, and User ISTEXCCS Exits

When the SNA View, ISM, and user ISTEXCCS exits are combined with the CiscoWorks Blue exit manager NSPEMGR, some interdependencies exist.

- When coding the NSPECCSL csect, which controls the order of executions of the exits, the user ISTEXCCS exit should be specified first.
- The SNA View and ISM exits request, during beginning vector processing, the VTAM calls the exit for the PUs that are already defined to VTAM (PU_DEFND). If the user exit is not coded to ignore or process this request type, then problems can occur during XID processing.

SNA View Exit Logs

The Maps and SNA View exit logs MAC, SAP, and RIF data to a VSAM data set each time a switched PU connects into the network. You must allocate and prime the VSAM data sets before the exit is activated and the modified VTAM PROC is restarted, as described in the "Allocating and Defining the VSAM Data Sets for the Configuration Services Exit" section on page 4-14.

Determining ISTEXCCS Exit Presence

To determine if you have an active ISTEXCCS exit, issue the following command:

D NET,EXIT,ID=ISTEXCCS

If the command shows an active exit, you have an active ISTEXCCS exit. If it indicates an inactive exit, you might have to activate the ISTEXCCS exit. Ask your system programmer to active the ISTEXCCS exit.

If your ISTEXCCS exit is not automatically started with VTAM, you should consider configuring your VTAM program to automatically activate the ISTEXCCS exit so it will be activated when VTAM starts.

Activating an ISTEXCCS Exit

To manually activate the ISTEXCCS exit, issue the following command:

f net,exit,opt=act,id=istexccs,module=modulename

Where:

modulename is the load module name of the exit you will use.

Inactivating an ISTEXCCS Exit

If an ISTEXCCS exit is active and you want to inactivate it, issue the following command:

f net,exit,opt=inact,id=istexccs

Allocating and Defining the VSAM Data Sets for the Configuration Services Exit

This section describes how to calculate the size of the primary and backup databases in SNPDBVSM. Regardless of which exit manager you use, you must allocate and define the VSAM data sets for use by the Maps and SNA View exit.

Note All samples provided in this document must be modified before use.

Step 1 Use the sample member provided in *prefix.NSPS301.NSPSSAMP(NSPDBVSM)* to allocate two VSAM databases used by the NSPEXCCS exit routine. Use the following formula to calculate the size of both the PRIMARY and BACKUP databases in NSPDBVSM:

a. Estimate the total number of switched PUs that will connect into the VTAM.

b. Multiply the number in step a. by 230.

c. Add a contingency factor (100 percent is suggested).

d. The result is the minimum number of bytes you should allocate for each VSAM database in NSPDBVSM (PRIMARY and BACKUP).

Step 2 Prime the VSAM database using the sample PRIMEJCL in the *prefix.NSPS301.NSPSSAMP* data set.

Step 3 Add two DD statements to the VTAM start-up procedure to include the data sets allocated in Step 2. The DDNAMEs must be XIDDATA and XIDBACK. Replace the names NSP.XIDDATA1 and NSP.XIDBACK1 with the names of the actual VSAM data sets. The following samples show DD statements:

```
//XIDDATA DD DSN=NSP.XIDDATA1,DISP=SHARE

//XIDBACK DD DSN=NSP.XIDBACK1,DISP=SHARE
```

Step 4 Restart VTAM with the VSAM data sets allocated and primed.

Installing the Maps and SNA View XID Configuration Services Exit Routine

If you are not using your own ISTEXCCS exit, use the following procedure to install the Maps and SNA View XID Configuration Services exit routine.

Note Before you proceed, back up your ISTEXCCS load module.

Step 1 Copy the NSPESNAV exit routine from the Maps and SNA View NSPSLOAD library to the VTAM library, and rename it ISTEXCCS.

Step 2 Restart VTAM with the VSAM data sets allocated.

Installing the Combined ISM, Maps, and SNA View XID Configuration Services Exit Routine

If you are already running ISM, use the following procedure:

Step 1 Copy the NSPESNIS exit routine from the Maps and SNA View NSPSLOAD library to the VTAM library, and rename it ISTEXCCS.

Step 2 Restart VTAM with the VSAM data sets allocated.

Adding Your Own ISTEXCCS Exit Routine to NSPEMGR

This section describes how to add your old ISTEXCCS exit routine to the exit manager.

A fully functional XID Configuration Services exit routine that logs MAC, SAP, and RIF data arising from switched PU connection activity is provided with this product. If you need to provide other XID related support (in particular, dynamic PU definition or other third-party support) you must configure the SNA View XID Configuration Services exit routine to call your exit routine. Maps and SNA View lets you make these modifications with minimum interdependence and without disrupting existing functions.

Use the following procedures to add exit routines to the Configuration Services exit routine:

- If you want the exit manager to invoke the Maps and SNA View exit and another exit you indicate, use the "Configuring Exit Manager for Maps and SNA View and Customer Exit" section on page 4-17.
- If you want the exit manager to invoke the Maps and SNA View exit, the ISM exit, and another exit you indicate, use "Configuring Exit Manager for Maps and SNA View, ISM, and Customer Exit" section on page 4-18.

Configuring Exit Manager for Maps and SNA View and Customer Exit

Use the following procedure to configure the exit manager to use both the Maps and SNA View exit and your own exit.

Step 1 Modify the NSPECCSL source module located in the NSPSAMP data set by changing the instance of ISTEXCCS to the name of the intended entry point.

Note The entry point name might be different than the load module name.

Step 2 Modify the *NSPELNKS* member located in the *NSPSSAMP* data set, changing ISTEXCCS to the correct load module name of your exit.

By default, the combined exit is linked with attributes AMODE(24),RMODE(24) as specified in *NSPELNKS*. If your exit requires a 31-bit address mode, you can change *NSPELNKS* to have attributes AMODE(31),RMODE(24) because the supplied SNA View exits will operate with a 31-bit address mode.

Step 3 Modify the *ASMEMGR* member located in the *NSPSSAMP* data set as necessary. Change the *SYSLMOD* data set to a VTAM library data set. Change SYSINC1 to the name of the data set where your current exit load module is located.

Step 4 Submit the ASMEMGR job that is in the *NSPSSAMP* data set.

Step 5 Save a copy of your current ISTEXCCS exit. Copy and rename the NSPEMGR load modules to ISTEXCSS.

Step 6 Restart VTAM after allocating and priming the VSAM data sets.

Configuring Exit Manager for Maps and SNA View, ISM, and Customer Exit

Use this procedure to configure the exit manager to use the Maps and SNA View exit, the ISM exit, and your own exit.

Step 1 Modify the NSPECCSL source module located in the *NSPSAMP* data set. Comment out the current entry and uncomment the entry for SNA View, ISM and the User Exit, the entry that has NAMES=(ISTEXCCS,NSPSVRIN,NSPSVRI2). Change the instance of ISTEXCCS to the name of the current exit entry point.

Note The entry point name might be different than the load module name.

Step 2 Modify the *NSPELNKB* member located in the *NSPSSAMP* data set, changing ISTEXCCS to the correct load module name of your exit.

Note By default, the combined exit is linked with attributes AMODE(24),RMODE(24) as specified in *NSPELNKB*. If your exit requires a 31 bit address mode, you can change *NSPELNKB* to have attributes AMODE(31),RMODE(24) because the supplied SNA View exits will operate with a 31-bit address mode.

Step 3 Modify the *ASMEMGRB* member located in the *NSPSSAMP* data set as necessary. Change the *SYSLMOD* data set to a VTAM library data set. Change SYSINC1 to the name of the data set where your current exit load module is located.

Step 4 Submit the ASMEMGRB job that is in the *NSPSSAMP* data set

Step 5 Save a copy of your current ISTEXCCS exit. Copy and rename the NSPEMGR load modules to ISTEXCSS.

Step 6 Restart VTAM after allocating and priming the VSAM data sets.

Guidelines for Creating XID Configuration Services Exit Routines in a Multiple Exit Routine Environment

The following guidelines will help to ensure that multiple XID Configuration Services exit routines are compatible. When you add your own exit routines to the Configuration Services exit routine, remember that each exit routine you add is an instance of the ISTEXCCS exit routine and that, in theory, it can perform any operation normally done in a single exit routine environment. However, because of practical limitations, some restrictions are necessary. If you plan to use multiple XID Configuration Services exit routines, especially those which provide dynamic PU definitions or other active controls, see the guidelines in this section.

General Interface Guidelines

The interface guidelines are identical to those required by the VTAM ISTEXCCS exit routine. For a description, see the *VTAM Customization* manual. There are two exceptions to the information found in this reference:

- The argument list passed to each exit routine depends on the version of VTAM installed. The list will contain either four parameters for VTAM V4R3 and earlier, or five parameters for VTAM V4R4 and later.
- The second word (offset +4) in the 8-byte work area, pointed to by the fourth argument in the exit routine's input parameter list, is used now by the NSP exit manager to pass a pointer to additional information. The nature and structure of this information depends on the type of vector being processed.

Storage Pool Guidelines

The NSPEMGR module allocates all of its working storage from subpool 16. You should use a different subpool for any dynamic storage requests in the exit routines you write. If you must use subpool 16 for dynamic storage requests, ensure that your exit routine does not free subpool 16. If you free subpool 16, program checks will occur in NSPEMGR.

Guidelines for Callers Requesting XID Data

The configuration services XID exit point (ISTEXCCS) is both an active decision point and a passive monitoring point within VTAM. As an active decision point, it permits the installation to control the connection of switched PUs or to dynamically define PUs. As a passive monitoring point, it permits the installation to monitor the connection activity of all switched PUs. In general, these capabilities still exist. However, due to the dual nature of the ISTEXCCS function, and because it is now possible for multiple instances of this exit routine to exist simultaneously, there are guidelines to prevent processing ambiguous and conflicting decisions.

For the purpose of describing these guidelines, it is useful to characterize a given exit routine as either a major exit routine, or a minor exit routine, based on which exit routine has the authority to make connection decisions. The rule implemented by the NSP exit routine manager is as follows:

- A major exit routine always makes the connection decision for a given XID request. A major exit routine functions exactly as a stand-alone ISTEXCCS exit routine would coexists with the Maps and SNA View data recording functions.
- A minor exit routine always accepts the decision made by the major exit routine.

  A minor exit routine always returns a 1-byte build vector that specifies the following:

  – For a known PU, the value returned must be hexadecimal 00.

  – For a dynamic PU, the value returned must be hexadecimal 80.

  These values are not returned to VTAM from minor exit routines. Instead, they are noted, checked for validity, and discarded. The NSP exit routine manager does not accept any other values from minor exit routines.

  If you have an exit routine that makes accept and reject decisions for switched PUs, place it first in the list of exit routines in the NSPCCS macro; the NSPEMGR module makes it the major exit routine. All other exit routines are minor exit routines.

Do Not Filter These Messages

Do not filter out the following messages:

- IST093I
- IST619I
- IST590I
- IST1132I
- IST1416I
- IST1136I
- IST259I
- IST105I
- IST621I
- IST1133I
- IST1135I
- NSP2000I

Updating the Mainframe Configuration File (NSPPARM)

This section describes the purpose and content of the NSPPARM configuration file (*NSPPARM*). A sample configuration file is provided in *prefix.NSPS301.NSPSSAMP*. The configuration file contains a sequence of parameter cards that control the way the mainframe application runs. The configuration file contains three sections; each section contains a sequence of parameter cards. Code all parameter cards in uppercase characters.

Note If the mainframe application detects invalid entries in the *NSPPARM* parameter file, it displays a message indicating the error and terminates. Fix the problem and restart the mainframe application to allow the mainframe application to continue processing the parameters and finish initialization.

Section 1 of the configuration file contains required control parameter cards that specify which PUs and LUs should be discovered and monitored from the mainframe. Section 2 contains a set of required parameter cards that govern the operation of the mainframe subtasks to discover and monitor PUs and LUs. Section 3 contains an optional parameter card for VTAM commands.

The configuration file is described in the following subsections:

- Coding Control Parameter Cards in Section 1, page 4-22
- Coding the Required Subtask Parameter Cards in Section 2, page 4-36
- Coding the Optional Subtask Parameter Card in Section 3, page 4-44

Coding Control Parameter Cards in Section 1

Section 1 of the configuration file contains a set of control parameter cards that specify the PU and LU names that the mainframe application sends to the workstation during discovery and status monitoring. Table 4-1 lists the parameter cards for Section 1.

Table 4-1 Parameters Card Values and Purpose

Parameter Card	Valid Values	Purpose
EXCLUDE_SW_MAJNODES	SNA switched major node names	Specifies a list of SNA switched major nodes whose PUs and LUs will not be discovered.
INCLUDE_PU4	PU 4 names	Specifies a list of PU4 names to be monitored.
INCLUDE_SW_MAJNODES	SNA switched major node names	Specifies a list of SNA switched major nodes whose PUs and LUs will be discovered.
LU_CONTROL	OPTION	Specifies the naming convention option by which LU names will be selected for discovery and monitoring.
MSG_LEVEL	ERROR or WARN	ERROR specifies that only error messages are displayed. WARN specifies that both error and warning messages are displayed.

Table 4-1 Parameters Card Values and Purpose (continued)

Parameter Card	Valid Values	Purpose
ONLY_SWITCHED_PUS	YES or NO	NO specifies the application discovers and monitors all PUs and LUs YES specifies the application discovers and monitors only those associated with SNA switched major nodes.
PRELOAD	A single load module name per PRELOAD statement	Identifies the load modules that will be loaded and will remain loaded while the NSPOPEN address space is active.
PULU_FILTER	[-] PULU_NAME	Specifies, by name, which PUs and LUs are (or are not) discovered and monitored.
TCP_LEVEL	BASE or IV3R4	BASE specifies that you are not using IBM TCP/IP V3R4 or later. IV3R4 specifies you are using IBM TCP/IP V3R4 or later.

EXCLUDE_SW_MAJNODES

Use the EXCLUDE_SW_MAJNODES parameter card to specify a list of SNA switched major nodes whose PUs and LUs will be excluded from discovery and monitoring.

If you also use the ONLY_SWITCHED_PUS parameter card with the YES option, then the application discovers and monitors only the switched PUs and LUs that are *not* associated with the switched major nodes specified on this EXCLUDE_SW_MAJNODES parameter card.

If you use the PULU_FILTER parameter card, then the filters specified in the PULU_FILTER parameter card are applied after the PUs and LUs are filtered by the EXCLUDE_SW_MAJNODES parameter card.

Default

If you do not include an EXCLUDE_SW_MAJNODES parameter card, all PUs and LUs are discovered for all SNA switched major nodes, otherwise not filtered out by another parameter card.

Occurrences

You can include more than one EXCLUDE_SW_MAJNODES parameter card.

Card Syntax

EXCLUDE_SW_MAJNODES *NODE1 ... NODEn*

Where:

NODE1 ... NODEn specifies the names of one or more SNA switched major nodes, separated by spaces, whose PUs and LUs will be excluded from discovery. You must use the complete node name; wildcard characters are not allowed.

Example

To have the application exclude all PUs and LUs associated with the SNA switched major nodes SWDOM1 and SWDOM2, code the following parameter card:

```
EXCLUDE_SW_MAJNODES SWDOM1 SWDOM2
```

INCLUDE_PU4

Use the INCLUDE_PU4 parameter card to specify a list of SNA PU 4 major nodes (these are NCP names local to this mainframe) that will be monitored for PU 4-PU 4 connections.

You need not specify a PULU_FILTER card. If you do, you must specify the following PU names:

- The patterns to match the names of the physical line PUs and logical line PUs in the specified NCPs for the PU 4 connections you want to monitor.
- The patterns to match the names of the PU 2s and LUs that you want to monitor.

VTAM DD Card

For the mainframe application to discover PU 4-PU 4 connections, the VTAM DD statement must be specified in the NSPOPEN procedure (located in *prefix.NSPS301.NSPSSAMP*). A VTAM DD card has been supplied as a comment. You can uncomment this card and supply the correct VTAM data set containing the NCP members (substitute your data set name in place of *SYS1.VTAMLST*):

```
//VTAM      DD DSN=SYS1.VTAMLST,DISP=SHR
```

Default

If you do not specify an INCLUDE_PU4 parameter card or a PULU_FILTER card, then all PU 4-PU 4 connections will be monitored.

Occurrences

You can include more than one INCLUDE_PU4 parameter card.

Card Syntax

INCLUDE_PU4 [*NODE1* [... *NODEn*]]

Where:

NODE1 [... *NODEn*] specifies the names, including wildcard characters, of one or more SNA PU 4 nodes (NCP names local to this mainframe), separated by spaces, whose PU 4-PU 4 connections are to be monitored.

You can use the asterisk (*) wildcard in each resource name in the following places:

- At the beginning of a name
- At the end of the name, or both
- In the middle of the name

If a resource name includes an asterisk, then any PU 4 name that meets the wildcard requirements will be included in the list of monitored PU 4 nodes.

Examples

The examples are based on this sample configuration on mainframe HOST1:

```
Member(NCP1A)
  N1G1    GROUP ....
  N1G1L1  LINE  LOCADD=XXXX3745YYYY,
                 ....
  N1G1PU1 PU    ADDR=01,
                ...
  N1G1LU1 LU    LOCADDR=0,
                ...
N1G2    GROUP PHYSRSC=N1G1PU1,
              ...
  N1G2L1  LINE ...
  N1G2PU1 PU   ADDR=04CCCC3745DDDD,
              ...
  N1G2L2  LINE ...
  N1G2PU2 PU   ADDR=04EEEE3745FFFF,
              ...
Member(NCP2C)
  N2G1    GROUP ....
  N2G1L1  LINE  LOCADD=XXXX3745YYYY,
                 ....
  N2G1PU1 PU    ADDR=01,
                ...
  N2G1LU1 LU    LOCADDR=0,
                ...
N2G2    GROUP PHYSRSC=N1G1PU1,
              ...
  N2G2L1  LINE ...
  N2G2PU1 PU   ADDR=04CCCC3745DDDD,
              ...
  N2G2L2  LINE ...
  N2G2PU2 PU   ADDR=04EEEE3745FFFF,
              ...
```

The following examples show how to use search patterns:

To monitor all PU 4-PU 4 connections for NCP1A and NCP2C, use this card:

```
INCLUDE_PU4 NCP1A NCP2C
```

To monitor all PU 4-PU 4 connections for all NCPs with names that begin with the characters NCP1 or NCP2 use this card:

```
INCLUDE_PU4 NCP1* NCP2*
```

This is a useful technique for installations that change the NCP name when changes to the NCP generation are made. The original NCP name might start as NCP1A. After the first change, it would be renamed NCP1B, then NCP1C, and so on. Instead of updating the filter card every time the NCP generation is changed, NCP1* matches all these changes.

To monitor all PU 4-PU 4 connections for all PU 4 nodes whose names contain with the character “1” or end in the letter “C”, use this card:

```
INCLUDE_PU4 *1* *C
```

In this example, we want to monitor the N1G2PU1 connection in NCP1A but not the N1G2PU2 connection in the same NCP. Also, we want to monitor all PUs and LUs with names in the format CWB* or IBD*.

```
INCLUDE_PU4 NCP1*
PULU_FILTER N1G1* N1G2PU1* CWB* IBU*
```

INCLUDE_SW_MAJNODES

Use the INCLUDE_SW_MAJNODES parameter card to specify one or more SNA switched major node names whose PUs and LUs will be included in the discovery and monitoring.

If you also use the ONLY_SWITCHED_PUS parameter card with the YES option, then the application discovers and monitors only the PUs and LUs associated with the switched major nodes specified on the INCLUDE_SW_MAJNODES parameter card.

If you also use the PULU_FILTER parameter card, then the filters specified in the PULU_FILTER parameter card are applied after the PUs and LUs are filtered by the INCLUDE_SW_MAJNODES parameter card.

Default

If you do not include an INCLUDE_SW_MAJNODES parameter card, all PUs and LUs are discovered for all SNA switched major nodes, otherwise not filtered out by other parameter cards.

Occurrences

You can include more than one INCLUDE_SW_MAJNODES parameter card.

Card Syntax

INCLUDE_SW_MAJNODES *NODE1 ... NODEn*

Where:

NODE1 ... NODEn specifies the names of one or more SNA switched major nodes, separated by spaces, whose PUs and LUs will be included in discovery. You must use the complete node name; wildcard characters are not allowed.

Example

To have the application include all PUs and LUs associated with the SWDOM1 and SWDOM2 switched major nodes, code the following parameter card:

```
INCLUDE_SW_MAJNODES SWDOM1 SWDOM2
```

LU_CONTROL

Use the LU_CONTROL parameter card to specify how LUs should be discovered and monitored. This card provides important information required to properly obtain initial status and maintain the status of VTAM resources defined to NSPOPEN.

Default

If you do not include an LU_CONTROL parameter card, the CONSISTENT option is the default.

Occurrences

You can include only one LU_CONTROL parameter card.

Card Syntax

LU_CONTROL *OPTION*

Where:

OPTION specifies how LU names are selected for discovery and monitoring. The following options are available:

- CONSISTENT—LUs are discovered and monitored based on a consistent naming convention of the associated PUs and LUs. An LU name is considered consistent when the PU and its associated LUs would pass the same filter. This occurs when the logical unit under a physical unit share a prefix.

For example, the following PUs and LUs would be defined in such a way to allow the CONSISTENT option to be applied.

```
WOMPU1    PU
WOMPU100  LU LOCADDR=00   PULU_FILTER WOM1*
WOMPU101  LU LOCADDR=01
WOMPU2    PU
WOMPU200  LU LOCADDR=00   PULU_FILTER WOM2*
WOMPU201  LU LOCADDR=01
```

The previous PU_LU_FILTERS could be combined to the following filter:

```
PU_LU_FILTER WOM*
```

The consistent option is the preferred option because it reduces processing on the host. In a consistent environment, the number of PULU_FILTERS is usually much lower than required for an UNIQUE environment.

When status information flows from VTAM to NSPOPEN, it is in the form of VTAM display messages. These messages do not always indicate the type of resource involved (PU versus LU). In a consistent environment, a single PU_LU_FILTER can control one or more physical units and the associated logical units.

Consider the following example using a different naming convention:

```
PU0000     PU
LU000000   LU LOCADDR=00
LU000001   LU LOCADDR=01
PU0001     PU
LU000100   LU LOCADDR=00
LU000101   LU LOCADDR=01
```

This example shows a clear mapping between the PUs name and the associated logical units. However, a single PULU_FILTER cannot be used for both the LUs and PUs.

When the CONSISTENT option is specified, additional host processing is saved because once a physical unit passes the PULU_FILTERS, the associated logical units are considered to have passed the filter.

- UNIQUE—LUs are discovered and monitored using different naming conventions than their associated PUs. During discovery, both Physical Units and Logical Units will be checked against the specified PULU_FILTERS. Only the resources that pass (satisfy) these filters will be processed by NSPOPEN. This option generally takes more host processing because of the requirement to process all resources (LUs and PUs) against a larger number of PULU_FILTERs.

  It is important to note that incorrectly specifying CONSISTENT for LU_CONTROL for an environment that is inconsistent can cause the status to not flow for certain logical units.

- IGNORE—The application does not discover or monitor LUs.

Example

To have the application discover LUs based on a consistent PU and LU naming convention, code the following parameter:

```
LU_CONTROL CONSISTENT
    AP000      PU
    AP000L00   LU LOCADDR=00
    AP000L01   LU LOCADDR=01
    AP001      PU
    AP001L00   LU LOCADDR=00
    AP001L01   LU LOCADDR=01
    BP000      PU
    BP000L00   LU LOCADDR=00
    BP000L01   LU LOCADDR=01
    BP001      PU
    BP001L00   LU LOCADDR=00
    BP001L01   LU LOCADDR=01
```

The following filters would be specified for the

```
PULU_FILTER AP* BP*
```

To have the application discover LUs based on a unique PU and LU naming convention, code the following parameter:

```
LU_CONTROL UNIQUE
    PUA000     PU
    LUA00000   LU LOCADDR=00
    LUA00001   LU LOCADDR=01
    PUA001     PU
    LUA00100   LU LOCADDR=00
    LUA00101   LU LOCADDR=01
    PUB000     PU
    LUB00000   LU LOCADDR=00
    LUB00001   LU LOCADDR=01
    PUB001     PU
    LUB00100   LU LOCADDR=00
    LUB00101   LU LOCADDR=01

    PULU_FILTER PUA* LUA* PUB* LUB*
```

If LU_CONTROL was set to CONSISTENT with the following PULU_FILTER:

```
    PULU_FILTER PUA* PUB*
```

The status of the logical units would be sent to the workstations during discovery. The status of the associated logical units are sent because the PU passes one or more PULU_FILTERs. However, later status updates for the logical units would not be sent to the workstations.

When NSPOPEN receives a status update for a logical unit, it evaluates the logical unit's name against the specified PULU_FILTERs. In this case, the logical units will not pass any filters preventing their status information from flowing to workstations.

If LU_CONTROL was set to consistent with the following PULU_FILTER, the status of the logical units would be sent to the workstations during discovery. However, later status updates would not be sent to the workstations.

```
    PULU_FILTER PUA* PUB*
```

MSG_LEVEL

Use the MSG_LEVEL parameter card to specify whether warning messages are displayed.

Default

If you do not supply a MSG_LEVEL parameter card, ERROR is used.

Occurrences

You can include only one MSG_LEVEL parameter card.

Card Syntax

MSG_LEVEL [ERROR | WARN]

Where:

- ERROR specifies that just error messages are displayed. If you omit this parameter, ERROR is the default value.
- WARN specifies that both error and warning messages are displayed.

ONLY_SWITCHED_PUS

Use the ONLY_SWITCHED_PUS parameter card to specify whether the application will discover all PUs and LUs, or just the PUs and LUs associated with SNA switched major nodes.

Note The ONLY_SWITCHED_PUS card is supported only in VTAM V4R3 and higher. If you are using a version of VTAM earlier than V4R3, use ONLY_SWITCHED_PUS NO.

Default

YES is the default: the mainframe application will discover and monitor only PUs and LUs associated with SNA switched major nodes.

Occurrences

You can include only one ONLY_SWITCHED_PUS parameter card.

Card Syntax

ONLY_SWITCHED_PUS {YES | NO}

Where:

{YES | NO} specifies whether the application will discover and monitor only PUs and LUs associated with an SNA switched major node.

- YES—The application will discover and monitor only PUs and LUs associated with an SNA switched major node. YES is the default value.
- NO—The application will discover and monitor all PUs and LUs. If you are using a version of VTAM earlier than V4R3, or if you want to monitor non-switched PUs, use the NO option.

Example

To have the application discover and monitor only PUs and LUs associated with an SNA switched major node, code the following parameter card:

```
ONLY_SWITCHED_PUS YES
```

PRELOAD

Use the PRELOAD parameter card to identify load modules, which will be loaded and will remain loaded while the NSPOPEN address space is active. This parameter card is used to improve performance when a particular load module is being loaded and unloaded frequently.

Note Use of this card is normally something Cisco TAC may recommend.

Default

If no parameter is specified then no PRELOADS are performed.

Occurrences

You can include several PRELOAD parameter cards. Only one load module name is allowed on each preload statement.

Card Syntax

PRELOAD

Example

To preload load module L$CALMTO, code the following parameter card:

```
PRELOAD L$CALMTO
```

PULU_FILTER

Use the PULU_FILTER parameter card to specify, by name, which PUs and LUs will be discovered and monitored, and which will not be discovered or monitored. Because this card filters PUs and LUs by name, name your PUs and LUs with a common naming convention. You can include a series of PULU_FILTER parameter cards.

Each PULU_FILTER parameter card contains one or more filter tokens to be applied to the PU and LU names. Each filter token is a portion of an PU or LU name, and includes asterisks (*) as wildcards at the beginning of the name, at the end of the name, or both. For example, to specify all PUs and LUs with names beginning with ABC, you would code a filter token as ABC*.

If you precede a filter token with a hyphen (-), PUs and LUs that match that filter token are *not* discovered.

Default

If you do not supply a PULU_FILTER parameter card, all PUs and LUs will be discovered and monitored regardless of naming conventions, unless filtered out by other parameter cards.

Occurrences

You can include several PULU_FILTER parameter cards.

Card Syntax

PULU_FILTER [-]*FILTER_TOKEN1 . . .* [-]*FILTER_TOKENn*

Where:

[-] specifies that PU and LU names that satisfy the following *FILTER_TOKEN* values will be ignored (not discovered).

If you omit the - character, then PU and LU names that satisfy the following *FILTER_TOKEN* values will be discovered.

FILTER_TOKEN1 ... FILTER_TOKENn specifies one or more filter tokens, each consisting of a portion of a PU or LU name with asterisks to be used as wildcards. You can code a series of filters on the same PULU_FILTER parameter card.

Each filter token must contain at least one asterisk at the beginning or end of the token. Each filter token can contain two asterisks, one at each end. The following tokens are valid:

- PUNAM*
- *NAME
- *NAME*

Examples

To discover and monitor only those PUs and LUs beginning with the characters VPU, but to ignore PUs and LUs beginning with the characters VPU5, code the following parameter card:

```
PULU_FILTER VPU* -VPU5*
```

To discover and monitor all PUs and LUs beginning with the characters ABC, ending with the characters DEF, or containing the characters GHI, code the following parameter card:

```
PULU_FILTER ABC* *DEF *GHI*
```

TCP_LEVEL

Use the TCP_LEVEL card to identify whether you are running the IBM TCP/IP V3R4 (or later) protocol stack.

Default

If you do not supply a TCP_LEVEL parameter card, BASE is used.

Occurrences

You can include only one TCP_LEVEL parameter card.

Card Syntax

TCP_LEVEL [BASE | IV3R4]

Where:

- BASE specifies that you are not running TCP/IP V3R4 or later. Choose BASE if you are running a non-IBM stack, or an IBM stack that is not V3R4 or later.
- IV3R4 specifies that you are running IBM TCP/IP V3R4 or later.

Coding the Required Subtask Parameter Cards in Section 2

Section 2 of the configuration file contains a set of parameter cards that govern the operation of the mainframe subtasks that discover and monitor PUs and LUs. Table 4-2 lists the parameter cards you can use in Section 2.

Table 4-2 Required Subtask Parameter Cards

Parameter Card	Valid Values	Purpose
DISCOVER	VTAM APPL definition	Specifies a VTAM APPL definition for SNA PU and LU discovery.
MVS	Console name	Provides the application with MVS console support.
PPI	None	Requests setup of the program-to-program interface (PPI) to NetView or SOLVE:Netmaster.
PPO	VTAM APPL definition	Specifies a VTAM APPL definition for a primary program operator (PPO) application; allows the application to act as VTAM primary program operator.
SEC	BLOCK, NO	Specifies the security method for workstation users when issuing mainframe commands.
SERVER	APPL, LU, mode, message server, command server	Identifies VTAM resources for LU 6.2 workstation connection.
STATUS	VTAM APPL definition	Specifies a VTAM APPL definition for SNA PU and LU status updates.
TCP	Host connection interface port, host command server port	Identifies port numbers for TCP/IP workstation connection.

DISCOVER

The DISCOVER parameter card specifies the name of a VTAM APPL definition to be used for the discover subtask.

Default

You must supply a DISCOVER parameter card; NSPDSC1 is the default value.

Occurrences

You can include only one DISCOVER parameter card.

Card Syntax

DISCOVER *VTAM_applid*

Where:

VTAM_applid specifies the name of a VTAM APPL definition card coded with AUTH=SPO. This APPL ID identifies an application that will discover SNA PUs and LUs. The named application is authorized to issue VTAM DISPLAY commands during the discover process.

In the *prefix.NSPS301.NSPSSAMP(NSPAPPL)* member, the sample discovery VTAM APPL is named NSPDSC1.

Example

If the ID of the discover subtask's APPL definition, coded with AUTH=SPO, is NSPDSC1, you would code the following DISCOVER parameter card:

```
DISCOVER NSPDSC1
```

MVS

The MVS parameter card specifies the name of the extended MCS mainframe console to be defined for receipt of MVS messages. You define this name for the application workstation to receive MVS messages. The MVS parameter card is required so the mainframe application will be notified if the VSAM data set is changed from the primary data set to the backup data set. In a Sysplex environment, console messages are retrieved from the local system only.

Default

If you do not supply an MVS parameter card, and the exit is forced to switch to the backup VSAM data set, the mainframe application is not notified.

Occurrences

You can include only one MVS parameter card.

Card Syntax

MVS *console_name*

Where:

console_name is the name of the extended MCS console to be defined for receipt of MVS messages. If this name is defined in RACF, the OPERPARM values for this name are used for the console definition. Otherwise, a console is defined with default parameters AUTH=INFO and ROUTCDE=ALL.

Example

To specify NSPCONS1 as the extended MCS console, use the following MVS parameter card:

```
MVS NSPCONS1
```

PPI

Use the PPI parameter card to connect the application to the NetView or SOLVE:Netmaster PPI for the receipt of VTAM messages. The PPI card tells the application to establish a program-to-program interface with NetView or SOLVE:Netmaster so that the application can receive solicited and unsolicited VTAM messages. The PPI must be active in accordance with the NetView or SOLVE:Netmaster documentation.

Default

You must supply a PPI or PPO parameter card.

Occurrences

You can include only one PPI parameter card.

Card Syntax

PPI

Example

To connect the application to the NetView or SOLVE:Netmaster PPI for receiving VTAM messages at the workstation, code the following PPI parameter card:

```
PPI
```

Note Do not include a PPI card if *neither* NetView nor SOLVE:Netmaster is present on the system. Instead, use the PPO parameter card to specify a primary program operator.

PPO

Use the PPO parameter card to specify the name of a VTAM APPL definition that will act as a primary program operator application program (PPO) to receive solicited and unsolicited VTAM messages.

Note Do not include a PPO parameter card if the application is running in combination with other management software, such as NetView or SOLVE:Netmaster; only one application in a domain can be the primary program operator. Instead, use the PPI parameter card to connect the application to NetView or SOLVE:Netmaster.

Default

You must supply a PPI or PPO parameter card.

Occurrences

You can include only one PPO parameter card.

Card Syntax

PPO *VTAM_applid*

Where:

VTAM_applid is the ID of the APPL definition coded with AUTH=PPO. This identifies the primary program operator application program that will receive unsolicited VTAM messages.

Example

If the ID of the APPL definition coded with AUTH=PPO is NSPPPO1, code the following PPO parameter card:

```
PPO NSPPPO1
```

SEC

Use the SEC parameter card to specify the security clearance level needed to activate and deactivate SNA resources from the workstation.

- If you specify **SEC NO** and uncomment the SPO card, the Maps motif applications at the workstation will be able to send ACT/INACT commands to the mainframe.
- If you do not specify **SEC NO** (**BLOCK** is the default), any SPO card will be ignored and the Maps motif applications at the workstation will be unable to send ACT/INACT commands to the mainframe.

Default

If you do not supply an SEC parameter card, BLOCK is used.

Occurrences

You can include only one SEC parameter card.

Card Syntax

SEC [BLOCK | NO]

Where:

BLOCK means that workstation users cannot issue mainframe commands. This is the default. The SPO subtask will not be started even if you provide the parameter card: the card is not needed because workstation commands are not allowed. If you omit this parameter, BLOCK is the default value.

If you use the BLOCK option, remove any SPO cards from the parameter file.

NO means that any workstation user can issue mainframe commands.

Example

To prevent workstation users from issuing mainframe commands, use the following parameter card:

```
SEC BLOCK
```

SERVER

Use the SERVER parameter card to provide the values needed to establish an LU 6.2 connection between the mainframe application and the workstation application.

Default

If you do not supply a SERVER parameter card, no LU 6.2 sessions are established. Code a TCP card to configure a TCP/IP connection instead. You must provide at least one SERVER or TCP card.

Occurrences

You can include up to ten SERVER parameter cards, one for each workstation attached using LU 6.2.

Card Syntax

SERVER *plu slu* PARALLEL NSPOPNMS NSPOPNCS

Where:

plu is the label of the VTAM APPL definition you coded with APPC=YES, which is the primary LU for the application at the mainframe.

slu is the label of a CDRSC for the independent secondary LU defined for the workstation and associated with the workstation PU.

PARALLEL is the logmode protocol. Enter PARALLEL.

NSPOPNMS is the name of the SNA LU 6.2 transaction program for the workstation message server.

NSPOPNCS is the name of the SNA LU 6.2 transaction program for the workstation command server.

Example

To define an LU 6.2 session between primary logical unit NSPAPL1 and secondary logical unit NSPLU01 using the logmode PARALLEL, code the following SERVER parameter card:

```
SERVER NSPAPL1 NSPLU01 PARALLEL NSPOPNMS NSPOPNCS
```

STATUS

The STATUS parameter card specifies a VTAM APPL definition that will update the status of SNA PUs and LUs.

Default

None. The STATUS card is required.

Occurrences

You can include one STATUS parameter card.

Card Syntax

STATUS *VTAM_applid delay_time*

Where:

VTAM_applid is the ID on an APPL definition card coded with AUTH=SPO. The APPL ID identifies a VTAM application, which will update the status of SNA PUs and LUs.

In the *prefix.NSPS301.NSPSSAMP(NSPAPPL)* data set, the sample status VTAM APPL is named NSPSTA1.

delay_time is the time delay (in seconds) before asking VTAM for the status of a resource. The default is 10 seconds. This delay time allows VTAM to synchronize before collecting PU details.

If the application detects a PU in the pending state *after* VTAM issues the IST590I message (IST590I indicates that a connection is established), the application retries the display of the PU in a minimum of 10 seconds and continues to retry the display 100 times. If at that point the PU is still pending, the application no longer queries its status. Instead, the application displays message NSP039, which indicates you need to increase the value of *delay_time*.

Example

If the ID of the status program's APPL definition, coded with AUTH=SPO, is NSPSTA1, code the following parameter card:

```
STATUS NSPSTA1
```

TCP

Use the TCP parameter card to configure a TCP/IP connection between the mainframe application and the workstation application. Specify the TCP/IP ports on the mainframe used for the host connection interface and the host command server.

Note If you are using only an LU 6.2 session between the workstation and mainframe, either remove this card from the *NSPPARM* file or comment it out.

Default

If you do not supply a TCP parameter card, no TCP/IP connections are available. Code a SERVER card to establish an LU 6.2 session instead. You must provide at least one SERVER or TCP card.

Occurrences

You can include up to 20 TCP parameter cards (1 for each workstation attached using TCP/IP).

Card Syntax

TCP *hciport cmdport*

Where:

hciport is the port number opened on the mainframe for establishing a TCP socket connection with the host connection interface on the workstation.

cmdport is the port number opened on the mainframe for establishing a TCP socket connection with the host command server on the workstation.

Example

To reserve ports 6104 and 6105 on the mainframe for the workstation host-connection interface and host command server, code the following TCP parameter card:

```
TCP 6104 6105
```

Note The TCP port numbers on the mainframe must match the port numbers on the workstation.

Coding the Optional Subtask Parameter Card in Section 3

Section 3 of the configuration file contains an optional parameter card that lets the workstation user issue VTAM commands. Table 4-3 shows the parameter card in Section 3.

Table 4-3 Parameter Card

Parameter Card	Valid Values	Purpose
SPO	VTAM APPL definition	Defines a secondary program operator (SPO) application that allows the workstation to activate and deactivate SNA resources.

SPO

Use the SPO parameter card to define a secondary program operator application program. You must provide one SPO card to be able to activate and deactivate LUs and PUs from the workstation Motif applications.

To prevent workstation users from activating and deactivating SNA resources, do not provide an SPO card. Code the BLOCK option on the SEC parameter card to override an SPO parameter card.

In the default *NSPPARM* file, this card has a comment delimiter so that it is not read as a parameter card. If you leave this card commented out in the *NSPPARM* file, workstation users cannot activate and deactivate SNA resources. If you need this card at a later time, you can remove the comment delimiter.

Default

If you do not supply an SPO parameter card, users cannot activate and deactivate SNA resources from a workstation.

Occurrences

You can include only one SPO parameter card.

Card Syntax

SPO *VTAM_applid*

Where:

VTAM_applid is the name on an APPL definition card coded with AUTH=SPO. The APPL ID identifies an SPO application that will receive solicited messages generated by Activate and Deactivate commands issued from the workstation.

In the *prefix.NSPS301.NSPSSAMP(NSPAPPL)* member, the sample SPO VTAM APPL is named NSPSPO1.

Example

If the ID of the APPL definition coded with AUTH=SPO is NSPSPO1, code the SPO parameter card:

```
SPO NSPSPO1
```

Updating NetView

This section describes how to update NetView. To update NetView, use the following subsections:

Note Customizing NetView is a required step the product to function properly. You must restart NetView for the changes to take effect.

NetView Version 1.1 Automation Facilities

If you are using Tivoli NetView for OS/390 Version 1.1 or later, SNA View and Maps can use the NetView automation facilities rather than the CiscoWorks Blue PPI task. This change improves performance of the status updates. To make this change, edit the *NSPSVTBL* member, and change the current automation table statement.

Verifying the Subsystem Interface Installation

Verify that the NetView subsystem address space is active and that the NetView PPI is enabled, as defined in the *NetView Installation and Administration Guide*. The NetView program-to-program interface is necessary for cross-memory communications with NetView. Before starting NetView, start the subsystem application.

Making Changes to the NetView DSIPARM Data Set

To change the NetView *DSIPARM* data set, use the following procedure:

Step 1 Define an additional NetView autotask, here named NSPAUTO1, by adding the following definition to the *DSIOPF* member of your NetView's *DSIPARM* data set:

```
NSPAUTO1 OPERATOR PASSWORD=PASSWORD
         PROFILEN NSPPROF
```

You can change the Operator ID (NSPAUTO1) to conform to your site requirements, but it must match the value used in the automation table entry (NSPSVTBL) for the NSPSVTAM CLIST. If you change this operator ID, you must change it also in the NSPSVTBL entry.

You can change the PROFILEN name (NSPPROF) to conform to your site requirements. The profile is defined in Step 2.

You might also need to define the Operator ID to the security product.

Step 2 Define a profile for the new NetView autotask (defined as NSPAUTO1 in Step 1) by adding a member named NSPPROF to your NetView's *DSIPRF* data set. NSPPROF must contain the following three lines:

```
NSPPROF     PROFILE
            AUTH      MSGRECVR=NO,CTL=GLOBAL
            END
```

You can change the member name to conform to your site requirements, but it must match the PROFILEN statement that was coded in Step 1.

Step 3 Add the following line to your initial command list to ensure that the new NetView autotask (NSPAUTO1) is started each time NetView is started:

```
AUTOTASK OPID=NSPAUTO1
```

Step 4 Copy the *NSPSVTBL* member from *prefix.NSPS301.NSPSAMP* to a NetView DSIPARM data set. Add an include for NSPSVTBL in the current automation table member (%INCLUDE NSPSVTBL).

Note The following two steps are only for users of NetView Version 2.3 or earlier. Unless you are running NetView Version 2.3 or earlier, skip these steps.

Step 5 Define the mainframe optional task by adding the following definition to the *DSIDMN* member of your NetView's *DSIPARM* data set:

```
TASK MOD=NSPVPPI,TSKID=NSPVPPI,PRI=8,INIT=Y
```

Verify that the NetView task (CNMCSSIR) is defined with INIT=N, and that CNMCSSIR is started in command list CNME1035 during NetView initialization. This task provides command and message forwarding services.

Step 6 Define a command model for the NSPMQS load module by adding the following definition to the *DSICMD* member of your NetView's *DSIPARM* data set:

```
NSPMQS    CMDMDL MOD=NSPMQS,RES=N
```

Copying the *NSPSVTAM* Member to the NetView List Data Set

Copy the *NSPSVTAM* member from *prefix.NSPS301.NSPSCLST* to a NetView CLIST data set (*DISCLD*).

If you are running NetView 2.3 or NetView 2.4 on the host, the NSPSVTAM command list in the NetView DSICLD dataset concatenation should be updated. The PIPE PPI stage used in the command list is not available on NetView 2.3 or 2.4 so the NSPSVTAM command should be replaced with the following 3 lines of code:

```
&CONTROL ERROR
&DATA = &CONCAT X'0006' &PARMSTR
NSPMQS NSPVPPI &DATA
```

Assembling and Linking NetView Modifications

The sample library (*prefix.NSPS301.NSPSSAMP*) contains the following programs you must add to customize NetView:

- NSPVPPI data services task (DST)
- NSPMQS command processor to run in NetView's address space

Create the load modules (NSPVPPI, NSPMQS) by modifying and submitting the JCL in *prefix.NSPS301.NSPSSAMP(ASMJCL)*, according to the instructions in the member. After the load modules are created, copy them to a NetView STEPLIB data set.

Activating the Changes to NetView

Restart NetView to activate the changes that you made to NetView.

Updating SOLVE:Netmaster

This section describes how to enable the mainframe application to interact with SOLVE:Netmaster. You do not have to restart SOLVE:Netmaster for the changes to take effect.

To enable SOLVE:Netmaster to work with the mainframe application, use the procedures in the following subsections:

- Verifying Subsystem Interface Installation, page 4-49
- Changing the SOLVE:Netmaster PPO Procedure, page 4-50
- Changing the SOLVE:Netmaster Command Data Set, page 4-50
- Starting the Mainframe Procedures, page 4-50
- Verifying the SOLVE:Netmaster Updates, page 4-51

The data set members listed in Table 4-4 are located in *prefix.NSPS301.NSPSCLST* and are used to change SOLVE:Netmaster for the mainframe application.

Table 4-4 Data Set Members

Member	Description
NSPKDOC	Describes the PROCs and how to implement the mainframe application.
NSPKPPO	Contains modifications to the SOLVE:Netmaster PPO procedure.
NSPKPPI	Contains the PROC that receives solicited and unsolicited VTAM messages and sends them to the mainframe application.
NSPKCMD	Contains the PROC that is the command PPI receiver (NSPNETV) and starts NSPKCM1.
NSPKCM1	Contains the PROC that is the command sender and receiver.

Verifying Subsystem Interface Installation

Verify that the SOLVE:Netmaster PPI address space is active, as defined in the *SOLVE:Netmaster Implementation and Administration Guide*. The PPI is necessary for cross-memory communications between SOLVE:Netmaster and the mainframe application.

Changing the SOLVE:Netmaster PPO Procedure

For the mainframe application to receive system message information from SOLVE:Netmaster, the network control language (NCL) code in *prefix.NSPS301.NSPSCLST(NSPKPPO)* must be added to the production PPO procedure (PPOPROC) at a point where all messages will be seen. The recommended point for this code addition is immediately following the mainline &PPOREAD.

For the PPO procedure to receive solicited and unsolicited VTAM messages, those messages must be sent to the PPO procedure by SOLVE:Netmaster. You can use the **DEFMSG** command to control which messages are sent to the PPO procedure. See the SOLVE:Netmaster documentation for more information.

Changing the SOLVE:Netmaster Command Data Set

Copy the following members from *prefix.NSPS301.NSPSCLST* to a SOLVE:Netmaster command data set (the *COMMAND DD* card): NSPKCMD, NSPKCM1, and NSPKPPI.

Starting the Mainframe Procedures

The NSPKPPI and NSPKCMD PROCs are the principal PPI procedures that send PPO data through the PPI and wait for commands coming through the PPI. The NSPKPPI and NSPKCMD PROCs must be active at all times and must run in a background environment within SOLVE:Netmaster. To accomplish this, add the following statements to your NMINIT or NMREADY initialization PROCs:

```
Sub BSYS NSPKPPI
Sub BSYS NSPKCMD
```

Note You can also issue these as commands from an MCS console.

Verifying the SOLVE:Netmaster Updates

When you complete all updates to SOLVE:Netmaster, you can issue the following command to verify correct installation:

SH PPIUSERS

The command displays two receivers, SNAVIEW and NSPNETV, and indicates the number of messages queued so you can monitor the number of messages sent to the mainframe application.

CHAPTER 5

Using the Mainframe Application

This chapter provides information on starting, stopping, and using the mainframe application.

This chapter includes the following main sections:

Starting and Stopping the Mainframe Application

You can start the mainframe application as a started task or a batch job.

Starting and Stopping the Application as an MVS Started Task

To start the mainframe application as a started task (an MVS task that is started with a **START** (**S**) command), copy the *prefix.NSPS301.NSPSSAMP(NSPOPEN)* procedure into the started tasks library, modifying the data set names according to the instructions at the top of the job. Start the NSPOPEN procedure from an MVS console with the **S NSPOPEN** command.

Starting the Application as an MVS Batch Job

To start the mainframe application as a batch job, modify and submit the JCL in *prefix.NSPS301.NSPSSAMP*(*NSPJCL*).

Stopping the Application

To stop the mainframe application, enter the following command from the operator console:

F NSPOPEN,STOP

If the application does not stop, enter the command again.

Mainframe Subtasks

When the mainframe program starts, it starts a number of subtasks simultaneously, depending on the way you configured the mainframe application. Table 5-1 lists the subtasks that can be started.

:

Table 5-1 Mainframe Subtasks

Subtask ID	Purpose	Occurrences
DISCOVER	Discovers the PUs and LUs in the network.	1
MVS	Retrieves MVS messages.	1

Table 5-1 Mainframe Subtasks (continued)

Subtask ID	Purpose	Occurrences
PPI	Supports the NetView or SOLVE:Netmaster program-to-program interface; lets the workstations receive solicited and unsolicited VTAM messages.	1
PPO	Lets the workstations receive unsolicited VTAM messages.	1
SERVER	Provides LU 6.2 connection to workstations; one occurrence for each workstation connected by LU 6.2.	0-10
SPO	Supports secondary program operators.	0-15 (0 means that no workstations can enter VTAM commands)
STATUS	Reports changes in the status of PUs and LUs to the workstations.	1
TCP	TCP/IP connection to a workstation; one occurrence for each workstation connected by TCP/IP.	0-20

Issuing Mainframe Commands

You can enter the mainframe commands shown in Table 5-2 from any defined MVS console or extended MVS console, including NetView and SOLVE:Netmaster

:

Table 5-2 Mainframe Commands

Command	Description
HELP	Displays command help information.
INIT	Starts a defined subtask.
KILL	Stops a defined subtask.

Table 5-2 Mainframe Commands (continued)

Command	Description
SHOW ADDR	Displays all subtasks memory addresses.
SHOW CONN	Displays the TCP/IP address of each Workstation client connected by TCP/IP.
SHOW DLC	Displays the Media Access Control (MAC), service access point (SAP), and Routing Information Field (RIF) data for a PU name.
SHOW FLOW	Displays the number of messages in the input and output queues for each subtask.
SHOW FREEQ	Displays the number of used and available buffers on the FREE queue for each subtask.
SHOW TASK	Displays the name and status of each subtask.
SHOW TRACE	Displays the current settings for the **TRACE** command.
SHOW VERSON	Displays the current version information.
STOP	Stops the mainframe task and its subtasks.
TRACE	Starts and stops tracing in the mainframe subtasks.

HELP

The **HELP** command displays the list of mainframe commands.

Command Syntax

F NSPOPEN,HELP

Example

To display the list of mainframe commands, enter the following command:

F NSPOPEN,help

Sample Output

```
NSP595 Command Entered: HELP
FILTER ADD msgid - Adds a message filter
FILTER DEL msgid - Deletes a message filter
HELP             - Display this list
INIT             - Activate a defined subtask
KILL             - Terminate a defined subtask
MESSAGES         - Control mainframe messages
SHOW ADDR        - Display the important memory addresses of each
subtask
SHOW CONN        - Display the client information for the TCP subtasks
SHOW DLC puname  - Display the mac, sap, and rif information for a pu
SHOW FILTER      - Display all defined message filters
SHOW FLOW        - Display task message flow information
SHOW FREEQ       - Display task free queue
SHOW MESSAGES    - Display the MESSAGES value
SHOW TASK        - Display all subtasks and their status
SHOW TRACE       - Display all subtasks and their traces
SHOW VERSION     - Display the compiled date and time of each subtask,
and product version
STOP             - Shutdown Sna Host NSP695 SNA Host HELP command
processed
```

INIT

The **INIT** command starts a mainframe subtask already defined in the NSPPARM configuration file. When you start a subtask, automatic restarts are enabled for that subtask.

Command Syntax

F NSPOPEN,INIT *subtask_name*

Where:

subtask_name is the name of the mainframe subtask that you want to start. Use the **F NSPOPEN,SHOW TASK** command to display a list of subtasks by name.

Example

To start subtask TCP-0, enter the following command:

F NSPOPEN,INIT TCP-0

Sample Output

NSP595 Command entered: INIT TCP-0

KILL

The **KILL** command stops a mainframe subtask. The mainframe application then displays a message to remind you that the automatic restart has been disabled for the subtask you are killing. The status of the subtask is changed to DOWN. The INPUTQ and OUTPUTQ values are invalid when a subtask is in the DOWN state and should be ignored. You can restart the subtask with the **INIT** command.

Command Syntax

F NSPOPEN,KILL *subtask_name*

Where:

subtask_name is the name of the subtask to be killed. Use the **F NSPOPEN,SHOW TASK** command to display a list of subtasks by name.

Example

To kill subtask TCP-0, enter the following command:

F NSPOPEN,KILL TCP-0

SHOW ADDR

The **SHOW ADDR** command displays the memory addresses of the internal header control block, subtask control block, and MVS task control block for each mainframe subtask. This command is for problem diagnosis only.

Command Syntax

F NSPOPEN,SHOW ADDR

Example

To display the mainframe subtask memory addresses, enter the following command:

F NSPOPEN,SHOW ADDR

Sample Output

```
NSP595 Command entered: SHOW ADDR
NSP603 TNUM TASKNAME   ADDRESS  HEADER   TCB
NSP603    0 MAINTASK   00000000 09259E88 00000000
NSP603    1 DISCOVER-0 0922F318 09253E88 008DD6D0
NSP603    2 MVS        0922F338 092685C8 008DD388
NSP603    3 PPI        0922F358 0926ADC8 008D6A88
NSP603    4 STATUS-0   0922F378 0928AE88 008D6858
NSP603    5 SERVER-0   0922F398 092B6608 00000000
NSP603    7 TCP-0      0922F3D8 09346608 008BEE88
NSP603    8 TCP-1      0922F3F8 0934A608 008D5E88
NSP603    9 CMD-0      0922F418 0934D5C8 008C0A88
NSP603   10 SPO-0      0922F438 0944AE88 008C0858
NSP695 SNA Host SHOW command processed
```

Table 5-3 describes the fields in the **SHOW ADDR** command output

Table 5-3 SHOW ADDR Fields

Field	Meaning
TNUM	Subtask number.
TASKNAME	Subtask name.
ADDRESS	Address of subtask control block for this subtask.
HEADER	Address of subtask header.
TCB	Task control block address.

SHOW CONN

The **SHOW CONN** command displays the TCP/IP addresses (or host names) of the client workstations connected via TCP/IP.

Command Syntax

F NSPOPEN,SHOW CONN

Example

To display the TCP/IP connections, enter the following command:

F NSPOPEN,SHOW CONN

Sample Output

```
NSP595 Command entered: SHOW CONN
NSP706 TNUM TASKNAME   SPECIFIC  CLIENT
NSP706    7 TCP-0      6506,6507 NO CONNECTION
NSP706    8 TCP-1      6516,6517 171.69.163.115
NSP695 SNA Host SHOW command processed
```

Table 5-4 describes the fields in the **SHOW CONN** command output.

Table 5-4 SHOW CONN Fields

Field	Meaning
TNUM	Subtask number.
TASKNAME	Subtask name.
SPECIFIC	TCP/IP port numbers.
CLIENT	IP address of connected workstation. NO CONNECTION = no workstation connected.

SHOW DLC

The **SHOW DLC** command displays the MAC, SAP, and RIF data for any switched PU name.

Command Syntax

F NSPOPEN,SHOW DLC *PU_name*

Where:

PU_name is the name of any switched PU.

Example

To display the MAC, SAP, and RIF data for PU IBUPC1, enter the following command:

F NSPOPEN,SHOW DLC IBUPC1

Sample Output

```
NSP595 Command entered: SHOW DLC IBUPC1
NSP708 IBUPC1 05DAA011 400137451088 04 0000F6419B36 04
NSP710 0A30A0441F42055
NSP695 SNA Host SHOW command processed
```

Table 5-5 describes the fields in the **SHOW DLC** command output.

Table 5-5 SHOW DLC Fields

Field	Meaning
IBUPC1	PU name.
05DAA011	XID.
400137451088	Local MAC.
04	Local SAP.
0000F64190B36	Remote MAC.
04	Remote SAP.
0A30A0441F42055	RIF data.

SHOW FLOW

The **SHOW FLOW** command displays information about the messages for each subtask. The output from the **SHOW FLOW** command is for use by the Cisco TAC.

Command Syntax

F NSPOPEN,SHOW FLOW

Example

To display the messages in the input and output queues, enter the following command:

F NSPOPEN,SHOW FLOW

Sample Output

```
NSP595 Command entered: SHOW FLOW
NSP605 TNUM TASKNAME   INPUTQ  OUTPUTQ   INFLOW  OUTFLOW  MC
NSP605    1 DISCOVER-0      0        0        0        0   0
NSP605    2 MVS             0        0     2340     2340   3
NSP605    3 PPI             0        0        0        0   0
NSP605    4 STATUS-0        0        0       26       26   0
NSP605    5 SERVER-0        0        0        0        0   0
NSP605    7 TCP-0           0        0        0        0   0
NSP605    9 CMD-0           0        0       94       19   0
NSP605   10 SPO-0           0        0        3        3   0
NSP695 SNA Host SHOW command processed
```

Table 5-6 describes the fields in the **SHOW FLOW** command output

Table 5-6 SHOW FLOW Fields

Field	Meaning
TNUM	Subtask number.
TASKNAME	Subtask name.
INPUTQ	Number of message buffers queued to the main task by this subtask.
OUTPUTQ	Number of message buffers queued to this subtask for processing.
INFLOW	Number of message buffers processed by this subtask.
OUTFLOW	Number of message buffers sent out by this subtask.
MC	Number of message buffers allocated to this subtask.

SHOW FREEQ

The **SHOW FREEQ** command displays the number of buffers used and available on the queue for each subtask.

Command Syntax

F NSPOPEN,SHOW FREEQ

Example

To display the buffer usage, enter the following command:

F NSPOPEN,SHOW FREEQ

Sample Output

```
NSP595 Command entered: SHOW FREEQ
NSP705 TNUM TASKNAME   FREE  USED
NSP705    0 MAINTASK   4000  0
NSP705    1 DISCOVER-0 4000  0
NSP705    2 MVS        1200  0
NSP705    3 PPI        2000  0
NSP705    4 STATUS-0   4000  0
NSP705    5 SERVER-0   2500  0
NSP705    6 SERVER-1   2500  0
NSP705    7 TCP-0      2500  0
NSP705    8 TCP-1      2500  0
NSP705    9 CMD-0      1500  0
NSP705   10 SPO-0      4000  0
NSP695 SNA Host SHOW command processed
```

Table 5-7 describes the fields in the **SHOW FREEQ** command output.

Table 5-7 SHOW FREQ Fields

Field	Meaning
TNUM	Subtask number.
TASKNAME	Subtask name.
FREE	Number of unused elements in the queue for this subtask.
USED	Number of used elements in the queue for this subtask.

SHOW TASK

The **SHOW TASK** command displays information about all the defined mainframe subtasks. As shown in Table 5-8, the data in the SPECIFIC column differs for each type of subtask.

Command Syntax

F NSPOPEN,SHOW TASK

Example

To display the mainframe subtasks, enter the following command:

F NSPOPEN,SHOW TASK

Sample Output

```
NSP595 Command entered: SHOW TASK
NSP600 TNUM TASKNAME   STATUS  RESTARTS/LIMIT  SPECIFIC
NSP600    1 DISCOVER-0 READY         0     10  NSPDSC1
NSP600    2 MVS        UP            0     10  NSPCONS2,03000002
NSP600    3 PPI        UP            0     10
NSP600    4 STATUS-0   UP            0     10  NSPSTA1
NSP600    5 SERVER-0   DOWN          0     10  NSPLU01
NSP600    6 SERVER-1   DOWN          0     10  NSPLU02
NSP600    7 TCP-0      READY         0     10  6106,6107
NSP600    8 TCP-1      CONNECTED     0     10  6126,6127
NSP600    9 CMD-0      UP            0     10  NSPCONS1,02000002
NSP600   10 SPO-0      UP            0     10  NSPSPO1
NSP695 SNA Host SHOW command processed
```

Table 5-8 describes the fields in the SHOW TASK command output.

Table 5-8 SHOW TASK Fields

Field	Meaning
TNUM	Subtask number.
TASKNAME	Subtask name.

Table 5-8 SHOW TASK Fields (continued)

Field	Meaning
STATUS	Status of subtask: INIT—Initialized. UP—Up and running. CONNECTED—Client workstation is connected. RUN—Discover subtask is running. READY— Discover subtask is ready to run, or TCP is ready for connection. DOWN—Subtask was taken down by the **KILL** command or has exceeded the maximum allowable automatic restarts. DOWNR—Subtask ended and is automatically restarting. QUIESCE—Subtask is ending but is waiting to free memory.
RESTARTS	Number of current automatic restarts for this subtask.
LIMIT	Maximum number of times subtask can be DOWNR before being DOWN.
SPECIFIC	Information specific to subtask type: CMD—Name and ID of the extended console. DISCOVER—VTAM secondary program operator access method control block (ACB) to which the discover subtask is connected. MVS—Name and ID of the extended console. PPI—No specific data. SERVER—LU name. SPO—VTAM secondary program operator ACB to which the SPO subtask is connected. STATUS—VTAM secondary program operator ACB to which the status subtask is connected. TCP—TCP/IP port number opened for each TCP/IP connection to a workstation.

SHOW TRACE

The **SHOW TRACE** command displays the current settings used for the **TRACE** command, which is described in the "TRACE" section on page 5-17.

Command Syntax

F NSPOPEN,SHOW TRACE

Example

To display the settings for current **TRACE** commands, enter the following command:

F NSPOPEN,SHOW TRACE

Sample Output

```
NSP595 Command entered: SHOW TRACE
NSP707 TNUM TASKNAME   NORMAL  DETAIL
NSP707    0 MAINTASK   ON      OFF
NSP707    1 DISCOVER-0 OFF     OFF
NSP707    2 MVS        OFF     OFF
NSP707    3 PPI        ON      OFF
NSP707    4 SEC        OFF     OFF
NSP707    5 STATUS-0   OFF     OFF
NSP707    6 TCP-0      ON      OFF
NSP707    7 CMD-0      OFF     OFF
NSP707    8 SPO-0      OFF     OFF
NSP695 SNA Host command processed
```

Table 5-9 describes the fields in the **SHOW TRACE** command output

Table 5-9 SHOW TRACE Fields

Field	Meaning
TNUM	Subtask number.
TASKNAME	Subtask name.

Table 5-9 SHOW TRACE Fields (continued)

Field	Meaning
NORMAL	Status of the **TRACE** command without the DETAIL option: ON—Normal tracing on. OFF—Normal tracing off.
DETAIL	Status of the **TRACE** command with the DETAIL option: ON—Detail tracing on. OFF—Detail tracing off.

SHOW VERSON

The **SHOW VERSION** command displays the version of the CiscoWorks Blue Maps and SNA View Mainframe program that is running, and the date and time that each mainframe subtask was compiled. Use this information to verify the level of code running on your system.

Command Syntax

F NSPOPEN,SHOW VERSION

Example

To display the application version, enter this command:

F NSPOPEN,SHOW VERSION

Sample Output

```
NSP595 Command entered: SHOW VERSION
NSP607 CiscoWorks Blue Maps and SNA View 3.0.1 Copyright 1996, 1997,
1998, 1999, 2000, 2001, 2002, 2003 Cisco Systems
NSP608 TASKNAME   DATE        TIME
NSP600 DISCOVER-0 Oct 15 1997 14:47:04
NSP600 MVS        Oct 15 1997 14:47:04
NSP600 PPI        Oct 15 1997 14:47:04
NSP600 STATUS-0   Oct 15 1997 14:47:04
NSP600 SERVER-0   Oct 15 1997 14:47:04
NSP600 SERVER-1   Oct 15 1997 14:47:04
NSP600 TCP-0      Oct 15 1997 14:47:04
NSP600 TCP-1      Oct 15 1997 14:47:04
NSP600 CMD-0      Oct 15 1997 14:47:04
NSP600 SPO-0      Oct 15 1997 14:47:04
NSP695 SNA Host SHOW command processed
```

Table 5-10 describes the fields in the **SHOW VERSION** command output

Table 5-10 SHOW VERSION Fields

Field	Meaning
TASKNAME	Subtask name.
DATE	Date that the load module for this task was compiled.
TIME	Time that the load module for this task was compiled.

STOP

The **STOP** command stops the mainframe application and its subtasks.

Command Syntax

F NSPOPEN,STOP

Example

To stop the mainframe application, enter the following command:

F NSPOPEN,STOP

TRACE

The **TRACE** starts and stops tracing for the mainframe main task and its subtasks. When you use the **TRACE** command to start tracing, the mainframe trace function records internal tracing events.

Command Syntax

[F NSPOPEN,]TRACE {OFF | PATH | INT} {*subtask_name* | MAIN} [DETAIL]

Syntax Description

F NSPOPEN is used when you issue the command from the command line. Omit this if you use the **TRACE** command from within the NSPPARM member.

OFF stops the specified mainframe tracing.

PATH starts path tracing. Path tracing records each entry to and exit from mainframe routines.

INT starts internal tracing. Internal tracing records each entry to and exit from mainframe routines, and records additional debugging information.

subtask_name specifies which mainframe subtask to trace. You can specify the name of a specific subtask (such as TCP-2), or you can specify just the root portion of a subtask name when several instances of that subtask type exist. For example, you could use the root name TCP to start tracing on all tasks with the same root TCP in their subtask name, or you could use the specific TCP-2 name. The valid subtask names are as follows:

- ALL—Starts tracing all subtasks and the main task.
- MAIN—Starts tracing just the main task.
- DISCOVER—Starts tracing just the DISCOVER subtask.
- MVS—Starts tracing just the MVS subtask.
- PPI—Starts tracing just the PPI subtask.
- PPO—Starts tracing just the PPO subtask.
- SERVER-[*n*]—Starts tracing the SERVER subtasks specified. Use the subtask name SERVER to trace all SERVER subtasks, or use the specific number of the subtask, such as SERVER-2.
- SPO-[*n*]—Starts tracing just the SPO subtasks specified. Use the subtask name SPO to trace all SPO subtasks, or use the subtask's specific number, such as SPO-2.

- STATUS—Starts tracing just the STATUS subtask.
- TCP- [*n*]—Starts tracing just the TCP subtasks specified. Use the subtask named TCP to trace all TCP subtasks, or use the subtask's specific number, such as TCP-2.

DETAIL includes traces of internal service routines. If you omit this operand, service routines are not traced.

Notes

You can use the **TRACE** command inside the *NSPPARM* member, as described below:

- Remove the *F NSPOPEN,* from the beginning of the **TRACE** command.
- Put the **TRACE** command in the *NSPPARM* file directly after the subtask card for the subtask it is tracing.

Examples

Step 1 To use the **TRACE** command interactively to start detail tracing of the TCP-1 subtask and all SERVER subtasks, enter the following commands:

F NSPOPEN,TRACE PATH TCP-1 DETAIL

F NSPOPEN,TRACE PATH SERVER DETAIL

Step 2 To use the **TRACE** command from within the *NSPPARM* member to start tracing of the DISCOVER subtask, put the following in *NSPPARM*:

DISCOVER NSPDSC1

TRACE INT DISCOVER

Step 3 To use the TRACE command from within the *NSPPARM* member to start internal detail tracing of the TCP subtask, put the following in *NSPPARM*:

TCP 6104 6104

TRACE INT TCP DETAIL

Troubleshooting the Mainframe Application

This chapter can help you to identify and resolve problems with the mainframe application. When you encounter an unfamiliar message or problem, refer to this chapter before you take any further action.

This chapter includes the following major sections:

- Subtask Problems, page 6-1 describes how to resolve failures that are attributable to mainframe subtask problems.
- Mainframe Messages, page 6-10 lists the meaning of each message generated by the mainframe application, and the appropriate user response.

Subtask Problems

This section describes corrective measures for mainframe subtask failures. Almost all subtask startup failures are accompanied by one or more messages written to the SYSLOG, or the output job log, or both. Check the list of mainframe messages in the "Mainframe Messages" section on page 6-10 for the course of action to take when an error message is displayed.

The following subsections describe the appropriate action to take when a failure of a specific subtask occurs:

- LU 6.2 Connection (SERVER Subtask), page 6-2

LU 6.2 Connection (SERVER Subtask)

A successful LU 6.2 connection requires exact compliance with the setup instructions given in this book and the *CiscoWorks Blue Maps and SNA View Workstation Installation and Administration Guide.* All network connections must be up and available, and all parameters used by both the mainframe and the workstation must match exactly.

If the LU 6.2 session fails to connect, perform the steps shown below.

Note Restart the mainframe application address space each time a modification is made to a data set or resource.

Step 1 Verify that the SYSIN deck has a SERVER parameter card with the following five parameters:

- VTAM APPL resource
- VTAM independent LU
- PARALLEL parameter
- Transaction program name for the host command interface (NSPOPNMS)
- Transaction program name for the host command server (NSPOPNCS)

Step 2 Verify that the VTAM APPL resource is defined in *SYS1.VTAMLST* with the parameter APPC=YES and that the resource is activated under VTAM.

Step 3 Verify that the VTAM LU is defined as an independent LU, either by a LOCADDR=0 definition under its PU, or as a CDRSC in a CDRSC major node definition. Verify that the LU is activated under VTAM.

Step 4 Use the VTAM APING command to verify that the workstation application is reachable from the mainframe:

D net,aping,id=*netid.resource*

Where:

netid.*resource* is the fully qualified name of the workstation. Until this command returns a positive response, the mainframe application cannot connect to the workstation.

Step 5 Stop the mainframe application's address space and check the bottom of the output job log for messages. The most significant messages are the primary and secondary return codes (RCPRI and RCSEC) from the LU 6.2 connection request.

Step 6 Check the job log for the first occurrence of message NSP111, NSP112, or NSP907. The RCPRI and RCSEC codes are fully documented in the IBM publication *VTAM Programming for LU 6.2*.

RCPRI/RCSEC combinations commonly encountered when installing the CiscoWorks Blue host component are listed in Table 6-1.

Table 6-1 Common RCPRI/RCSEC Combinations

RCPRI (hex)	RCSEC (hex)	Meaning and User Action
0	0	Connection completed successfully. No action required.
0	1	Connection completed successfully. No action required.
0	2	Connection completed successfully. No action required.
4	8	Workstation did not recognize the transaction program name sent from the mainframe. Check for a mismatch in the spelling of the fourth and fifth parameters of the SERVER SYSIN card against the spelling of the transaction program name profiles on the workstation.
4	9	Workstation recognizes the executable name but cannot start it. Verify that the executable file name specified in the workstation transaction program name profiles match the actual executable program name (*cwbstarttp*).

Table 6-1 Common RCPRI/RCSEC Combinations (continued)

RCPRI (hex)	RCSEC (hex)	Meaning and User Action
4	A	Check the file permissions and ownership of the */opt/CSCOcb/bin/cwbstarttp* workstation executable programs. It should be owned by "root" with the 'S' bit set. Restart SNA.
8	0	Verify that SNA and the SNA attachment are active on the workstation. If the workstation connection is Token Ring, verify that the 12-digit MAC address is entered correctly on the workstation and that the network identifier (NETID) matches the NETID in the mainframe data set *SYS1.VTAMLST(ATCSTRxx)*. Verify that the LU name in the second parameter of the SERVER SYSIN card matches the available VTAM LU.
8	1	Verify that the link station is active on the workstation. Verify that the VTAM APPL definition and independent LU (specified in the first and second parameters of the SERVER SYSIN card) are active.
10	2	VTAM is attempting to process two requests of the same mode type on the same LU. This problem is frequently caused by multiple restart attempts within a short period of time. Stop the SERVER subtask, wait 30 seconds, and then restart the subtask.
14	0	Transaction program on the workstation terminated abnormally. Wait until all ports on the workstation are cleared. Before retrying the connection, enter the **ps -ef | grep cwbh** command on the UNIX workstation to verify that no extraneous processes are running.
2C	1	Logmode name specified in the third parameter of the SERVER SYSIN card was not recognized on the workstation. If you receive an NSP112 message to indicate that the SNASVCMG CNOS succeeded, but the PARALLEL CNOS failed, verify that SNA and the SNA attachment are active on the workstation.
48	0	LU 6.2 conversation was terminated prematurely. Check to see if SNA was stopped on the workstation.

Table 6-1 Common RCPRI/RCSEC Combinations (continued)

RCPRI (hex)	RCSEC (hex)	Meaning and User Action
4C	0	Session used for the LU 6.2 conversation is terminated, possibly because of a line or modem failure. Check for a break in the physical connection between the workstation and the mainframe.
98	0	Overload of commands with large replies is causing a memory shortage while sending data.

TCP/IP Connection (TCP Subtask)

Setting up a TCP/IP connection is less complex than setting up an LU 6.2 connection, but requires a TCP/IP package installed on the mainframe. If a TCP/IP stack is not available, you must use an LU 6.2 connection instead.

A common problem with the TCP/IP connection to the IBM TCP/IP protocol stack is the inability of the TCP subtask to communicate with IBM's TCP/IP package. Usually, the name of either the TCP/IP address space, or the TCP/IP data sets, has been changed from the default. When the mainframe application is unable to find the TCP/IP address space or the TCP/IP profile data sets, it usually generates the following error message:

```
NSP150  TCP/IP communications: socket() for workstation message agent
failed with errno 39
```

If this message is displayed during startup, ensure that all steps required to configure TCP/IP were performed.

If the TCP subtask starts but is unable to connect, check that the ports on the TCP card match the ports in the */etc/svopen_config_DOMAINNAME* file.

If the problem persists, call the Cisco TAC.

PPI Subtask with NetView

The most common cause for failure of the PPI subtask is that the NetView subsystem interface is not active. This failure results in the display of an NSP091 message with a return code from NetView. The possible return code values for the

NSP091 message are listed in an appendix in the *NetView Application Programming Guide*. For information about the NSP091 message, see the "Mainframe Messages" section on page 6-10.

Check the startup JCL to ensure that the STEPLIB includes the NetView load library (loadlib) that contains the CNMCNETV load module.

Also, check the SYSLOG during NetView startup for messages indicating a failure of the NSPVPPI optional task that must be running in the NetView address space.

PPI Task Failure under NetView

The NSPVPPI optional task under NetView starts automatically while NetView is initializing. A successful initialization of the NSPVPPI task generates both the NSP302 and an NSP311 message. If the NSPVPPI task fails to initialize, additional NSP*xxx* messages are displayed during NetView initialization. Check the list of messages in this chapter to determine why the NSPVPPI task failed to initialize.

If the NetView subsystem interface is not active when NetView starts, the NSPVPPI task will not load, and the following message is issued from NSPVPPI before it terminates:

```
NSP313  NSPVPPI  : NETVIEW INTERFACE FAILURE, RC=0018
```

To avoid this problem, (and avoid restarting the NSPVPPI task manually), start the subsystem interface address space before starting the NetView address space.

If you have not run the subsystem interface before, review the *NetView Installation Guide* for instructions on subsystem interface setup, paying special attention to the following:

- The subsystem interface address space must start with the same four characters as the NetView address space. For example, if the NetView address space is named "NETVIEW", then the first four characters of the subsystem interface address space must be "NETV".
- The four-character subsystem name must be listed in the SYS1.PARMLIB(IEFSSN*xx*) member. Any change to this member requires an IPL.
- You must define the CNMCSSIR task in NetView's DSIDMN member. CNMCSSIR is defined with INIT=N.

After the problem is corrected, restart the NSPVPPI task from the NetView operator session using the **START TASK=NSPVPPI** command.

Data Gathering Errors

The Status/Discovery process at the workstation receives updated PU information from the mainframe when a status change occurs or when a new PU is discovered. If status updates are not received at the workstation, you should determine whether the message is flowing from VTAM to the mainframe application. You can use the following methods to help diagnose the problem.

- Use the **F NSPOPEN,SHOW TASK** command to ensure that the PPI or PPO subtask and the STATUS subtask are active.
- Use the **F NSPOPEN,SHOW FLOW** command to check the INFLOW and OUTFLOW values for the PPI or PPO subtask and for the STATUS subtask. Those values should be greater than 0 and should increment each time a status change occurs.

If the PPI or PPO INFLOW and OUTFLOW values are incrementing, but the STATUS INFLOW or OUTFLOW values are not incrementing, check the **PULU_FILTER** and **INCLUDE_SW_MAJORNODES** and **EXCLUDE_SW_MAJORNODES** configuration statements.

If the PPI or PPO INFLOW and OUTFLOW values are not incrementing, use the following procedure:

Step 1 Ensure that the VTAM PPOLOG parameter is set to yes (**MODIFY vtamproc,PPOLOG=YES**) as documented in the "Updating MVS and VTAM" section on page 4-9 in Chapter 4, "Updating the Mainframe Application Software."

Step 2 Ensure that the NetView NSPVPPI DST (data services terminal) and NSPAUTO1 autotask are active:

```
list nspvppi
list nspauto1
```

If you used a different name for the autotask, use it here instead of NSPAUTO1.

Step 3 Ensure that the NSPSVTBL member is included in the automation table and that the automation table is reloaded.

Step 4 Issue the NetView **DISBQL ALL** command to check PPI connectivity to the mainframe task.

Using Mainframe Commands for Troubleshooting

You can use some of the mainframe commands described in the "Issuing Mainframe Commands" section on page 5-3 to help debug the mainframe task.

Using the SHOW ADDR Command

The **SHOW ADDR** command displays important subtask information that might be required by the Cisco TAC.

Using the SHOW FLOW Command

The **SHOW FLOW** command displays several key fields for troubleshooting, as shown inTable 6-2

Table 6-2 SHOW FLOW Command Output

Field	Purpose
TASKNAME	Identifies the subtask by its name.
INPUTQ	Number of message buffers queued to the main task by this subtask. If this number grows rapidly and continues to stay high, there might be a problem with the subtask or the main task.
OUTPUTQ	Number of message buffers queued to this subtask for processing. If this number grows rapidly and continues to stay high, there might be a problem with the subtask.
MC	Number of message buffers allocated to this subtask. If this number grows rapidly and continues to stay high, there might be a problem with the subtask.

Using the SHOW FREEQ Command

The **SHOW FREEQ** command displays the number of elements in the queue for each subtask. Look at the number in the FREE column. If the number in the FREE column goes to 0, there is a problem with the subtask.

Using the SHOW TASK Command

Use the **SHOW TASK** command to ensure that the PPI or PPO subtask and the STATUS subtask are active.

Workstation Overrun by Mainframe

The mainframe component can send PU/LU status updates and discovery data. Consequently, the workstation must be able to process this data as fast as it is sent to the workstation.

If the workstation's performance is not fast enough (for example, if it is an overloaded machine), the workstation may be unable to keep up with the mainframe.

Here are some symptoms of this problem that you might see:

- Commands from the workstation fail to receive responses.
- The status that the workstation displays for a PU or LU is inaccurate for long periods of time even though the view is refreshed.

If you see these symptoms, issue the following mainframe command:

F NSPOPEN,SHOW FLOW

If there are large numbers in the INPUTQ or OUTPUTQ columns of the transport subtask you are using (TCP or SERVER), your workstation probably is not powerful enough to keep up with the traffic. You could also lengthen the polling cycles to free up some of the CPU to get more processing time.

Mainframe Messages

This section provides information about the messages generated by the mainframe application. The messages are described in numerical order.

NSP001 *RCVSYN* returned a *RPLRTNCD* of number

Where:

number is an return code from *RCVSYN*.

Explanation An internal error has occurred in *NSPOPEN* application.

Action Notify the Cisco TAC.

NSP002 *type* SUBTASK INITIALIZED

Where:

type is the subtask type.

Explanation The mainframe subtask has successfully initialized. If the subtask has an associated VTAM access method control block (ACB), the ACB information is provided. Task-specific information also is given. Processing continues.

Action No user action is required.

NSP003 *FUNCTION* returned with rc = *NUMBER*

Explanation This message is used to issue unexpected errors from internal function to the system log and job stream.

Action Notify the Cisco TAC.

NSP004 *RCVSYN* received an unexpected *RPLFDB2* of NUMBER

Where:

NUMBER is the return code from *RCVSYN*.

Explanation An unexpected return code was issued from the *RCVSYN* function.

Action Notify the Cisco TAC.

`NSP005 COULD NOT OPEN VSAM FILE WITH DDNAME DDNAME.`

Where:

DDNAME is the name of the VSAM file used to store PU data; it is either *XIDDATA* or *XIDBACK*.

Explanation You issued a **SHOW DLC** command, but the VSAM file named *DDNAME*, used to store PU information for the mainframe task (either *XIDDATA* or *XIDBACK*), could not be opened.

Action Determine why the file cannot be opened. The XIDDATA or XIDBACK DD cards might be incorrect.

`NSP006 Invalid length %d in message received`

Explanation An invalid messages length was specified for a message sent to *NSPOPEN*.

Action Notify the Cisco TAC.

`NSP007 Lost connection with the workstation`

Explanation This message issued when the connection to a workstations broken. This message is followed by additional message describing the effected workstation.

Action No user action is required.

`NSP008 INVALID INPUT PARAMETER CARD ON LINE number`

Where:

number is the line number of SYSIN.

Explanation The mainframe application read an invalid card from SYSIN. The invalid card is skipped. Processing continues with the next SYSIN card.

Action Correct the input card on the given line *number* of SYSIN. Chapter 4, "Updating the Mainframe Application Software," provides valid values for task cards. Ensure that all other lines begin with an asterisk (*), which indicates that they are comment lines.

`NSP009 DUPLICATE subtask CARD ON LINE number IGNORED`

Where:

subtask is the type of the subtask.

number is the line number of SYSIN.

Explanation The mainframe application read a definition card from SYSIN for a subtask name, which was already defined. The duplicate card is skipped. Processing continues with the next SYSIN card.

Action Correct or remove the duplicate input card on the given line number of SYSIN. See Chapter 4, "Updating the Mainframe Application Software," for names of input parameter cards that might be defined multiple times.

`NSP010 MAXIMUM NUMBER OF subtask CARDS REACHED; IGNORING LINE number`

Where:

subtask is the type of the subtask: SERVER, SPO, TCP, or SUBTASK.

number is the line number of SYSIN.

Explanation The mainframe application has reached the maximum number of subtasks of the type named in that field. The definition card on the line specified by the *number* field will not be processed. The subtask field identifies the type of subtask whose maximum has been reached: SERVER, SPO, or TCP. If the subtask field contains the word "SUBTASK", the mainframe program has reached the maximum allowed subtasks that can be defined; consequently, all SYSIN parameter cards from that line number forward will be ignored. Processing continues.

Action Decrease the number of SYSIN parameter cards of the type named.

`NSP011 Invalid msg/command type %d received by task %s`

Explanation An internal error has occurred in the *NSPOPEN* application.

Action Notify the Cisco TAC.

`NSP012 Invalid command COMMAND received by task TASK`

Explanation An internal error has occurred in the *NSPOPEN* application.

Action Notify the Cisco TAC.

`NSP013 Enqueue failed for TASK OPERATION rc RC`

Explanation An internal error has occurred in the *NSPOPEN* application.

Action Notify the Cisco TAC.

`NSP014 Size:BYTES of message too large, max:MAX`

Explanation An internal error has occurred in the *NSPOPEN* application.

Action Notify the Cisco TAC.

`NSP015 Error in retrieving VTAM data, location ADDRESS`

Explanation An internal error has occurred in the *NSPOPEN* at the *ADDRESS* specified in message.

Action Notify the Cisco TAC.

`NSP016 Enqueue failed for TASK OPERATION rc RC or NSP016 Dequeue failed for TASK OPERATION rc RC`

Explanation An internal error has occurred in the *NSPOPEN* application.

Action Notify the Cisco TAC.

NSP017 NO DATA FOUND MATCHING REQUESTED KEY

Explanation You issued a **SHOW DLC** command, but no data was available for the requested PU name.

Action No user action is required.

NSP018 VTAM ACB GENERATION FOR *subtask acb* FAILED, RC = *rcnumber*

Where:

subtask is the type of the subtask.

acb is the name of the failing ACB.

rcnumber is the return code from the Get VTAM ACB routine.

Explanation An initializing subtask attempted to get a VTAM ACB and failed. The subtask terminates.

Action Verify that the ACB named in *acb* is active. Use the **INIT** command to reinitialize the subtask.

NSP019 VTAM *subtask* OPEN FOR *acb* FAILED, RC = *rcnumber*, ERROR = *enumber*

Where:

subtask is the type of the subtask.

acb is the name of the failing ACB.

rcnumber is the return code from the Open VTAM ACB routine.

enumber is the error code within the ACB.

Explanation An initializing subtask attempted to open a VTAM ACB and failed. The subtask terminates.

Action Verify that the ACB is correctly defined and not already in use. See the IBM *VTAM Programming* manual for descriptions of the return code and error code.

NSP020 *subtask* IS CURRENTLY IN USE

Where:

subtask is the type of the subtask.

Explanation This message immediately follows the NSP019 message if an exclusive subtask ACB is in use by another program. The mainframe subtask terminates.

Action Verify that the ACB is not being taken by another program on the host, such as NetView or SOLVE:Netmaster.

NSP021 UNSOLICITED *msgtype* DATA IS UNAVAILABLE

Where:

msgtype is the type of the message.

Explanation This message follows the NSP019 message to alert you that the mainframe program is not able to receive unsolicited data because it is unable to access an ACB. The subtask terminates.

Action Verify that the ACB is defined.

NSP022 *LEVEL* trace enabled for *SUBTASK*

Where:

LEVEL is Detail or Normal.

SUBTASK is name of subtask that has been marked for tracing

Explanation Tracing is starting on the specified class.

Action No user action is required.

`NSP023 %s trace disabled for %s`

Where:

SUBTASK is name of subtask that has been marked.

Explanation Tracing is stopped on the specified class.

Action No user action is required.

`NSP024 No task matching name SUBTASK found`

Where:

SUBTASK is name of subtask that has been marked.

Explanation The task specified for the previous command cannot be found.

Action Verify the subtask name is specified correctly and `re-issue the command.`

`NSP026 UNEXPECTED subtask RETURN CODE, RC = rcnumber`

Where:

subtask is the type of the subtask.

rcnumber is the return code from the Receive routine.

Explanation The subtask Receive routine received an unexpected return code while attempting to receive messages. The subtask terminates.

Action Use the **INIT** command to reinitialize the subtask.

`NSP027 Memory allocation failed`

Explanation *NSPOPEN* was unable to allocate required storage.

Action Increase the amount of storage specified for the *NCPOPEN* application. If this problem persists notify the Cisco TAC.

`NSP028 Task going down, message not queued`

Explanation One of the subtask is terminating. This message indicates that information sent to task will be lost.

Action No action required. This is a diagnostic message that can be used to determine cause of other problems that will be indicated with their own error messages.

`NSP029 Attempt to free %p when counter %d`

Explanation An internal error has occurred in the *NSPOPEN* application.

Action Notify the Cisco TAC.

`NSP030` *`task`* `COULD NOT ADD ADDRESS TO` *`task2`* `FREE QUEUE (`*`count`*`)`

Where:

`task1` is the name of the subtask attempting to add a message to another subtask's free queue.

`task2` is the name of the subtask whose queue is being added to.

`count` is the number of times the attempt has failed.

Explanation An attempt by one subtask to add to another subtask's free queue has failed. The request is retried.

Action No action is required. This is a diagnostic message, which can be used to determine the cause of other problems that are indicated by their own error messages.

`NSP031 Attempt to free a message for invalid task %p`

Explanation An internal error has occurred in the *NSPOPEN* application in the *ADDRESS* specified in message.

Action Notify the Cisco TAC.

`NSP032 Task %s failed to free memory`

Explanation An internal error has occurred in the NSPOPEN application.

Action Notify the Cisco TAC.

`NSP033 SNA Host COMMAND = commandtext`

Where:

commandtext is a mainframe command.

Explanation The command text issued via the mainframe application is logged to SYSLOG and processing continues.

Action No user action is required.

`NSP034 INITIALIZATION OF SPO name FAILED IN reqtype PROCESSING, RC1 = addr RC2 = size`

Where:

name is the name of the secondary program operator (SPO) subtask.

reqtype is the type of the request being processed.

addr is the returned address from the Get RPL routine.

size is the returned size from the Get RPL routine.

Explanation The SPO subtask failed while calling the VTAM Get RPL routine. The SPO subtask terminates.

Action Use the **INIT** command to reinitialize the subtask.

`NSP035 SPO WARNING: FAILURE RETRIEVING COMMAND RESPONSES, MAX RETRIES REACHED`

Explanation The mainframe application's secondary program operator (SPO) interface subtask encountered a failure while attempting to retrieve the command responses from an issued VTAM command. All responses were not retrieved. Processing continues.

Action Reissue the VTAM command. If the proper responses still are not returned, contact the Cisco TAC.

`NSP036 Failure in obtaining rif data, location` *`NUMBER`*

Where:

NUMBER is a numeric value used by *NSPOPEN* application to mark failure points in application.

Explanation An error has occurred while retrieving rif data for a Physical Unit.

Action No action required. This is a diagnostic message that can be used to determine cause of other problems that will be indicated with their own error messages.

`NSP037 Wrong` *`MDB`* `type` *`TYPE`* `received, expected` *`ETYPE`*

Where:

TYPE is the *MDB* type received from MVS

ETYPE is the *MDB* type expected to be received from MVS.

Explanation An internal error has occurred in the *NSPOPEN* application.

Action No action required. This is a diagnostic message that can be used to determine cause of other problems that will be indicated with their own error messages.

NSP038 VTAM COMMAND SUPPORT UNAVAILABLE

Explanation The mainframe application could not process the VTAM command because there are no active secondary program operator (SPO) subtasks. The command is discarded.

Action If command support from the workstation is desired, restart the mainframe application with an SPO card in the SYSIN deck that identifies a valid VTAM APPL resource to be used as an SPO. Verify that the named VTAM resource is active. Also, you might code a SEC NO parameter card.

NSP039 NODE *node_name* REMAINS IN PENDING STATE

Where:

node_name is the node name of a PU.

Explanation The mainframe application detected a PU (*node_name*) in the pending state after VTAM message IST590I indicated that the connection was established. The mainframe application then tried to display the PU within the time defined by the *delay_time* option of the STATUS parameter card, but failed.

Action Increase the value of the *delay_time* option on the STATUS parameter card.

NSP040 Resource *RESOURCE* not found in database:*VSAM RC* = *RC* errno = *ERRNO*

Where:

RESOURCE is the name of a physical unit.

VSAM is the *VSAM* data set name or *DD* statement defining the *VSAM* dataset used by *NSPOPEN*.

RC is the return code from the call to retrieve data.

ERRNO is the system provided error number.

Explanation The NSPOPEN application was unable to locate information for a physical unit.

Action Verify the VTAM ISTEXCCS exit has been activate and is operational. If this problem persists, contact Cisco TAC.

`NSP041 Unable to obtain DLC information for RESOURCE`

Where:

RESOURCE is the name of a physical unit.

Explanation The NSPOPEN application was unable to locate DLC information for a physical unit.

Action Verify the VTAM ISTEXCCS exit has been activate and is operational. If this problem persists, contact Cisco TAC.

`NSP042 ONLY_SWITCHED_PUS not supported for this VTAM level`

Explanation The level of VTAM currently running does not support the following format of the display clusters command.

d net,clstrs,idtype=swseg,max=*

The NSPOPEN application may process non-switch physical units.

Action None.

`NSP043 SPO card ignored since SEC NO not specified`

Explanation You did not provide a SEC NO parameter card, so SNA Resources cannot be activated or deactivated from the workstation. You also provided an SPO parameter card, which specifies a secondary program operator for activating and deactivating SNA resources from the workstation. The SPO card is ignored.

Action If you want to allow SNA Resources to be activated or deactivated from the workstation, provide a SEC NO parameter card.

`NSP044 [PPO|PPI] already specified. You cannot specify both PPO and PPI`

Explanation You supplied both the PPO and PPI parameter cards. Only one can be specified.

Action Determine whether or not an existing product on the mainframe is defined as the Primary Program Operator (PPO) (such as NetView or Solve:NetMaster). If a PPO already exists, then code the PPI card and ensure that you accomplished the correct NetView or Solve:Netmaster customization. If no product is currently defined as the PPO, then code the PPO card. These definitions are contained in the parameter file.

`NSP045 subtask_name card was not specified and is required`

Explanation The indicated subtask card (*subtask_name*) was not specified and is a required card.

Action Add a correctly coded entry for this card to the parameter file.

`NSP046 subtask_name card was not specified and is recommended`

Explanation The indicated subtask card (*subtask_name*) was not specified and is a recommended card.

Action If you understand the implications of not coding this card, then this message can be ignored. If you accidently omit the card, you must add a correctly coded card to the parameter file.

`NSP047 No PPI or PPO card was specified. One or the other is required`

Explanation Neither the PPI or PPO card was specified. You have to specify one or the other.

Action Determine whether or not an existing product on the mainframe is defined as the Primary Program Operator (PPO) (such as NetView or Solve:NetMaster). If a PPO already exists, then code the PPI card and ensure

that you accomplished the correct NetView or Solve:Netmaster customization. If no product is currently defined as the PPO, then code the PPO card. These definitions are contained in the parameter file.

`NSP048 No TCP or SERVER card was specified. At least one is required`

Explanation No TCP or SERVER cards were specified. You need to configure at least one of these in order for a workstation to connect to the mainframe server to acquire the VTAM SNA information.

Action Determine what connectivity you would like to use (TCP/IP or LU 6.2) and code the appropriate card.

`NSP049 ERROR OPENING FILE` *`file`*

Where:

`file` is the name of the member that could not be opened.

Explanation The application could not open the requested file. You might receive this message with the member name being that of the HOSTPU for the VTAM, this is not a problem.

Action If the error is received for any member name other than the host VTAM, contact the Cisco TAC.

`NSP050` *`task`* `COULD NOT FREE MESSAGES (`*`count`*`)`

Where:

`task` is the name of the subtask that is attempting to free messages

`count` is the number of times the operation has been attempted

Action No action is required. This is a diagnostic message, which can be used to determine the cause of other problems that are indicated by their own error messages.

NSP051 NO PARAMETERS ALLOWED ON THE STOP COMMAND. TRY KILL

Explanation The **STOP** command was issued with a parameter. This is not a valid command.

Action Enter the **STOP** command without any parameters to stop the address space. If you are attempting to stop only one subtask, use the **KILL** command.

NSP091 PPI INITIALIZATION FAILED, STEP = *stepnum* RC = *rcnumber*

Where:

stepnum is the step of initialization that failed:

- 1 indicates that a subsystem interface is not active.
- 2 indicates that an attempt to get an address space control block (ASCB) value failed.
- 3 indicates that an attempt to register as a receiver failed.

rcnumber is the return code from call to NetView (CNMNETV).

Explanation An attempt by the primary program operator subtask to access the CNMNETV module failed. The PPI subtask terminates.

Action Table 6-3 shows you what to do depending on the value of *stepnum:*

Table 6-3 Action Depending on stepnum Value

stepnum	Action
1	Check the status of the subsystem interface.
2	Check the status of the NetView or SOLVE:Netmaster program-to-program interface. If it is inactive, activate it and restart the PPI subtask.
3	Verify that no other application is attached to the NetView or SOLVE:Netmaster program-to-program interface.

`NSP095 SNA Host PPI BUFFER SIZE ERROR, RC = rcnumber`

Where:

rcnumber is the return code from the PPI call.

Explanation A Receive request for the PPI failed due to the length of the buffer size. The PPI subtask terminates.

Action Use the **INIT** command to reinitialize the subtask.

`NSP096 SNA Host PPI INTERFACE FAILED, ID = requestid, RC = rcnumber`

Where:

requestid is the ID of the task request.

rcnumber is the return code from the PPI call.

Explanation A Receive request for the PPI failed. The PPI subtask terminates.

Action See the *NetView Application Programming Guide* to determine the meaning of the given return code.

`NSP111 subtask CNOS mode FAILED, RC = rcnum RCPRI = primaryrc RCSEC = secondaryrc`

Where:

subtask is the name of the subtask.

mode is the mode name (SNASVCMG or PARALLEL).

rcnum is the return code from the CNOS call.

primaryrc is the primary return code from the CNOS request.

secondaryrc is the secondary return code from the CNOS request.

Explanation The LU 6.2 CNOS command for *mode* failed during the initialization of the *subtask*. The mainframe subtask terminates.

Action See the publication *VTAM Programming for LU 6.2* for the meaning of the RCPRI and RCSEC return codes.

Note The RCPRI and RCSEC codes are valid only if *rcnum* = B0000. Depending on the codes, you might need to make a correction to a workstation profile or the mainframe VTAM definitions. A dump of the VTAM request parameter list (RPL) control block follows this message. Restart the mainframe application to recover the subtask.

NSP112 *subtask* CNOS FOR *mode status* (RCPRI=*primaryrc* RCSEC=*secondaryrc*)

Where:

subtask is the name of the subtask.

mode is the mode name (SNASVCMG or PARALLEL).

status is the status of the task; it is either "succeeded" or "failed."

primaryrc is the primary return code from the CNOS request.

secondaryrc is the secondary return code from the CNOS request.

Explanation The LU 6.2 CNOS command for *mode* is completed for *subtask*. Processing continues.

Action If status is "failed," stop the mainframe application and check the output job log for more information. See the "Configuring LU 6.2 Connectivity" section on page 4-2. Also, see the publication *VTAM Programming for LU 6.2* for a translation of the RCPRI and RCSEC return codes.

NSP113 UNABLE TO ESTABLISH *msgtype* LU 6.2 COMMUNICATION WITH THE WORKSTATION *wsagent* AGENT, RC = *rcnumber*

Where:

msgtype is the type of LU 6.2 message. The following message types are possible:

- outbound—Allocation from the mainframe to the workstation.
- inbound—Allocation from the workstation to the mainframe.

- send—Message transmission from the mainframe to the workstation.
- receive—Message transmission from the workstation to the mainframe.

wsagent is the workstation agent. The following agents are possible:

- command—Workstation command agent.
- message—Workstation message agent.

rcnumber is the return code from the LU 6.2 call.

Explanation Establishment of LU 6.2 sessions between the mainframe and the workstation failed. The SERVER subtask terminates.

Action Verify that the parameters of the SERVER SYSIN card match the VTAM resource names defined for the mainframe application, and that they match the resource names entered in the workstation communications setup.

`NSP114 ERROR IN LU 6.2 direction WORKSTATION wsserver SERVER, RC = rcnumber`

Where:

direction is the direction of the message. The following directions are possible:

- Send to—Message flows from the mainframe to the workstation.
- Receive from—Message flows from the workstation to the mainframe.

wsserver is the workstation server. The following servers are possible:

- command—Host command server.
- HCI—Host connection interface.

rcnumber is the return code from the LU 6.2 call.

Explanation An error occurred in the sending or receiving of an LU 6.2 message between the mainframe and the workstation. The SERVER subtask terminates.

Action Verify the status of the LU 6.2 sessions between the mainframe and the workstation.

NSP115 DUMPING *subtask direction action ctrlblk*

Where:

subtask is the subtask name.

direction is the direction of the message. The following directions are possible:

- outbound—From the mainframe to the workstation.
- inbound—From the workstation to the mainframe.

action is the attempted action. The following actions are possible:

- allocate—Allocation of an LU 6.2 conversation.
- receive—Sending a message over an established conversation.

ctrlblk is the control block being dumped. The following control blocks are possible:

- RPL—VTAM Request Parameter List.
- RPL6X—VTAM Request Parameter List Extension.

Explanation The mainframe application subtask encountered an error in LU 6.2 communications to the workstation. Following this message, a dump of the control block specified by *ctrlblk* is given to aid in problem determination. The SERVER subtask terminates.

Action See the *VTAM Programming* manual for a mapping of the control block. Correct the error and use the **INIT** command to reinitialize the subtask.

NSP119 *num* MESSAGES QUEUED ON *subtask*. COMMAND REJECTED: *command*

Where:

num is the number of messages.

subtask is the subtask name.

command is the command entered.

Explanation The subtask will not process the command issued from the workstation because of a backlog of messages to be sent to the workstation. The command is discarded. Processing continues on the remaining messages in the subtask's queue.

Action Wait until the existing backlog of messages is processed and then reissue the command. Use the **SHOW TASK** command to view the number of messages in the subtask's output queue.

`NSP121 MVS CONSOLE name COULD NOT OBTAIN A MIGRATION ID`

Where:

name is the name of the console to be defined.

Explanation The MVS console being defined requested a 1-byte migration ID, but the console initialization routine was unable to provide one. Initialization of the console continues.

Action No user action is required.

`NSP122 MVS CONSOLE name INITIALIZATION FAILED, RC = rcnumber`

Where:

name is the name of the console to be defined.

rcnumber is the return code from the initialization routine.

Explanation The initialization of the MVS console failed. The MVS subtask terminates.

Action Record the return code and report it to the Cisco TAC.

NSP150 TCP/IP COMMUNICATIONS: *function* FOR WORKSTATION *component* AGENT FAILED WITH ERRNO *value*

Where:

function is the failing communication function.

component is the workstation component that detected the failure.

value is the integer error value.

Explanation A TCP/IP communication error has occurred. The error might have taken place during TCP/IP communication setup, or during a send or receive between the mainframe and the agent specified. The TCP subtask terminates.

Action Verify the availability of TCP/IP communication between the workstation and the mainframe. Use the **INIT** command to reinitialize the subtask.

NSP151 SNA Host FAILURE IN COMMUNICATION TO TCP/IP

Explanation The mainframe application received an error while attempting to receive data from a TCP/IP socket or event control block (ECB). The TCP/IP subtask terminates.

Action Use the **INIT** command to reinitialize the subtask.

NSP160 CONSOLE COMMAND RETURN CODE = *rcnumber*

Where:

rcnumber is the return code from the **SEND** command subroutine.

Explanation An MVS command request completed with a non-zero return code. Processing continues.

Action If an expected command response is not received, record the return code and contact your support representative.

NSP205 MVS CONSOLE *name* REACHED MEMORY LIMIT. DATA LOST

Where:

name is the console name.

Explanation The extended console defined for the mainframe application has filled all the available cells in the data space. The incoming message is not queued. Processing continues.

Action Check the status of the extended console with the following command:

DISPLAY CONSOLES, CN=*name*

If messages do not resume queuing to the extended console, restart the mainframe application to ensure that the console comes down cleanly. You might need to define a new console with a larger message data space.

NSP206 MVS CONSOLE *name* REACHED QUEUE LIMIT. DATA LOST

Where:

name is the console name.

Explanation The extended console defined for the mainframe application has reached its maximum queue depth. The incoming message will not be queued. Processing continues.

Action Check the status of the extended console with the following command:

DISPLAY CONSOLES, CN=*name*

If messages do not resume queuing to the extended console, restart the mainframe application, ensuring that the console comes down cleanly.

NSP207 MVS CONSOLE *name* STOPPED BY INTERNAL ERROR

Where:

name is the console name.

Explanation The extended console defined for the mainframe application received an error during the processing of its message queues. The mainframe application deactivates the MVS console.

Action Check status of extended console with the following command:

DISPLAY CONSOLES, CN=*name*

Restart the mainframe application.

NSP208 MVS CONSOLE *name* REACHED ALERT PERCENTAGE

Where:

name is the console name.

The number of messages queued to the extended console for the mainframe application has reached a certain percentage of the maximum queue depth. Processing continues.

Action Verify that messages are being displayed on the MVS messages client on the workstation. Check status of extended console with the following command:

DISPLAY CONSOLES, CN=*name*

If the queue shortage is not relieved quickly, stop and restart the mainframe PROC.

NSP209 MVS CONSOLE *name* SUSPENDED BY REQUEST

Where:

name is the console name.

Explanation An error occurred in the extended console defined for the mainframe application, and MVS has requested the deactivation of the console. The mainframe application deactivates the console.

Action Check the status of the extended console with the following command:

DISPLAY CONSOLES, CN=*name*

Restart the mainframe application.

NSP210 MVS CONSOLE *name* ALERT ECB POSTED FOR UNKNOWN REASON

Where:

name is the console name.

Explanation The extended console defined for the mainframe application was posted with an alert indicating a problem, but no error flags are set in the console's status area. Processing continues.

Action Check the status of the extended console with the following command:

DISPLAY CONSOLES, CN=*name*

NSP302 *name*: SNA Host PPI TASK INITIALIZED

name is the name of the NetView program-to-program interface (PPI) subtask.

Explanation The PPI subtask for the mainframe application was successfully initialized in the NetView address space. Processing continues.

Action No user action is required.

NSP303 *name*: SNA Host PPI TASK TERMINATED

Where:

name is the name of the NetView PPI subtask.

Explanation The PPI task is terminated in the NetView address space. Processing continues, but the mainframe application no longer receives unsolicited VTAM messages or alerts from NetView.

Action If NetView has terminated, restart NetView. If only the PPI subtask has terminated, restart the subtask from a NetView operator session with the following command:

START TASK=*name*

NSP304 *name*: DSIFRE FAILED FOR USER STORAGE

Where:

name is the name of the NetView PPI subtask.

Explanation The PPI subtask received an error return code from the NetView DSIFRE macro while attempting to free the 4-KB work area of memory during subtask shutdown. Subtask shutdown processing continues.

Action Notify the system programmer that a potential memory leak exists in the currently running NetView.

NSP305 *name*: DSIFRE FAILED FOR QUEUED STORAGE

Where:

name is the name of the NetView PPI subtask.

Explanation The PPI subtask received an error return code from the NetView macro DSIFRE while attempting to free all remaining subtask memory during subtask shutdown. Subtask shutdown processing continues.

Action Notify the system programmer that a potential memory leak exists in the currently running NetView.

NSP306 *name*: DSIFRE FAILED FOR MQS BUFFER

Where:

name is the name of the NetView PPI subtask.

Explanation The PPI subtask received an error return code from the NetView DSIFRE macro while attempting to free the memory allocated for the private message queue. Processing continues.

Action Notify the system programmer that a potential memory leak exists in the currently running NetView.

NSP307 *name*: DSIGET FAILED FOR USER STORAGE

Where:

name is the name of the NetView PPI subtask.

Explanation The PPI subtask failed to get a 4 KB block of memory for use during processing. The task termination flag is set.

Action Notify the system programmer that a potential memory shortage exists in the currently running NetView. You might need to increase the region size of the NetView address space.

NSP308 *name*: ENQ ERROR

Where:

name is the name of the NetView PPI subtask.

Explanation An enqueue request (ENQ) on the NetView's task vector block (TVB) chain failed. If the process is not already terminated, the task termination flag is set.

Action Notify the system programmer. Restart the PPI subtask.

NSP309 *name*: DEQ ERROR

Where:

name is the name of the NetView PPI subtask.

Explanation A dequeue request (DEQ) on the NetView's task vector block (TVB) chain failed. If the process is not already terminated, the task termination flag is set.

Action Notify the system programmer. Restart the PPI subtask.

NSP310 *name*: TASK ALREADY EXISTS

Where:

name is the name of the NetView PPI subtask.

Explanation The PPI subtask attempted to add itself to the NetView TVB chain, but found another task with the same name already on the chain. The task termination flag is set.

Action Verify that another instance of the subtask is not already running under this NetView. Restart the PPI subtask.

NSP311 *name*: LOAD OF CNMNETV COMPLETE

Where:

name is the name of the NetView PPI subtask.

Explanation The loading of the CNMNETV module into virtual storage completed successfully. Processing continues.

Action No user action is required.

NSP312 *name*: UNABLE TO LOAD CNMNETV

Where:

name is the name of the NetView PPI subtask.

Explanation The loading of the CNMNETV module into virtual storage failed. The subtask terminates.

Action Verify that load module CNMNETV exists in a NetView STEPLIB data set. Restart the PPI subtask.

NSP313 *name*: NETVIEW INTERFACE FAILURE, RC = *rcnumber*

Where:

name is the name of the NetView PPI subtask.

rcnumber is the hexadecimal return code from CNMNETV call.

Explanation A call to the CNMNETV interface routine failed. The message is discarded.

Action See the *NetView Application Programming Guide* to determine the meaning of the given return code.

`NSP314 name: NETVIEW COMMAND RECEIVED`

Where:

name is the name of the NetView PPI subtask.

Explanation A message was successfully received from the PPI interface routine. Processing continues.

Action No user action is required. This message appears for debugging purposes, and appears only if you reassembled the subtask with the CMDREC lines uncommented.

`NSP595 COMMAND ENTERED:cmdtxt`

Where:

cmdtxt is the text of the command entered.

Explanation The mainframe application received a command from a console. Processing continues with the execution of the command.

Action No user action is required.

`NSP600 TNUM TASKNAME STATUS RESTARTS/LIMIT SPECIFIC`

Explanation This message is issued in response to the **SHOW TASK** command. An entry is displayed for each subtask along with its status, the number of times it was restarted, and task-specific information.

Action No user action is required.

NSP603 TNUM TASKNAME ADDRESS HEADER TCB

Explanation This message is issued in response to the **SHOW ADDR** command. An entry is displayed for each subtask along with its address, task header address, and TCB address.

Action No user action is required.

NSP605 TNUM TASKNAME INPUTQ OUTPUTQ INFLOW OUTFLOW MC

Explanation This message is issued in response to the **SHOW FLOW** command. An entry will be displayed for each subtask, along with the size of its I/O queues, the number of messages that passed through, and the number of message buffers currently allocated to the task.

Action No user action is required.

NSP607 CiscoWorks Blue SNA Host *version* Copyright *copyright* Cisco Systems

Where:

version is the version number of the installed mainframe application.

copyright lists the copyright years for the installed mainframe application.

Explanation This message is issued in response to the **SHOW VERSION** command to specify the version of the workstation program.

Action No user action is required.

NSP608 TASKNAME DATE TIME

Explanation This is the header line issued in response to the **SHOW VERSION** command. The date and time of the load-module compilation are listed below this line for each subtask.

Action No user action is required.

NSP690 SNA Host STOP Command accepted

Explanation Displays the output of the **STOP** command. The mainframe application terminates.

Action No user action is required.

NSP695 SNA Host *cmdtype* command processed.

Where:

cmdtype is the name of the command just processed.

Explanation The mainframe application completed the initial processing of a console command. Additional messages might be forthcoming, depending on whether any additional work is to be done by the subtasks.

Action No user action is required.

NSP698 Subtask *task* is already *status*

task is the subtask name.

status is the subtask status.

Explanation A subtask modification command was invalid because of the current state of the target subtask. Processing continues.

Action Use the **SHOW TASK** command to check the subtask's status.

NSP699 Invalid operator command entered

Explanation You entered an invalid command.

Action Check the command and try again. Use the **HELP** command to see a list of valid commands.

NSP701 Starting subtask #*idnum* for *info*

Where:

idnum is the numerical ID for the newly started subtask.

info is the information sent to the ATTACH macro.

Explanation The mainframe application attached a subtask with the information given in *info*. Processing continues with the ATTACH attempt.

Action No user action is required.

NSP702 Buffer size = *size*M, Queue depth = *totalmsg*, Maximum = *maxmsg*

Where:

*size*M is the size (in megabytes) allocated for messages.

totalmsg is the total message queue depth.

maxmsg is the maximum message queue depth permitted.

Explanation A message queuing problem occurred for an MCS console defined for the mainframe application. This message appears only in the mainframe application job log and is followed immediately by a more detailed message. Processing continues. The MCS console might be terminated, depending on the severity of the queueing problem.

Action Monitor the mainframe application job log for the next message.

NSP705 *TNUM TASKNAME FREE USED*

Where:

TNUM is the task number for the subtask.

TASKNAME is the task name for the subtask.

FREE is the number of free slots in the queue.

USED is the number of used slots in the queue.

Explanation This message displays the output of the **SHOW FREEQ** command. The mainframe application lists the number of free and used slots on each task's queue.

Action No user action is required.

NSP706 *TNUM TASKNAME SPECIFIC CLIENT*

Where:

TNUM is the task number for the subtask.

TASKNAME is the task name for the subtask.

SPECIFIC is information specific to the subtask.

CLIENT identifies the workstation connected to the subtask.

Explanation This message displays the output of the **SHOW CONN** command. The mainframe application displays the host name or TCP/IP address of the workstation connected to each TCP/IP subtask. If no workstation is connected, "NO CONNECTION" is displayed.

Action No user action is required.

NSP708 *pu_name xid local_mac local_sap remote_mac remote_sap rif*

Where:

pu_name is the PU name.

xid is the exchange identifier.

local_mac is the MAC address for the local node.

local_sap is the SAP for the local node.

remote_mac is the MAC address for the remote node.

remote_sap is the SAP for the local remote.

Explanation This message displays the output of the **SHOW DLC** command. The data recorded for the PU name during the exchange of XIDs is displayed.

Action No user action is required.

`NSP710 rif_data`

Where:

rif_data is the Routing Information Field data.

`NSP901 Stopping subtask # number: name`

Where:

number is the subtask number.

name is the subtask name.

Explanation This message is issued in response to a **STOP** command. One message is issued for each subtask. A termination command is sent to each named subtask.

Action No user action is required.

`NSP902 name subtask terminated, RC = rcnumber`

Where:

name is the name of the subtask.

rcnumber is the return code from the termination call.

Explanation The named subtask has been terminated. Any queues or memory allocated for the subtask are freed.

Action No user action is required.

`NSP903 name in/out queue freed, RC = rcnumber`

Where:

name is the name of the subtask.

in/out is input or output.

rcnumber is the return code from the Free call.

```
An allocated message queue for the named subtask is freed.
Processing continues.
```

Action No user action is required.

```
NSP904 All SNA Host subtasks completed
```

Explanation The mainframe application has completed the shutdown of all subtasks. Processing continues with the shutdown of the main task.

Action No user action is required.

NSP905 Restart # *attempt* of subtask *task* will be attempted in *time* seconds

Where:

attempt is the number of times the subtask has been restarted.

task is the subtask name.

time is the number of seconds until the next restart attempt is made.

Explanation The named subtask terminated and attempts to restart automatically in the stated number of seconds. The subtask sleeps for the specified number of seconds before attempting to restart itself.

Action No user action is required. The **INIT** command is used during the delay to bypass the delay and perform an immediate restart of the subtask. The **KILL** command is used to bring down the subtask and cancel any further automatic restart attempts.

NSP906 No auto restart for *task*. Use INIT command to restart.

task is the subtask name.

Explanation The named subtask terminated and does not attempt to restart itself because either the operator explicitly issued a **KILL** command against the subtask, or the maximum number of restart attempts was exceeded. The **INIT** command might be used to reinitialize the subtask and reset the number of restart attempts. The subtask is terminated.

Action No user action is required.

NSP907 LU6.2 return codes RCPRI = *x*, RCSEC = *y*

Explanation This message is issued for LU 6.2 error conditions, listing the primary and secondary return codes.

Action Record the return codes for problem determination.

NSP910I STATUS UPDATE PROCESSED FOR PU xxxxxxx, DLUR NAME yyyyyyyy

Explanation This message is issued by ISTEXCCS indicating that a change has occurred in the DLUR served PU status.

Action No user acction is required.

NSP2000I DATA RECORDED TO *dbtype* FOR PU *puname*

Where:

dbtype is the identifier for the database to which the PU data was recorded. Values for this variable are PRIMARY and BACKUP.

puname is the PU name for which data was recorded.

Explanation Data for the PU identified by *puname* was successfully recorded in the database identified by *dbtype*.

Action No action is required.

NSP2003E *dbtype* XID OUTPUT DATA SET UNAVAILABLE.

Where:

dbtype is the identifier for the database that did not open. Values for this variable are PRIMARY and BACKUP.

Explanation OPEN failed for the database identified by *dbtype*.

- If *dbtype* is PRIMARY, all subsequent vector processing will be suppressed.
- If *dbtype* is BACKUP, processing continues, but switching from primary to backup cannot occur.

  In the event that switching is necessary and no BACKUP is available, all vector processing is suppressed. In either case, loss of data is probable.

Action Failure to open the PRIMARY database must be corrected immediately for the data recording function to initialize. Failure to open the BACKUP is less severe and can be deferred, with the possible consequence of lost data in the event switching is required. Usually a failure to open is caused by a configuration problem rather than a component failure. Check the allocation for the identified database to ensure that it is allocated correctly. Databases are identified in the VTAM startup procedure. The PRIMARY database is identified by the XIDDATA DD statement; the BACKUP database is identified by the XIDBACK DD statement. Verify that the DSNAMES (DSN keyword) identify the correct data sets.

NSP2005E WRONG VECTOR PASSED.

Explanation An internal failure in the Exit Manager caused an invalid vector to be passed to the XID data recording function. Though no damage is reported by this message, it can be caused by storage overlays.

Action Use the following command sequence to reinitialize the exit routine:

F NET,EXIT,ID=ISTEXCCS,OPT=INACT

F NET,EXIT,ID=ISTEXCCS,OPT=ACT,MODULE=*exitname*

Where *exitname* is the name given to the NSP Exit Manager load module. If the problem recurs, notify the Cisco TAC.

`NSP2006E PUT NOT SUCCESSFUL R15 = r15value RC = rcvalue`

Where:

r15value is the decimal value of the return code resulting from the VSAM PUT.

rcvalue is the decimal value of the FDBK field associated with the above return code.

Explanation An attempt to write a data record to a database failed. If recording to the PRIMARY database is active, the Data Recording Function attempts to switch to the BACKUP database. This message can also mean that the VSAM databases are too small for tracking the network. In this case, reallocate the databases. A sample job stream for database allocation can be found in *prefix.NSPS301.NSPSSAMP* in member *NSPDBVSM*.

Action If the databases are too small, reallocate them and increase their size. Otherwise, verify the integrity of the VSAM databases. Databases are identified in the VTAM startup procedure. The PRIMARY database is identified by the XIDDATA DD statement; the BACKUP database is identified by the XIDBACK DD statement.

`NSP2007E NO DATASET AVAILABLE FOR OUTPUT.`

Explanation One of the following errors occurred:

- During initialization of the Data Recording Function, the PRIMARY database, identified by the XIDDATA DD statement in the VTAM startup procedure, could not be opened.
- During routine operation, an attempt was made to switch from the PRIMARY to the BACKUP database, and the switch operation failed.

Action Failure to open the PRIMARY database must be corrected immediately for the data recording function to initialize. Failure to open the BACKUP database is less severe and can be deferred, with the possible consequence of lost data in the event switching is required.

If you have set the database sizes large enough, it should not be necessary for Maps and SNA View to switch databases. Database switching usually occurs because the database is too small for the network being tracked. Increase the amount of space allocated to the PRIMARY and BACKUP databases. A sample job stream to allocate databases is included in *prefix.NSPS301.NSPSSAMP* in member *NSPDBVSM*. Change the allocation parameters to those appropriate for your network.

`NSP2008E DID NOT PUT DB KEY R15 = r15value RC = rcvalue`

Where:

r15value is the decimal value of the return code resulting from the VSAM PUT.

rcvalue is the decimal value of the FDBK field associated with the above return code.

Explanation An attempt to write a VSAM key to a database failed.

Action Verify the integrity of the VSAM databases. Databases are identified in the VTAM startup procedure. The PRIMARY database is identified by the XIDDATA DD statement, and the BACKUP database is identified by the XIDBACK DD statement.

`NSP2009E MODCB OF RECLEN FAILED FOR NEW RECORD`

Explanation An attempt to set the record length (RECLEN) value for a new database record failed.

Action Notify the Cisco TAC.

`NSP2010E MODCB OF RECLEN FAILED FOR UPDATE`

Explanation An attempt to set the RECLEN value for update of an existing database record failed.

Action Notify the Cisco TAC.

NSP2011E MODCB OF PUT RPL TO BACKUP FAILED

Explanation The attempted switch from PRIMARY to BACKUP databases failed. All subsequent data recording will be suppressed.

Action Notify the Cisco TAC.

NSP2012E MODCB OF UPDATE RPL TO BACKUP FAILED

Explanation The attempted switch from PRIMARY to BACKUP databases failed. All subsequent data recording will be suppressed.

Action Notify the Cisco TAC.

NSP2013E MODCB OF RECLEN FAILED FOR NEW DB KEY

Explanation An attempt to set the record length for a new database record failed.

Action Notify the Cisco TAC.

NSP2014E MODCB OF RECLEN FAILED FOR UPDATE OF DB KEY

Explanation An attempt to set the record length for an existing database record failed.

Action Notify the Cisco TAC

Installing CiscoWorks Blue in a Different Zone

This appendix provides information on installing the CiscoWorks Blue mainframe components in a mainframe zone other than the default MVS zone. For an example of creating such a zone, see the following members in the *NSPSINST* data set:

- NSPSMPA—Job to allocate a new CSI
- NSPCSI—Deletes, defines, and primes the VSAM data set
- NSPSMPS—Proc to be used with the receive and apply JCL (needs to be placed in a system proclib)

This appendix includes the following sections:

Installing the Mainframe Application in Another Zone

This section describes how to install the mainframe application in a zone other than the MVS zone. To install the mainframe application in another zone, use the following procedure:

Step 1 Unload the SMP/E installation samples from the CD-ROM using the "Installing the Mainframe Application Using SMP/E" section on page 3-4.

Step 2 Use the sample SMP/E JCL named ALLOC, in *prefix.NSPS301.NSPSINST,* to allocate your target and distribution libraries.

Step 3 Submit (run) the SMP/E JCL named MOVNSP that is located in the *prefix.NSPS301.NSPSINST* data set. This JCL copies the contents of the CD-ROM to DASD.

Step 4 Edit the *prefix.NSPS301.SMPMCS* member, changing the value of ++VER(Z038) to the preferred zone.

Step 5 Submit (run) the SMP/E JCL named RECNSPD, in the *prefix.NSPS301.NSPSINST* data set, to receive the mainframe application.

Step 6 Use the sample SMP/E JCL named APPNSP, in *prefix.NSPS301.NSPSINST,* to apply the mainframe application.

Step 7 Use the sample SMP/E JCL named ACCNSP, in *prefix.NSPS301.NSPSINST,* to accept the mainframe application.

You have completed loading the mainframe application files from the CD_ROM. The next step is to configure the mainframe application software. Go to Chapter 4, "Updating the Mainframe Application Software."

Defining the Option and Utility for Assembler

You might be required to define the option and utility for the assembler. You can use the following procedure (these steps might differ slightly at your site):

Step 1 Access the SMP/E Primary Menu.

Step 2 Select **ADMINISTRATION** and select the data set name.

Step 3 Select **DEFINITION**.

Step 4 Select the **GLOBAL** zone and confirm the selection.

Step 5 From the Entry Type Selection menu, select **OPTIONS**.

Step 6 Select **NSPOPT**.

Step 7 Select the option to define a utility.

Step 8 For assemblies, define ASMA90.

Step 9 Return to the Entry Type Panel, and select **UTILITY**. The utility name is ASMA90.

Step 10 Enter your installation information. For example:

- Load module—ASMA90
- Highest return code—4
- Print DD name—SYSPRINT
- PARM—LIST,DECK,NOOBJECT,USING(WARN(2))

Mainframe and Workstation Installation Checklist

This appendix lists the parameters that must be coordinated between the mainframe and workstation configurations. We recommend that mainframe or workstation installers agree on these values before installation is started.

This Appendix includes information about the following two types of parameters:

TCP/IP Connection Parameters

The following table lists the TCP/IP parameters that configure TCP/IP connection between the mainframe and the workstation. On the workstation, these parameters are contained in the */etc/svopen_config_DOMAINNAME* file you configure using a text editor. The related mainframe parameters are set using the TCP parameter card in the NSPPARM member. The following worksheet lists the TCP/IP configuration parameters.

TCP/IP Configuration Parameters Worksheet

TCP/IP host name or address of the mainframe:	
Mainframe value:	_____
Workstation value: SVMF_AGENT_ADDR=	_____
Default:	None
Connection protocol:	
Mainframe value:	TCP parameter card
Workstation value: SVMF_CONNECT=	TCP
Default:	None
Host connection interface port:	
Mainframe value:	_____
Workstation value: SVMF_HCI_AGENT_PORT=	TCP *HCS_port* _____
Default:	6104
Host command server port:	
Mainframe value:	
Workstation value: SVMF_CMDS_AGENT_PORT=	TCP _____ *HCS_port*
Default:	6105

The TCP/IP address of mainframe is the host name or IP address of the mainframe on which the application is installed. Use the SVMF_AGENT_ADDR parameter in the workstation */etc/vopen_config_DOMAINNAME* file.

The connection protocol is the communication protocol between the mainframe and the workstation. The SVMF_CONNECT parameter indicates whether the workstation will connect to the mainframe using TCP/IP. To connect via TCP/IP, specify **SVMF_CONNECT = TCP** in the workstation */etc/svopen_config_DOMAINNAME* file and provide a TCP parameter card at the mainframe.

The host connection interface port is the port number used by the host connection interface. Use this port number in the SVMF_HCI_AGENT_PORT parameter in the workstation */etc/svopen_config_DOMAINNAME* file and on the mainframe TCP parameter card.

The host command server port is the port number used by the host command server. Use this port number in the SVMF_CMD_AGENT_PORT parameter in the workstation */etc/svopen_config_DOMAINNAME* file and on the mainframe TCP parameter card.

Coding the TCP Parameter Card

Use the TCP parameter card at the mainframe to identify the ports to be used for the host command server and the host connection interface. The TCP parameter card has the following format:

`TCP hciport cmdport`

Where:

`hciport` is the port for the host connection interface (usually 6104).

`cmdport` is the port for the host command server (usually 6105).

LU 6.2 Parameters

This section describes an additional set of parameters for connecting the workstation to the mainframe using LU 6.2. The following worksheet lists the LU 6.2 parameters.

Note Before you configure LU 6.2, you must have already configured the mainframe and workstation to support an LU 6.2 connection between them. The mainframe application configuration adds only the information necessary for the workstation application to communicate with the mainframe application. Items like the Remote Link address have been omitted, since this should be already configured.

LU 6.2 Configuration Parameters Worksheet

Connection protocol:	
Mainframe value:	LU62 parameter card
Workstation value: SVMF_CONNECT=	LU62
Default:	None
Secondary independent LU:	
Mainframe value:	_____
Workstation value:	_____
Default:	None
Log mode:	
Mainframe value:	PARALLEL
Workstation value:	PARALLEL
Default:	PARALLEL
Message transaction program:	
Mainframe value:	NSPOPNMS
Workstation value:	NSPOPNMS
Default:	NSPOPNMS
Command transaction program:	
Mainframe value:	NSPOPNCS
Workstation value:	NSPOPNCS
Default:	NSPOPNCS

The connection protocol is the communication protocol between the mainframe and the workstation. The SVMF_CONNECT parameter indicates whether the workstation will connect to the mainframe using LU 6.2. To connect via TCP/IP, specify **SVMF_CONNECT=LU62** in the */etc/svopen_config_DOMAINNAME* file and provide a TCP parameter card at the mainframe.

The logmode specifies the VTAM logmode entry. Use PARALLEL on the SERVER parameter card.

The message transaction program identifies the workstation message server transaction program (MESSAGE_TP). Code NSPOPNMS on the SERVER parameter card.

The command transaction program identifies the workstation command server transaction program (COMMAND_TP). Code NSPOPNCS on the SERVER parameter card.

Coding the SERVER Parameter Card

You specify the mainframe parameters on the SERVER card in the NSPPARM member in the mainframe samples data set:

SERVER *plu slu* PARALLEL NSPOPNMS NSPOPNCS

Where:

plu is the label of the VTAM APPL definition that you coded with APPC=YES, that is the primary LU for the mainframe.

slu is the label of a CDRSC for the independent secondary LU defined for the workstation and associated with the workstation application PU.

PARALLEL is the logmode.

NSPOPNMS is the name of the SNA LU 6.2 transaction program for the workstation message server (MESSAGE_TP). You must use NSPOPNMS.

NSPOPNCS is the name of the SNA LU 6.2 transaction program for the workstation command server (COMMAND_TP). You must use NSPOPNCS.

Defining the Workstation SNA Package

On the workstation, you configure LU 6.2 workstation parameters using the SNA package on the workstation (Communications Server for AIX or SNAplus2Link). Use the following values:

- The independent secondary LU defined for the workstation must match the *slu* parameter on the mainframe SERVER card.
- The workstation message server (MESSAGE_TP) is NSPOPNMS.
- The workstation command server (COMMAND_TP) is NSPOPNCS.
- The log mode is PARALLEL.

Symbols

A

B

C

D

E

F

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

General Information

1 Years of networking experience _____ Years of experience with Cisco products _____

2 I have these network types: ☐ LAN ☐ Backbone ☐ WAN
☐ Other: _____

3 I have these Cisco products: ☐ Switches ☐ Routers
☐ Other: Specify model(s) _____

4 I perform these types of tasks: ☐ H/W Install and/or Maintenance ☐ S/W Config
☐ Network Management ☐ Other: _____

5 I use these types of documentation: ☐ H/W Install ☐ H/W Config ☐ S/W Config
☐ Command Reference ☐ Quick Reference ☐ Release Notes ☐ Online Help
☐ Other: _____

6 I access this information through: ____% Cisco Connection Online (CCO) ____% CD-ROM
____% Printed docs ____% Other: _____

7 Which method do you prefer? _____

8 I use the following three product features the most:

Document Information

Document Title: CiscoWorks Blue Maps and SNA View Mainframe Installation and Administration Guide

Part Number: 278-11349-01 S/W Release (if applicable): Release 3.0.1

On a scale of 1–5 (5 being the best) please let us know how we rate in the following areas:

_____ The document was written at my technical level of understanding.
_____ The document was complete.
_____ The information was well organized.
_____ The information was accurate.
_____ The information I wanted was easy to find.
_____ The information I found was useful to my job.

Please comment on our lowest score(s):

Mailing Information

Company Name Date

Contact Name Job Title

Mailing Address

City	State/Province	ZIP/Postal Code
Country	Phone ( )	Extension
Fax ( )	E-mail	

Can we contact you further concerning our documentation? ☐ Yes ☐ No

You can also send us your comments by e-mail to **bug-doc@cisco.com**, or fax your comments to us at **(408) 527-8089**.

Feedback on the CiscoWorks Blue Maps and SNA View Mainframe Installation and Administration Guide, OL-6388-01, Release 3.0