Network Utility

Installation, Getting Started, and User's Guide

IBM

Network Utility

Installation, Getting Started, and User's Guide

> **Note**
>
> Before using this information and the product it supports, be sure to read the general information under Appendix A, "Notices" on page A-1 and safety information in Appendix B, "Safety Information" on page B-1.

First Edition (August 1998)

This edition applies to the Network Utility Models TN1 and TX1.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

> Department CGF
> Design & Information Development
> IBM Corporation
> PO Box 12195
> RESEARCH TRIANGLE PARK NC 27709-9990
> USA

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Getting Started

Learning About Network Utility

Configuration and Management Specifics

Appendix

About This Guide

This guide explains how to set up the IBM Network Utility, perform initial configuration, correct problems that might occur during installation, and use the Network Utility. It also contains detailed configuration examples for some common Network Utility network configurations.
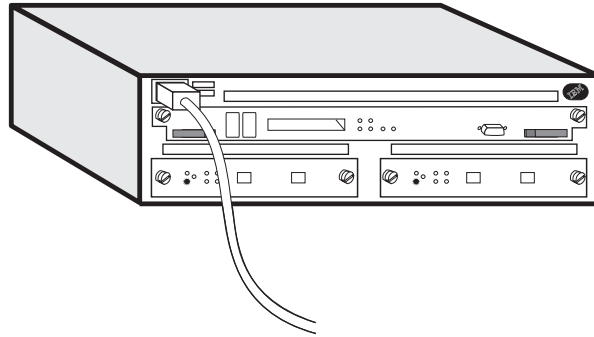
There are two models of the IBM Network Utility: the Network Utility TN3270e Server (Model TN1) and the Network Utility Transport (Model TX1). Unless explicitly stated, the term *Network Utility* applies to both the Model TN1 and the Model TX1.

This guide is an addition to the existing documentation for the Network Utility that is described in "Library Overview" on page viii. This guide helps you get started with the more detailed reference information that is documented in the other books.

Who Should Use This Guide

This guide is intended to be used by the person responsible for installing, configuring, and managing the Network Utility.

How to Proceed

Installation and Initial Configuration

1. Install the chassis and the cables (see Chapter 1).

   Note: Installation of the cables for the Parallel Channel Adapter (FC 2299) requires IBM service or channel-trained personnel.

2. Connect a terminal or workstation to be able to configure and operate the product (see Chapter 2).

3. Decide which configuration method that you want to use and perform an initial configuration of the Network Utility (see Chapter 3).

Learning

1. If you already have some experience with the command-line interface of IBM routing products or if you prefer to try tasks without following a tutorial, use Chapter 4 to review some of the basics of

navigating the command-line interface. Scan the other chapters in Learning About Network Utility so that you know where to find additional information that you may need.

2. If the command-line interface of IBM routing products is new to you, use Chapter 5 as a tutorial to help you learn about basic concepts and navigation. Scan the other chapters in Learning About Network Utility so you know where to find additional information that you may need.

3. If you are familiar with basic configuration and operation functions, select from the configuration scenarios that we provide in Configuration and Management Specifics. Select a configuration that resembles your network characteristics:

   - Model TN1 Users - see Chapter 12, "TN3270E Server"

   - Model TX1 Users - see Chapter 14, "Channel Gateway" or Chapter 16, "Data Link Switching"

   - All users - see Chapter 18, "Sample Host Definitions" if your configuration involves IBM host networking products

Final Configuration and Operation

1. Use the operations and management tasks that are introduced in Learning About Network Utility and the scenarios that are documented in Configuration and Management Specifics to debug and complete your initial configuration.

Library Overview

The Network Utility and the IBM 2216 Model 400 share many of the same publications. The following figure shows the publications in the library, arranged according to tasks.

Planning and Installation

Introduction and Planning Guide GA27-4105	**Installation and Initial Configuration Guide** GA27-4167 GA27-4106	**Hardware Configuration Quick Reference Card** GX27-3988

Configuration

Configuration Program User's Guide **GC30-3830**	**Configuration Program READ.ME**	Configuration Help
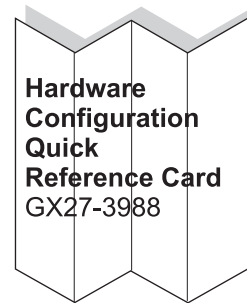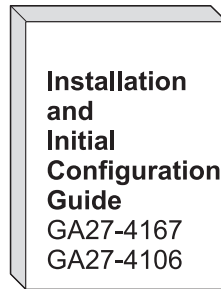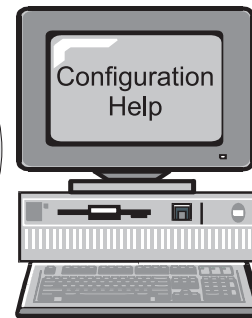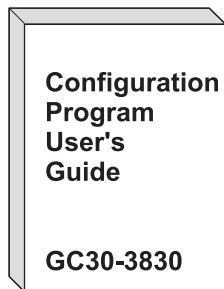
Diagnostics/ Maintenance

Service and Maintenance Manual SY27-0350
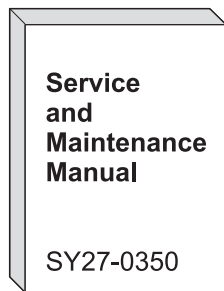
Operations and Network Administration

Software User's Guide SC30-3886	**Protocol Configuration and Monitoring Reference** SC30-3884 SC30-3885	**Event Logging System Messages Guide** SC30-3682
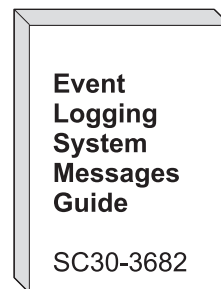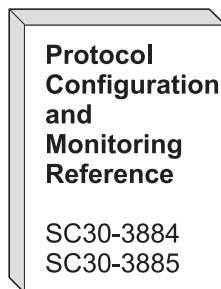
Common Tasks and the Library for the IBM 2216 Model 400 and Network Utility

Table 0-1. Hardcopy Publications that Are Shipped with the product. These documents are shipped in hardcopy and are also contained on this product's Documentation CD-ROM, SK2T-0405.

Planning

GA27-4105   *2216 Nways Multiaccess Connector and Network Utility Introduction and Planning Guide*

This book explains how to prepare for installation and select the hardware that you want to purchase. It includes specifications for the hardware and software for your network. It also provides information on the management of routing networks.

Installation

GA27-4167   Network Utility only:

Network Utility Model TX1 or TN1 Installation and Initial Configuration Guide

This booklet explains how to install a Network Utility and verify its installation.

GA27-4106   2216 Model 400 only:

2216 Nways Multiaccess Connector Model 400 Installation and Initial Configuration Guide

This booklet explains how to install the 2216 Model 400 and verify its installation.

GX27-3988   2216 Model 400 only:

2216 Nways Multiaccess Connector Hardware Configuration Quick Reference

This reference card is used for entering and saving hardware configuration information used to determine the correct state of an IBM 2216 Model 400.

Diagnostics and Maintenance

SY27-0350   *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual*

This book provides instructions for diagnosing problems with and repairing the Model 400 or the Network Utility.

Safety

SD21-0030   *Caution: Safety Information—Read This First*

This book provides translations of caution and danger notices applicable to the installation and maintenance of a device.

Configuration

GC30-3830   *Configuration Program User's Guide*

This book discusses how to use the Nways Multiprotocol Access Services Configuration Program.

> *Table 0-2. Publications that Are Shipped as Softcopy on the CD-ROM. These publications are also separately orderable as hardcopy.*
>
> **Operations and Network Management**
>
> The following books support the Nways Multiprotocol Access Services program.
>
> SC30-3886   *Nways Multiprotocol Access Services Software User's Guide*
>
>   This book explains how to:
>
> - Configure, monitor, and use the Nways Multiprotocol Access Services software and microcode.
> - Use the Nways Multiprotocol Access Services command-line router user interface to configure and monitor the network interfaces and link-layer protocols shipped with the 2216 base.
>
> SC30-3884   *Nways Multiprotocol Access Services Protocol Configuration and Monitoring Reference, Volume 1*
>
> SC30-3885   *Nways Multiprotocol Access Services Protocol Configuration and Monitoring Reference, Volume 2*
>
>   These books describe how to access and use the Nways Multiprotocol Access Services command-line user interface to configure and monitor the routing protocol software shipped with the product.
>
>   They include information about each of the protocols that the device supports.
>
> SC30-3682   *Nways Event Logging System Messages Guide*
>
>   This book contains a listing of the error codes that can occur, along with descriptions and recommended actions to correct the errors.

Visit our Web Sites

These IBM web pages provide product information:

For the Network Utility - `http://www.networking.ibm.com/networkutility`
For the Model 400 - `http://www.networking.ibm.com/216/216prod.html`

This IBM web page provides 2216 base books online:

`http://www.networking.ibm.com/did/2216bks.html`

Information, Updates, and Corrections

This page provides information on engineering changes, clarifications, and fixes that were implemented after the books were printed:

`http://www.networking.ibm.com/216/216changes.html`

Getting Started

Chapter 1. Setting Up the Hardware

This chapter covers the following:

- Defining what you need to install and configure the Network Utility
- Rack-mounting or surface-mounting the Network Utility chassis
- Inserting PCMCIA cards
- Powering on the Network Utility for the first time
- Verifying that the LEDs show a healthy system

Installing the Network Utility

Before You Begin: The illustrations assume that all of the adapter slots are filled. A fully populated Network Utility weighs about 15 kg (33 lb).

Preinstallation Requirements - You need to provide the following:

- An ASCII terminal or a workstation (PC)
- For the workstation, either Telnet client or ASCII terminal emulation software (for example, ProComm)
- If you are dialing into the Network Utility PCMCIA modem, a modem for your remote workstation
- If you will transfer configuration files or code into Network Utility (other than via XMODEM), a LAN adapter for your workstation
- If you will use the Network Utility PCMCIA EtherJet card, a small Ethernet hub or a cross-over cable to directly attach an Ethernet-capable workstation

1. Unpack and Verify

Unpack the Network Utility and verify that, along with this guide, the following items were included. (This is not a packing list. These are items that you need during installation.):

Documentation

- *Caution: Safety Information–Read This First*, SD21-0030
- *2216 Nways Multiaccess Connector and Network Utility Introduction and Planning Guide*, GA27-4105
- *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual*, SY27-0350
- *Configuration Program User's Guide*, GC30-3830
- *2216 Documentation CDROM*, SK2T-0405

Hardware

- Network Utility with the adapters already installed
- Any cables that were ordered
- Rack-mount installation aid
- Power cord
- PCMCIA modem (except in countries where the PCMCIA modem is not available)
- IBM EtherJet PC card
- Rack-mounting cable bracket if the Network Utility contains FC 2299 (Parallel Channel Adapter)
- Null modem and two 9-to-25 pin serial communications cables

Software

- IBM 2216 Model 400 and Network Utility Configuration Program CDROM
- The operating code is pre-loaded onto the Network Utility

Proceed with:

2. Rack-Mounting the Network Utility
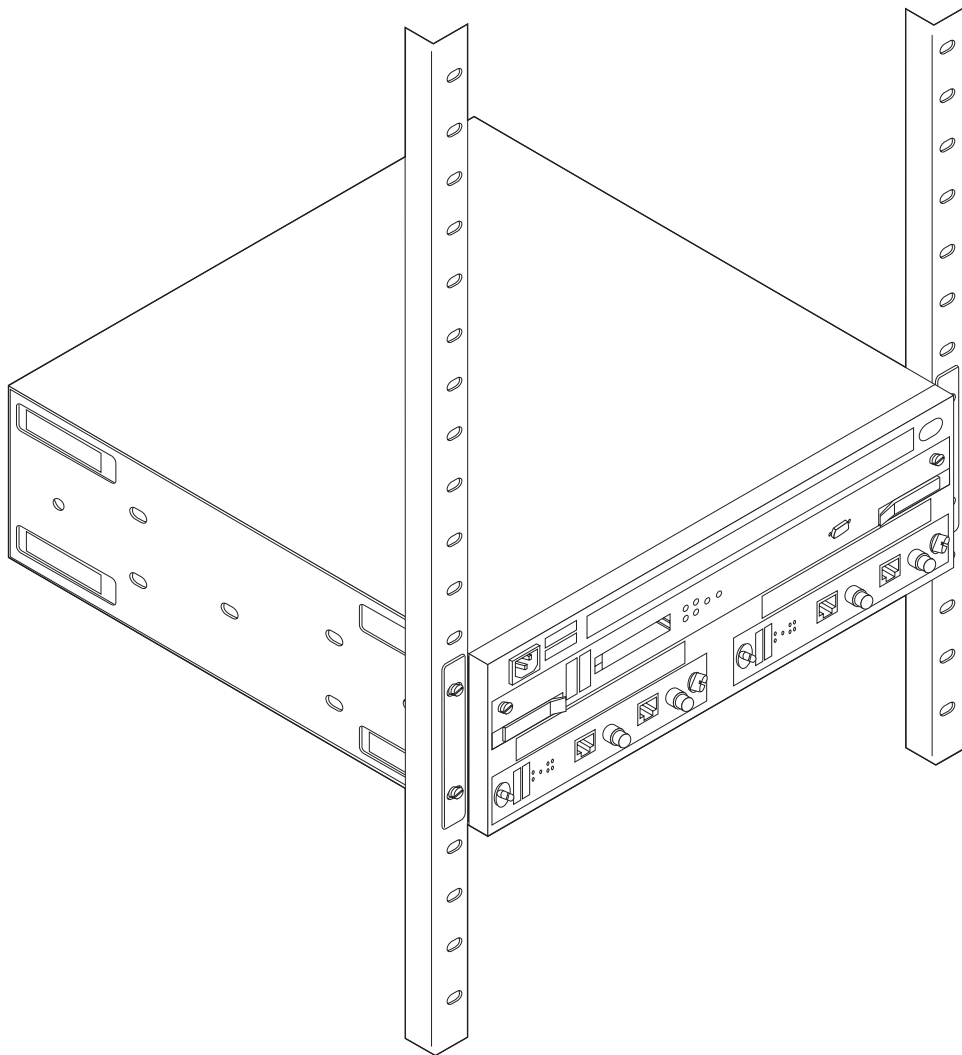
You need the following items:

- Cables, as required
- Four rack-mounting screws
- Screwdriver

Notes:

1. If you have a shelf for the rack, install it before continuing.

2. Do not use the installation aid if you have a shelf installed.

Continue with step 3 on page 1-3.

3. Rack-Mounting (Optional for Surface-Mounting)
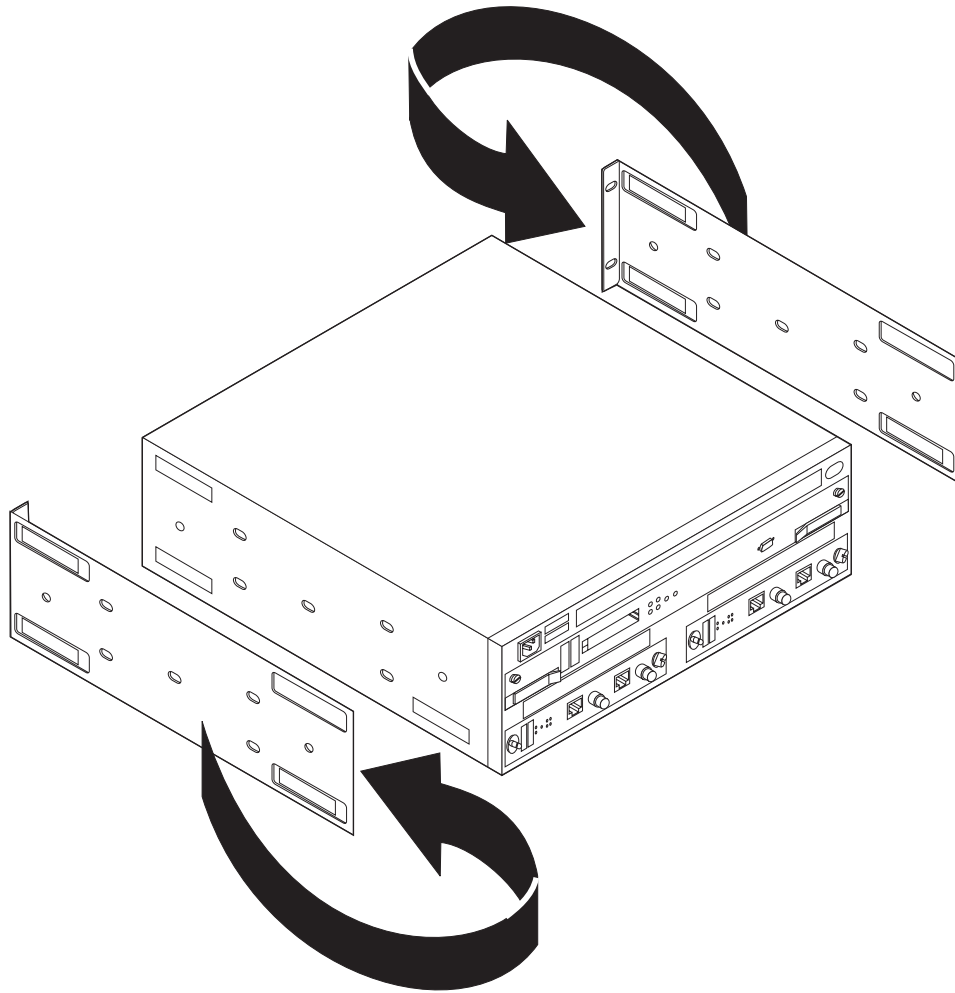
The Network Utility mounting brackets are shipped with the flanges facing the rear:

1. Remove the two screws from each bracket (one at the front and one at the rear).

2. Reverse each bracket so that the Network Utility can be rack-mounted.

3. Reinstall the four screws.

When the brackets are fitted correctly, the letter embossed on each bracket is on the rear edge; an A on the right side and a B on the left side.

4. Rack-Mounting
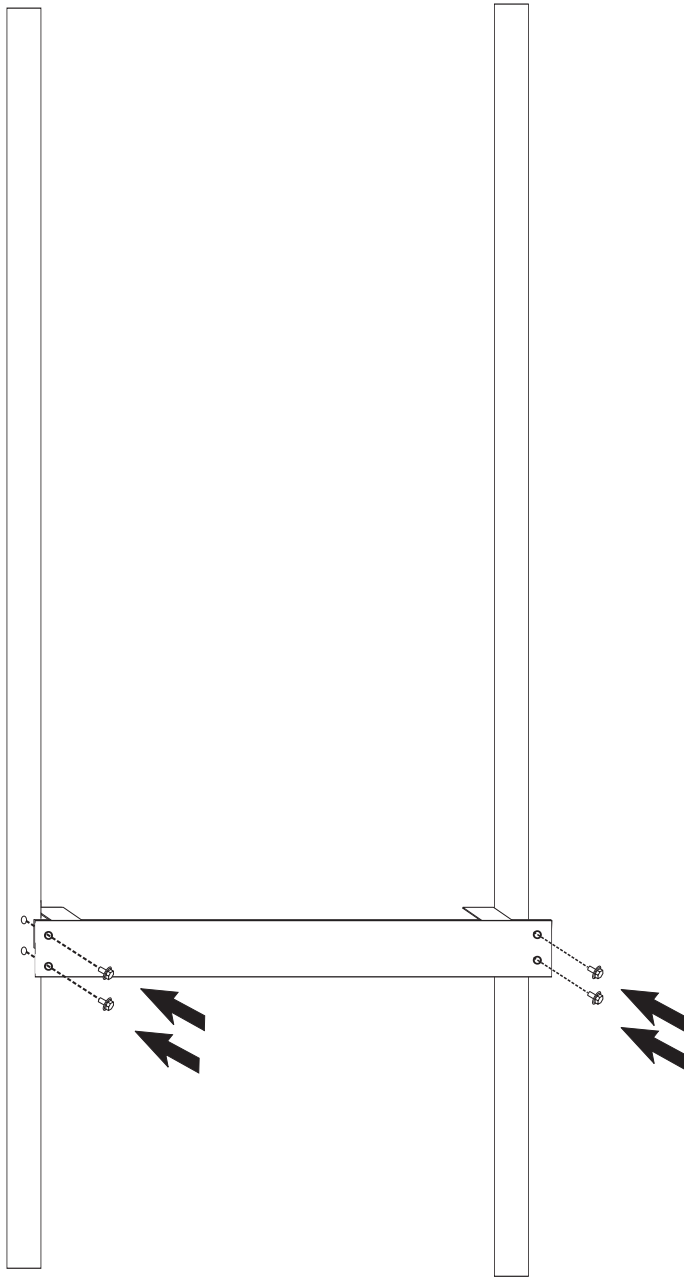
The installation aid is a metal bar that supports the Network Utility as you install it in the rack. The installation aid ensures that the Network Utility and rack are lined up correctly.

Line up the holes in the installation aid with the rack and install all screws.

5. Rack-Mounting

Set the Network Utility on the IBM 2216 installation aid or on the shelf. The mounting brackets keep the Network Utility from falling into the rack during installation.

With the installation aid installed, steady the Network Utility while you complete the next step.

6. Rack-Mounting
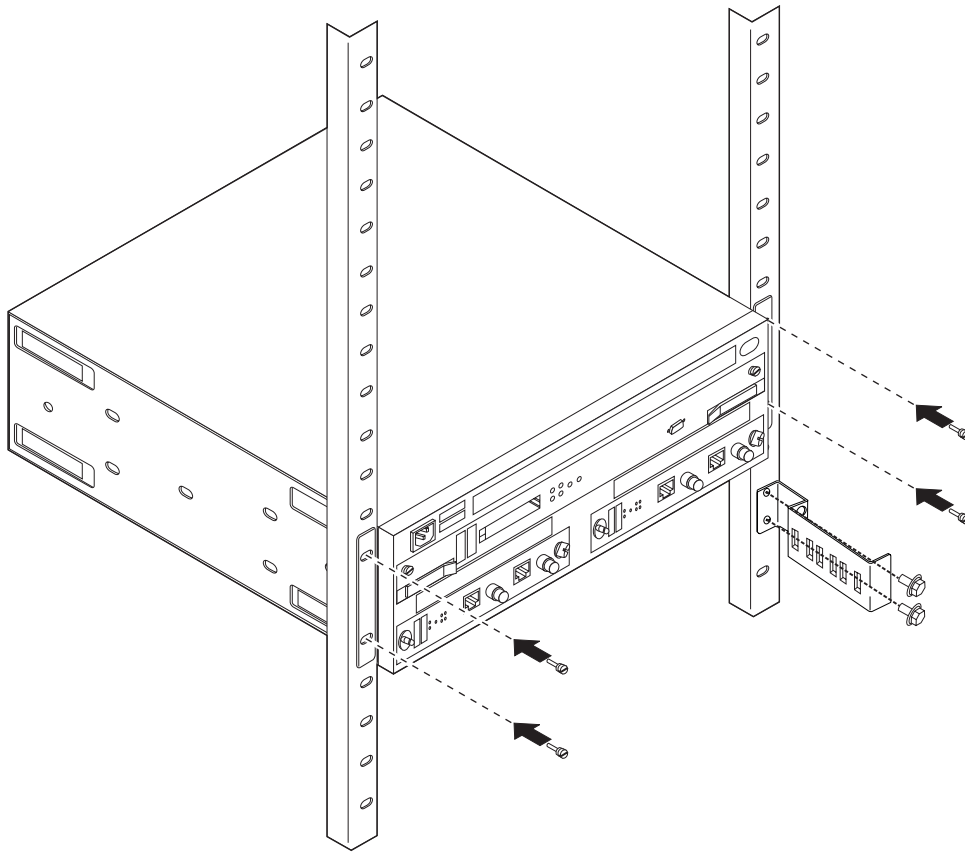
1. Install the screws beginning with the lower screws.

2. For FC 2299: Using 2 screws, install the rack-mounting cable bracket onto the front of the rack below the Network Utility.

7. Rack- or Surface-Mounting
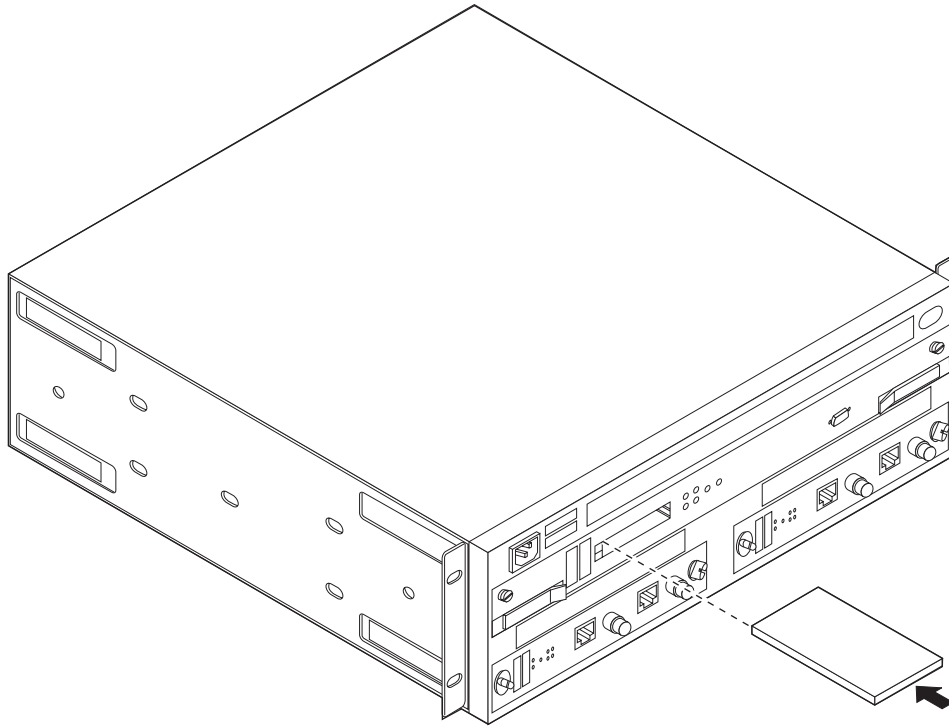
If you are installing a PCMCIA modem or PCMCIA EtherJet LAN adapter, slide it into either of the PCMCIA slots on the system card. Attach the telephone cable to the modem (a triangle identifies the left side of the cable).

Notes:

1. You cannot substitute a different Ethernet PCMCIA card, for the EtherJet LAN adapter that is shipped with the Network Utility.

2. The system will not boot up if you try to install two PCMCIA modems or two PCMCIA Ethernet adapters into a Network Utility.

8. Rack- or Surface-Mounting
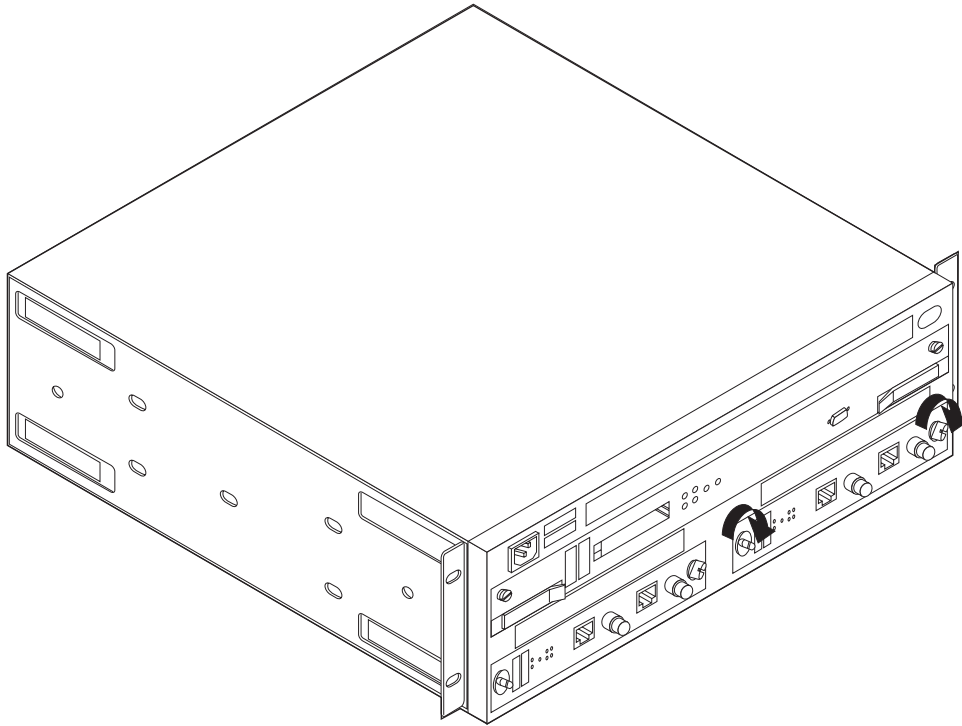
1. Verify that **all** thumbscrews are tight (even if you did not loosen them during installation).

2. Connect the power cord to the Network Utility and the power outlet (to power on the unit). After about 4-5 minutes, verify that the correct LEDs are on (see Table 1-1 on page 1-8). Monitor the LED states as shown in Figure 1-1 on page 1-9.

   While the unit is booting and the adapters are being tested, it is normal for:

   • Both the green and yellow System Card LEDs to be on for a short period of time.

   • Both the green and yellow Adapter Card LEDs to be on for a short period of time.

   • The Hard Drive and the adapter Wrong Slot yellow LEDs to be on for a short period of time.

   If you see a problem, use the tables and procedures in "Problem Solving" on page 1-11 to resolve or report the problem.

9. Complete the Setup (Rack- or Surface-Mounting)

1. Connect the cables (except for the Parallel Channel Adapter, FC 2299).

   Note: If you have FC 2299, the installation of the cables requires a channel-trained IBM service representative or a customer's channel-trained person.

   Call the IBM service representative to install the cables for FC 2299. The Parallel Channel and its attached devices will be disrupted if the cables are not installed correctly.

2. Proceed with Chapter 2, "Bringing Up a User Console" on page 2-1 to setup a user terminal console.

10. IBM Service Representative Tasks for FC 2299

1. Connect the adapter cables to FC 2299 (using the procedures in the *Service and Maintenance Manual* under "Installing Channel Adapters."). Do not connect to the host channel cables yet.

2. Run wrap tests to verify that all adapter cables are OK.

3. Connect the host channel cables to the adapter cables.

Verifying the Hardware Setup

Table 1-1 shows the desired state of the LEDs on the front of the unit after it has completed booting (**about 4-5 minutes after a power-on**). If all LEDs are in the correct state, you can begin to configure the unit. See Figure 1-1 on page 1-9) for the locations of the LEDs on the Network Utility.

Table 1-1. Machine LEDs When Operational

LED	Status
System card PCMCIA (with device installed) Port 1 yellow	OFF
System card PCMCIA (with device installed) Port 2 yellow	OFF
System card green	ON
System card yellow	OFF
For all adapter cards, OK green LEDs	ON
For all adapter cards, not OK yellow LEDs	OFF
For all adapters cards, wrong slot yellow LED	OFF
For all adapter cards, I/O port green LEDs (before the configuration is loaded on the unit)	OFF
For all adapter cards, I/O port yellow LEDs	OFF

LED Indicators

The Network Utility has a number of light-emitting diodes (LEDs) that indicate how the unit is functioning.

System Card
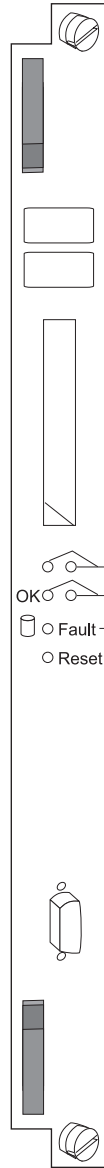
PCMCIA LEDs (Yellow)
System Card LEDs (OK Green, not OK Yellow)
Hard Drive LED

OK
Fault
Reset
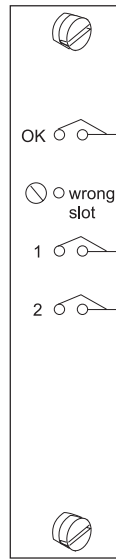
Adapter Card

OK — Adapter Card LEDs (Green and Yellow)

wrong slot — Wrong Slot (Yellow)

1 — Port 1 LEDs (Green and Yellow)

2 — Port 2 LEDs

Figure 1-1. System Card and Adapter Card LEDs

System Card Status

LEDs	Meaning
PCMCIA 1 or PCMCIA 2 (Yellow)	On - PCMCIA device has a fault, is not installed, or is not seated correctly.
	Off - Device passed self-tests
OK (Green)	On - Card hardware is operating normally.
	Blinking - Loading from hard file
(Yellow)	On - Card hardware has a fault.
Fault Hard Drive (Yellow)	On - Hard drive has failed.

Adapter Card Status

LEDs	Meaning
OK (Green)	On - Adapter is operational.
(Yellow)	On - Adapter has a fault.
Wrong Slot (Yellow)	On - Contact your service representative.
Green port [1]	On - Port is operating normally (enabled and configured).
	Off - Port is not configured or is disabled.
	Blinking (for ESCON adapter only) - The optical power measurement test is running.
Yellow port [1]	On - One or more ports has a hardware fault.
	Blinking - One or more ports has a port I/O or network failure. Use the Maintenance Analysis Procedures (MAPs) in the *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual* to isolate.
	Off - No problem detected.

Important Phone Numbers

Contact Name	Telephone Number
System Administrator:	
Service Representative:	

---

[1] The port LEDs of the multiport WAN adapters (FC 2282, FC 2290, and FC 2291) reflect the status of one or more of the ports.

Problem Solving

To identify and correct any problems that occur during setup, answer the questions and take the appropriate actions, as indicated:

On the system card, is the NOT OK yellow LED on?

> **Yes:** There is a fault in the card.
> 1. Disconnect the system from its power source.
> 2. Reseat the card.
> 3. Reconnect the system to its power source.
> 4. Wait 4-5 minutes, and verify the state of the LEDs.
> *If the problem is not corrected, contact your service representative.*
> **No:** Go to the next question.

On the system card, is the OK green LED off?

> **Yes:** The green LED is switched on by the operational code.
> *If the green LED fails to come on, contact your service representative.*
> **No:** Go to the next question.

On the system card, is the PCMCIA port LED on?

> **Yes:** Either the PCMCIA card slot is empty or the card failed the power-on self-test. Reseat the card.
> *If the problem is not corrected, contact your service representative.*
> **No:** Go to the next question.

On I/O cards in slots 1 and 2, are the NOT OK yellow LEDs On?

> **Yes:** There is a fault in the card. Reseat the adapter.
> *If the problem is not corrected, contact your service representative.*
> **No:** Go to next question.

On I/O cards in slots 1 and 2, are the OK green LEDs On?

> **Yes:** The Network Utility appears to be OK.
> **No:** Reseat the card. If the Green LED still fails to come on, the card is bad. Contact your service representative.

Chapter 2. Bringing Up a User Console

You must set up a terminal to access the Network Utility for configuration and operation. The information in this chapter helps you:

- Learn about the ways you can set up a terminal
- Choose the best method for your environment
- Attach and activate the terminal using default settings

When you are finished with this chapter, you should have an active terminal and it should be at the initial command prompt ready configuration.

Access Methods

You can access and connect to the Network Utility in several ways that are summarized in Table 2-1.

Table 2-1. User Console Attachment Options			
Physical Attachment	**Line Protocol**	**Access Protocol**	**Default IP Addresses**
Service port + null modem Service port + external modem PCMCIA modem	Asynchronous characters	ASCII Terminal emulation	Not Applicable
	SLIP	Telnet	Network Utility = 10.1.1.2 Workstation = 10.1.1.3
PCMCIA EtherJet	IP	Telnet	Network Utility = 10.1.0.2 Workstation = 10.1.0.3
Any IP network interface	IP	Telnet	No defaults

Make the physical connections in one of the following ways when you want to use:

1. **An ASCII terminal** or **a workstation** that is running terminal emulation software:

   - Local connection through a null-modem cable attached to the EIA 232 service port (see Figure 2-1 on page 2-2). This type of connection uses the null-modem adapter and the two 9-to-25 pin serial cables that are supplied with this product.

   - Remote dial-in (using telephone lines) through the PCMCIA modem (see Figure 2-2 on page 2-2)

   - Remote dial-in (using telephone lines) with an external modem (not pictured) attached to the EIA 232 service port. This configuration would be used in countries where there is no approved PCMCIA modem. Use an asynchronous modem that supports the Hayes AT command set. (Refer to the product support literature sales pages starting at www.networking.ibm.com/netutility to determine which modems are supported.)

2. **Telnet protocol** on a workstation that is running TCP/IP software:

   - Any of physical connections that are described in the methods in step 1.

For the Telnet method the telnet workstation is running TCP/IP software that supports SLIP (Serial Line Internet Protocol). SLIP is a method for sending IP packets across asynchronous lines.

- Local cable from the Network Utility PCMCIA LAN adapter (the IBM EtherJet PC card) to a workstation or local Ethernet hub. Figure 2-3 on page 2-3 shows a version of this configuration.

  The workstation Ethernet adapter could also be directly attached to the EtherJet card via a cross-over cable, or there could be a wide area network between the Ethernet LAN and the Telnet workstation.

  The Network Utility IBM EtherJet PC card is for service and operations purposes, such as providing a user console and transferring files. It cannot be used as a normal network routing interface.

- A network-connected workstation that is attached to any IP-capable network interfaces of the adapters that are in the adapter slots.

  This configuration is not pictured. The network interface could be a LAN adapter such as Token-ring, 10/100Mbps Ethernet, or FDDI. It could also be any other Network Utility adapter, since all of them support IP routing. The Telnet workstation could locally or remotely connected.
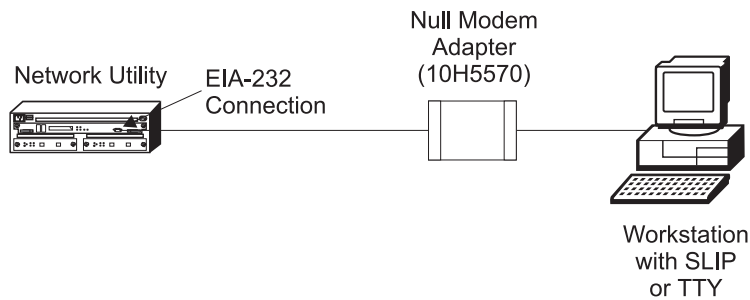
Figure 2-1. Local Workstation Serial Connection to the EIA 232 Port
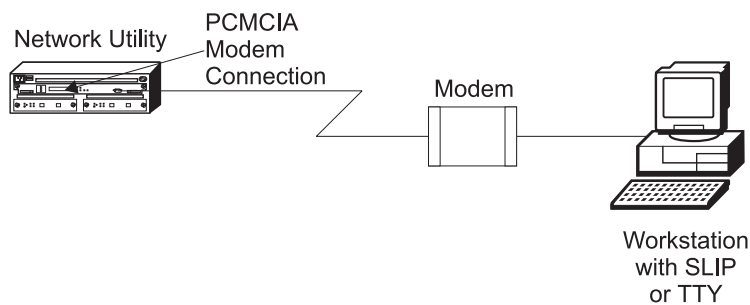
Figure 2-2. Remote Serial Connection to the PCMCIA Modem
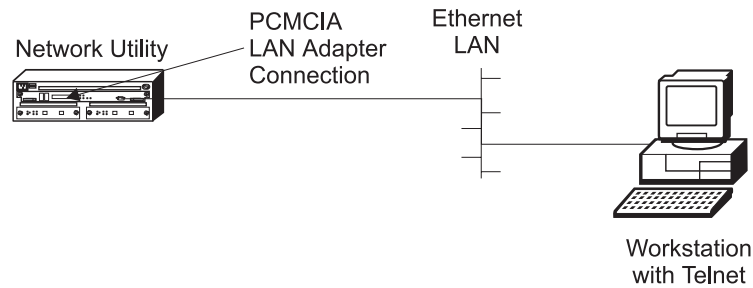
Figure   2-3. LAN Connection through the PCMCIA LAN Adapter

Which Access Method Should I Use?

If you are **a new user and are physically adjacent to the Network Utility** -
Attach a workstation directly to the unit (see Figure 2-1 on page 2-2) using ASCII
terminal emulation for your terminal console (see "ASCII Terminal Setup and
Usage"). The key advantages to this method are:

- Easy set up
- Works well with basic terminal emulation software
- It does not require the unit to be configured
- It provides a steady connection if you repeatedly configure and reboot the unit
  while you learn how to use the product
- It provides access to the firmware user interface, which you may want to learn
  about or use

If you are **a new user and are remote from the Network Utility** - Dial-up terminal
emulation is preferable to Telnet for some of the same reasons as for a new user
who is physically adjacent to the unit.

If you are **placing a configured Network Utility into a production network** -
Choose the terminal console access method that best fits your network
configuration and also your service and operations strategy. You can use Telnet as
the "everyday" terminal console access method, and dial-up terminal emulation as
the backup service method when either the network is not available or firmware
access is required. IBM service personnel will use either method that is available,
when they debug configuration and network problems.

ASCII Terminal Setup and Usage

Use this section if you are setting up an ASCII terminal or a workstation with
terminal emulation. You can use ASCII terminal emulation to access Network Utility
whether or not it has ever been configured.

An ASCII terminal console provides access both to the main operational code (the
command-line interface), and to the firmware user interface (see "Firmware" on
page 5-19). If you are remotely dialed-in and you reboot the unit, you will lose your
console connection and need to re-dial. If you are locally connected, your console
connection is maintained during a reboot.

Attaching an ASCII Terminal

Attach an ASCII terminal or emulator (with the appropriate emulation software) to provide local or remote access as shown in Figure 2-2 on page 2-2 and Figure 2-1 on page 2-2.

DEC VT100 and DEC VT220 ASCII terminals are supported, as well as devices such as personal computer systems that are configured to emulate them. Configure either with:

- No parity
- 8 data bits
- 1 stop bit
- 19.2 Kbps bit rate

> **Note:** The speed must match the speed of the connected terminal. You can modify the PCMCIA modem's speed as mentioned in "Serial Port and PCMCIA Modem Default Settings."

Serial Port and PCMCIA Modem Default Settings

These are the default settings for the serial port:

Speed   19.2 Kbps
Parity   None
Data Bits 8
Stop Bits 1

The PCMCIA modem is a standard item that is shipped with the Network Utility. The modem is a 33.6-Kbps V.34 data modem. It is set up with a default speed of 19.2 Kbps. For information on changing the default speed, see the *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual*. Use the instructions on managing the configuration in the section "Using Firmware."

ASCII Terminal Setup Attributes

This is a list of all the options required to set up a terminal for use with the Network Utility that is connected to the service port. Not every terminal (particularly 3151 and 3161) will have all these options. You should use the information to set the options that you can set on your terminal.

Terminal Settings and Function Keys

Baud Rate: 19200 bits per second

> **Note:** Baud rate is modifiable through the firmware as mentioned in "Serial Port and PCMCIA Modem Default Settings."

Parity: None
Stop bits: 1

Duplex: Full Duplex
Flow Control: XON/XOFF and RTS/CTS (see Note 1)

Screen Control: ANSI Full screen
Screen Width: 80 Characters
Screen Height: 24 Lines
Line Wrap: ON

Screen Scroll: ON

Carriage Return Translation: CR (0Dx)
Backspace Translation: Destructive

Notes:

1. Terminals and Terminal emulator programs which do not have flow control options should be set to "Permanent Request to Send."

2. Terminal emulators that require a terminal type selection should be set to VT-220.

Function Keys

When accessing the firmware, you will need to use function keys F1, F2, F3, and F9. Not all terminals or terminal emulators provide standard support for these function keys (for example the VT100 types). The following key sequences simulate function keys in some terminal emulation packages:

- Type **Ctrl-a** followed by the number (not the function key itself) to simulate the function key you want to press

- Type one of the following escape Key sequences:

```
Function 1 (F1):  <Esc> O P        Hex: 1B 4F 50
Function 2 (F2):  <Esc> O Q        Hex: 1B 4F 51
Function 3 (F3):  <Esc> O R        Hex: 1B 4F 52
Function 4 (F4):  <Esc> O S        Hex: 1B 4F 53

Function 6 (F6):  <Esc> [ 0 0 6 q  Hex: 1B 5B 30 30 36 71
Function 9 (F9):  <Esc> [ 0 0 9 q  Hex: 1B 5B 30 30 39 71
```

Note: In the definitions of function keys:

```
O = uppercase O
0 = the number zero
All characters are case-sensitive
```

Multiple Terminal Users

One user at a time can have an active terminal console through the system card serial port or the PCMCIA modem interface. If a workstation is connected locally to the serial port and a call comes in over the PCMCIA modem, priority is given to the call. After the call, the user at the local workstation will have to reconnect.

Telnet Setup and Usage

Use this section if you are setting up Telnet terminal console access.

Telnet provides access only to the main operational code (the command-line interface), and not to the firmware user interface. If you reboot the unit from the command-line interface, you lose your Telnet connection and you need to re-establish it after the unit has rebooted.

If your unit has never been configured, the only way you can Telnet to it is by using the default SLIP or PCMCIA EtherJet IP addresses (see Table 2-1 on page 2-1).

SLIP Addresses

The default SLIP IP addresses for use with the PCMCIA or external modems are:

For the workstation:

> 10.1.1.3

For the Network Utility:

> 10.1.1.2

For instructions about installing SLIP, refer to the documentation for your version of TCP/IP.

PCMCIA LAN IP Addresses

The default IP addresses for use with the PCMCIA EtherJet PC card are as follows:

For the workstation:

> 10.1.0.3

For the Network Utility:

> 10.1.0.2

You can change these addresses either from the operational code (the command-line) or from the firmware. (Use the procedures that are documented in "Basic IP Configuration and Operation" on page 4-8.) You must first bring up your initial user console using ASCII terminal emulation or by telnetting to the default IP addresses.

Network Interface IP Addresses

There are no default IP addresses for network interfaces (those on the adapters in the adapter slots). Use either the command-line interface or the Configuration Program to set up IP addresses for network interfaces. All the example configuration tables later in this book show how to set up IP addresses on interfaces. You cannot Telnet in through a network interface until you activate the IP address configuration change.

In addition to assigning IP addresses to an interface, you can assign one to the entire unit. This IP address is known as the *internal* IP address, and it remains active independent of the state of individual network interfaces.

If you have a Model TN1 and are using the TN3270 server function, you can configure the IP address and TCP port number to be used by TN3270. If you accept the default Telnet port number 23 for TN3270, you must attach your console Telnet sessions to a different IP address than the one you have configured for the TN3270 server. This allows the unit to distinguish console Telnet sessions from TN3270 client sessions.

Multiple Telnet Users

Two users at a time may bring up Telnet consoles through network interfaces. A third user's Telnet attempt will be rejected until one of the first two users has disconnected. One user at a time can have an active console through the system card service port or PCMCIA interfaces, including Telnet through SLIP or the PCMCIA LAN card.

Getting to the Command Prompt

After you have set up your user console, look for the messages and go to one of the command prompts described here.

What You Should See

If you have an active user console from the time you power on a Network Utility until it presents the first command prompt, you see a sequence of informational status messages about the following:

- Escaping to change the terminal type
- Memory initialization
- System board diagnostics
- Other diagnostics
- Terminating the boot (to reach the firmware menus)
- Loading the operational code from disk, ending with the following messages:

```
Loading /hd0/sys0/LMX.ld from disk ...
Loading /hd0/sys0/LML.ld from disk ...
Loading /hd0/sys0/sysext.ld from disk ...

Please press the space bar to obtain the console.
Loading /hd0/sys0/diags.ld from disk ...
Loading /hd0/sys0/snmp.ld from disk ...
Loading /hd0/sys0/router.ld from disk ...
Loading /hd0/sys0/appn.ld from disk ...
Loading /hd0/sys0/tn3270e.ld from disk ...

<you press the space bar>
Console granted to this interface

The Standalone Configuration Process. You are here because
No network devices configured.

Config (only)>
```

At any time after you see the prompt Please press the space bar to obtain the console, press the space bar to attach the Network Utility console process to your session. The system acknowledges this action with the message Console granted to this interface, and by displaying a command prompt after the code loading is complete.

If you are at a Network Utility that has never been configured, the system presents the command prompt Config (only)>. You can then proceed as described in Chapter 3, "Performing the Initial Configuration" to configure the Network Utility. If the Network Utility has been configured adequately, the system presents the command prompt ∗.

Only a directly-attached ASCII (emulated) terminal can show you all the messages from the entire boot sequence. If you are dialing in through a modem or telnetting in to bring up your user console, the Network Utility needs to be at least partially booted before it can respond to your connection attempt. When you do connect, the boot process may be in one of its later phases. The system grants you the console immediately, then gives you a command prompt after the boot process completes.

Solving ASCII Terminal Problems

Garbage, random characters, or the inability to connect your terminal to the Network Utility service port can have a number of causes. The most common cause of garbage or random characters is that the terminal baud rate is not synchronized with the Network Utility.

The Network Utility is always set to a specific baud rate, which by default is 19.2 Kbps. The only way to change this rate is through the firmware, so you must have a working console to change it. If your console in unreadable, try different baud rate values on the terminal side until you find the one that gives you readable status messages or command prompts.

Other causes of connection problems include:

- No null modem on the serial cable
- Defective terminal or Network Utility AC grounds
- Defective, incorrectly shielded, or incorrectly grounded cable between the terminal and the Network Utility.
- Defective terminal or terminal emulator
- Defective Network Utility system board.

Refer to "Service Terminal Display Unreadable" in the *Network Utility Service and Maintenance Manual* for more information on handling these problems.

Solving Telnet Problems

The most common telnet problem is the inability to reach the Network Utility through your IP network. You can use the standard debug tools ping and traceroute to determine what is happening. If you are trying to ping to the Network Utility internal IP address, you need to set up a host route in your workstation to that address, with the next hop being the interface IP address through which you will be entering the Network Utility.

You can also try to ping from the Network Utility back to your workstation. The firmware provides a way to do this from an EtherJet or SLIP port, and the operational code Console process provides a way to do this from network interfaces. See "Basic IP Configuration and Operation" on page 4-8 for a summary of these procedures.

Chapter 3. Performing the Initial Configuration

This chapter introduces the basics of configuring Network Utility, and gives specific procedures for configuring a new Network Utility. These procedures move the Network Utility from a passive state where it is waiting to be configured, to a state where it has active network interfaces and protocols.

Before using these procedures, you must connect a user console as described in Chapter 2, "Bringing Up a User Console."

Configuration Basics

A Network Utility configuration is a collection of data items that control how the software operates, including such elements as:

- What interfaces to activate
- What links to bring up
- What protocols and features to make active
- What functions in a given protocol or feature to make active
- What network addresses and names to use

When you boot up a Network Utility, the system reads its configuration information from a file on the hard disk, and activates interfaces and protocols according to the information in that file. You create the file in one of two ways:

- Using the command line interface from a user terminal console

  You type commands to create configuration data items in memory, then write the configuration to the Network Utility hard disk.

- Using a graphical configuration program that runs on a stand-alone workstation

  You create the configuration on the workstation, then transfer it to the Network Utility hard disk.

  The Network Utility Configuration Program is shipped on a CD-ROM in the carton with every new Network Utility, and is also downloadable from the Web. Versions are available for Windows 95 and NT, AIX, and OS/2. Workstation requirements are documented in the *Configuration Program User's Guide*, which is also shipped hard-copy in the carton with Network Utility.

Choosing Your Configuration Method

Some IBM routing product users prefer the Configuration Program, others prefer the command line interface, and still others use a combination of the two. The approach you take is up to you.

Here are some of the factors users cite in favor of the Configuration Program:

- It enables centralized maintenance of configuration files for multiple Network Utilities and 2216s
- It provides table-oriented, intuitive organization of data items
- It performs more input validation and cross-checking of parameters than the command line method
- It includes on-line helps for individual data items

Here are some of the factors users cite in favor of the command line interface:

- You can have a single integrated method for configuration, dynamic reconfiguration, and monitoring
- It is well documented in product publications and IBM "red books"
- It is simple to make and try quick configuration changes
- Setting up a user console does not require as many workstation resources or as much time as installing the Configuration Program

Getting Started from Config-only Mode

If you boot a Network Utility and see the `Config (only)>` prompt from the user console, you are in config-only mode. A Network Utility boots up into config-only mode when the current configuration file on the hard disk has no data items that would allow it to do any useful functions like forward data packets.[1] You need to configure at least one adapter port and one protocol (for example, IP, DLSw, or APPN) and reboot, in order for the Network Utility to start up in normal working mode.

If you have a Network Utility at the `Config (only)>` prompt, do the following:

1. Choose whether you want to use the command line or the Configuration Program for your initial configuration. It is easy to switch methods later if you want to try both.

2. Based on your choice, follow one of these procedures:

   - "Command Line Procedure for Initial Configuration" on page 3-2
   - "Configuration Program Procedure for Initial Configuration" on page 3-5

Command Line Procedure for Initial Configuration

Use this procedure to configure a Network Utility for the first time starting from the `Config (only)>` command line prompt:

Part 1 - Create a minimal, basic configuration

1. Use the **add device** command to configure at least one network interface as follows:

   a. Type **add dev ?** to see a list of supported adapter types.

   b. Type **add dev** *type*, where *type* consists of the first few letters from a row of the adapter list. For example, **add dev tok** selects the token-ring adapter. Type enough letters to uniquely identify the adapter you want.

   c. When prompted for slot number, enter **1** for the left-hand adapter slot of the Network Utility, or **2** for the right-hand slot.

   d. If you are adding a multiport adapter, the system prompts you for the port number of the interface you want to configure. Port numbers on adapters are fixed as follows:

---

[1] This also happens if your configuration becomes corrupted.

- Ports on multi-port LAN adapters are numbered 1 and 2 and are labelled on the adapter face.
- Ports on multi-port WAN adapters are numbered starting with 0 and are labelled on the connectors at the end of the adapter cable.

e. The system then assigns a logical *interface number*, also known as a *net number*. This is the key number by which you refer to this interface on every other command in the system. For example, if you want to delete the configuration for this interface, type **delete interface** and then give the logical interface number.

f. If necessary, make the following adjustments to the default device configuration:

If you added a Token-ring port and you want it to run at 16Mbps instead of the default 4Mbps, type these commands:

> **net** *interface number*
> **speed 16**
> **exit**

If you added a 10Mbps (not 10/100) Ethernet port and you want to use the BNC (10BASE2) connector instead of the default RJ45 (10BASET) connector, type these commands:

> **net** *interface number*
> **conn bnc**
> **exit**

Repeat step 1 for each interface you want to configure.

2. Use the **qconfig** command to start the "Quick Config" program. Use this program to configure IP and SNMP access to Network Utility as shown below.

Quick Config is a feature of the command line configuration process. Instead of waiting for you to type commands, it asks you questions and creates configuration data based on your replies. An example Quick Config question is:

```
Configure Bridging? (Yes, No, Quit): [Yes]
```

The values in parentheses are the expected responses. The value in square brackets is the default response. To accept the default, press **Enter**.

Respond as follows to the Quick Config questions (some of these are default responses):

a. Configure Bridging

   1) Enter **no** to `Configure Bridging?`

b. Configure Protocols

   1) Enter **yes** to `Configure Protocols?`

c. Configure IP

   1) Enter **yes** to `Configure IP?`

   2) For any interfaces to which you want to assign an IP address, enter **yes** to `Configure IP on this interface?` If you intend to use the PCMCIA EtherJet card as your only IP interface, you can answer **no** for every configured network interface.

   3) Enter the IP address at the `IP Address` prompt.

4) Enter the IP mask at the `Address Mask` prompt.

5) If you want to enable RIP or OSPF, respond **yes** to `Enable Dynamic Routing?`, and respond to subsequent related questions.

6) If at some point you may want to send a configuration directly from the Configuration Program to this Network Utility, enter **yes** to `Define Community with Read_Write_Trap Access?` Enter any single-word name you want as the community name.

   If you never expect to use the Configuration Program, respond **no**.

7) Enter **yes** to `Save this configuration?` This saves the IP part of the configuration in memory.

d. Save the Configuration file

1) Enter **yes** to `Do you want to write this configuration?`

Part 2 - Activate the new configuration

At this point, you have now configured at least one interface and one protocol (IP, with SNMP). This small configuration is sufficient to leave config-only mode.

1. From the `Config (only)>` prompt, type **reload** and respond **yes** to the confirmation prompt. The Network Utility reboots and activates your new configuration.

   If you see a prompt about saving configuration changes, that means you have made some configuration changes after saving the configuration file in step 2d. Type **yes** to save these changes as part of your new configuration before the reboot proceeds.

2. Verify the Network Utility reboot

   If your user console is through a dial or telnet connection, reboot causes you to lose your connection. Re-connect after a few minutes. Otherwise just watch the boot messages from your user console.

   When the reboot completes, your console should display the * command prompt, indicating that you are in normal operating mode and no longer in config-only mode. The configuration you created in Part 1 of this procedure is now active.

Part 3 - Add additional protocol information

You are now in normal operating mode with the interfaces you configured, running only IP.

If you are a new user and want to become familiar with the product before configuring the rest of your functions (such as TN3270 or DLSw), skip the rest of this procedure and see the guidelines in "What to Do Next" on page 3-9.

If you want to configure all your functions right now, continue here.

1. Select the configuration scenario from the "Configuration and Management Specifics" part of this guide that most nearly resembles the use to which you are placing this Network Utility.

   • Model TN1 Users - See Chapter 12, "TN3270E Server."

- Model TX1 Users - See Chapter 14, "Channel Gateway" or Chapter 16, "Data Link Switching."

If none of these scenarios is suitable, use the *MAS Protocol Configuration and Monitoring Reference* and *MAS Software Users Guide* manuals to determine what you need to configure.

2. In the "Example Configuration Details" chapter that follows your selected scenario, find the configuration parameter table that corresponds to that scenario. Use the "Command Line Commands" column to guide you in configuring that scenario, changing the values for your particular adapters and network.

If you find that you are having trouble navigating the command line and entering commands, you may want to get more familiar with general command line configuration before proceeding. See "What to Do Next" on page 3-9 for suggestions on how to proceed.

3. When you are done entering configuration commands, repeat the steps in "Part 2 - Activate the new configuration" on page 3-4, but issue the **reload** command from the * prompt instead of the `Config (only)>` prompt.

Configuration Program Procedure for Initial Configuration

Use this procedure to configure a Network Utility for the first time using the Network Utility Configuration Program.

Part 1 - Create the configuration at the Configuration Program

1. From the Configuration Program CD-ROM, install the appropriate version of the Configuration Program onto your workstation.

   For installation instructions, see:

   - The Network Utility README file on the CD-ROM.
   - The *Configuration Program User's Guide*, which is shipped along with the CD-ROM.

   Start the Configuration Program. If you want to try the program by doing a new configuration from scratch, select **New configuration** and **Network Utility** from the **Configure** option on the menu bar in the Navigation Window.

2. Select the configuration scenario from the "Configuration and Management Specifics" part of this guide that most nearly resembles the use to which you are placing this Network Utility.

   - Model TN1 Users - See Chapter 12, "TN3270E Server."
   - Model TX1 Users - See Chapter 14, "Channel Gateway" or Chapter 16, "Data Link Switching."

   If none of these scenarios is suitable, use the *MAS Protocol Configuration and Monitoring Reference* and *MAS Software Users Guide* manuals to determine what you need to configure. Use any of configuration parameter tables in "Configuration and Management Specifics" of this book as an example of

---

2   If no corresponding table exists, use the "Keys to Configuration" section for that scenario, to get started.

mapping command line commands to Configuration Program panels. When you are done with your configuration, move to step 7 on page 3-6.

3. In the "Example Configuration Details" chapter that follows your selected scenario, find the configuration parameter table that corresponds to that scenario.[2]

4. From your web browser, follow the `Downloads` link from the main Network Utility web page `http://www.networking.ibm.com/networkutility`, and find the example configuration file that matches your selected scenario. Download this file in binary and transfer it to the workstation running the Configuration Program.

5. Select **Open Configuration ...** from the Navigation Window and select the path and file name of the example configuration file you downloaded.

6. Use the "Configuration Program Navigation" and "Configuration Program Values" columns in the table from step 3 to guide you in moving through the configuration and changing the values for your particular adapters and network.

7. When you have a configuration ready to send to your Network Utility, do a **Save configuration as ...** operation to save the configuration on your workstation. You may want to choose a new name so you can leave the original example configuration file unchanged.

Part 2 - Transfer the configuration to the Network Utility and activate it

At this point, you have created the initial configuration. All that remains is to transfer the configuration to the Network Utility hard disk and reboot the Network Utility to activate it. How you should do this transfer depends on your connection setup, as follows:

- If your Configuration Program workstation supports TCP/IP and has physical connectivity to either the Network Utility PCMCIA EtherJet card or a network adapter in slot 1 or 2, use Procedure A.

- If your user console is via ASCII terminal emulation and you prefer using XMODEM to setting up the above IP connectivity, use Procedure B.

You can also refer to "Loading New Configuration Files" on page 7-4 for a complete list of all the ways to transfer a configuration to Network Utility. You will need TFTP server software on a TCP/IP workstation if you choose not to follow either Procedure A or B.

Procedure A - Direct transfer through Network Utility PCMCIA EtherJet or a network adapter

Use this procedure if your Configuration Program workstation supports TCP/IP and has physical connectivity to the Network Utility PCMCIA EtherJet card or a network adapter in slot 1 or 2.

1. Configure Network Utility quickly from the command line, so that it has an IP address on at least one interface, and IP and SNMP enabled

   a. From your user console, perform the steps from Part 1 of Procedure A above. Be sure to:

      - Use **add device** to define at least one interface in slot 1 or 2

- In Quick Config, add an IP address to at least one network interface (even if you plan to use PCMCIA EtherJet)
- In Quick Config, respond **yes** to `Define Community with Read_Write_Trap Access?`

b. In the Configuration Program, verify that the configuration you are about to send has SNMP enabled and the same community name defined. This is required so that after you activate this configuration, you will be able to repeat step 3 of this procedure to send another configuration.

c. Perform the steps in "Part 2 - Activate the new configuration" on page 3-4 of Procedure A to reboot Network Utility and activate this temporary command line configuration.

2. If you plan to use the PCMCIA EtherJet card, set up its IP addresses as follows after the Network Utility reboot is complete:

From the `*` prompt, type **talk 6**. From the `Config>` prompt, type **system set ip** and enter the following values as prompted:

- IP address: the IP address you want to use for the EtherJet card
- Netmask: the mask for the subnet attached to the EtherJet card
- Gateway address: the IP address for the Configuration Program workstation, or the IP address of a router through which the Network Utility can reach it

Next to each prompt, the system shows the current value as the default. To accept the default, press **Enter**. When you are done entering values, any address change you specified takes effect immediately. The values are stored in Network Utility NVRAM and not as part of any configuration file.

3. Send the configuration from the Configuration Program (using SNMP)

a. From the **Configure** drop-down menu, select **Communications** and **Single router**

b. On the `Communicate` panel, enter:
- IP address or name - the IP address of the Network Utility interface you want to send the configuration through. This is either the PCMCIA EtherJet IP address, or the network interface IP address you assigned in Quick Config.
- Community - the community name you assigned in Quick Config.

c. Select **Send configuration** and **Restart router**. Accept or enter the current date and time, so that Network Utility will reboot with the new configuration immediately after receiving it.

d. Click on **OK**, and the Configuration Program immediately starts sending configuration data items to the specified router(s) using SNMP.

The Configuration Program provides status and result messages about the transfer. If the send operation fails, the Configuration Program lists some possible reasons which you should then verify and correct.

After the Configuration Program completes its configuration transfer, the Network Utility stores the configuration on disk and reboots itself as you directed.

4. Verify the Network Utility reboot

If your user console is through a dial or telnet connection, reboot causes you to lose your connection. Re-connect after a few minutes. Otherwise just watch the boot messages from your user console.

When the reboot completes, your console should display the * command prompt, indicating that you are in normal operating mode and no longer in config-only mode. The configuration you created in Part 1 of this procedure is now active.

Procedure B - Indirect XMODEM transfer through user console session

Use this procedure if your user console is via ASCII terminal emulation and you prefer using XMODEM to setting up IP connectivity from the Configuration Program workstation.

1. From the Configuration Program, export your configuration into the file format understood by Network Utility

   From the **Configure** drop-down menu, select **Create router configuration** and specify the path and file name for a .cfg file. Click on **OK** to write the file.

2. If necessary, transfer the .cfg file from the Configuration Program workstation to your terminal emulation workstation.

3. From your user console at the `Config (only)>` prompt, follow this sequence:

   ```
   Config (only)>boot
   Boot configuration
   Boot config>dis auto
   AutoBoot mode is now disabled.

   Operation completed successfully.
   Boot config>exit
   Config (only)>rel y
   ```

   If you are prompted about saving configuration changes, reply **no**. Network Utility reboots and stops at the firmware menu.

   If your user console is through a dial or telnet connection, reboot causes you to lose your connection. Re-connect after a few minutes and you see the firmware menu.

4. Make the following sequence of firmware menu selections:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 12, "Change Management"
   c. Change Management Software Control: Option 12, "XMODEM software"
   d. Select Type: "Config"
   e. Select Bank: choose Bank A (active bank)
   f. Select Config: choose position 1[3]

   The firmware tells you when to start the file transfer.

5. Go to your terminal emulation package and start the transfer of the file from your workstation server, using whatever name you like. When the Network Utility has received the configuration file, the status of the file position will

---

[3] This selection of bank and configuration file position assumes that this is the first time you have booted this Network Utility. For more background on this topic, see "Configuration Files" on page 6-2.

change from CORRUPT to AVAIL. You can verify that this has happened using option 7, "List Software", from the firmware Change Management menu.

6. Boot the Network Utility using the configuration you just loaded

   a. Use Option 9 "Set Boot Information" to select the current op-code bank and the new configuration.
   b. Press **esc** to reach the main menu, then **F9** (Start OS) to boot the Network Utility with the new configuration.

7. Verify the Network Utility boot

   If your user console is through a dial connection, you do not lose the connection when you use the Start OS option. Watch the boot messages from your user console.

   When the boot completes, your console should display the ∗ command prompt, indicating that you are in normal operating mode and no longer in config-only mode. The configuration you created in Part 1 of this procedure is now active.

8. Follow this command line sequence to reset the auto-boot mode so you will not re-enter the firmware with each reboot:

```
*talk 6
Config>boot
Boot configuration
Boot config>en auto
AutoBoot mode is now enabled.

Operation completed successfully.
Boot config>exit
Config>     <Ctrl-p>
*
```

What to Do Next

If you have followed the procedures in this chapter, your Network Utility is now in full operational mode with a configuration you created. With your user console at the ∗ prompt, you are now in a position to use the command line interface to:

- Query the status of interfaces and protocols
- Activate events and monitor the event log
- Issue operator commands to effect status changes
- Make dynamic configuration changes without rebooting

These are the basic tools to see whether your new configuration is working properly, and make to small adjustments to that configuration.

If the command line interface is new to you, you can use Chapter 5, "A Guided Tour through the Command-Line Interface" at a tutorial to get familiar with its concepts and how to use it.

If you have some previous experience with IBM routing products or prefer to try tasks without following a tutorial, you can use Chapter 4, "Quick Reference to the User Interface" as summary information about command line navigation and some common tasks.

You can use Chapters 6 through 10 in "Learning About Network Utility" to get more background on:

- Managing configuration files
- Dynamic reconfiguration
- Managing what Network Utility is doing, both locally and using remote network management products
- Updating software and firmware
- Requesting service and support

You may have already used the example configuration information in the "Configuration and Management Specifics" part of this book. The chapters there also contain introductory information about configuring and monitoring the functions:

- TN3270E server
- Channel gateway
- Data Link Switching

If you have already configured one of these functions in your initial configuration, use the "Managing" section from the corresponding chapter to begin monitoring and debugging that configuration.

Chapter 4. Quick Reference to the User Interface

This chapter contains summary information about navigating the command line interface, entering commands, and performing common tasks. For a more complete explanation of this material with examples, see Chapter 5, "A Guided Tour through the Command-Line Interface"

Navigating

The command line interface consists of a tree of menus whose root is the *
(asterisk) prompt. You type commands and use control keys to move to various places in the tree, then you type commands to actually perform tasks.

Processes and Prompts

From the * prompt, use the **talk** command (abbreviated **t**) to attach to one of several processes, or ways of viewing the system. Each process from which you enter commands is identified by a different command prompt.

Table 4-1. Key Processes			
Name	**Command to Access**	**Purpose**	**Top-level Prompt**
Config	**t 6**	View and modify the configuration	`Config>`
Console	**t 5**	View and control running status, make dynamic configuration changes	+ (plus sign)
Monitor	**t 2**	View real-time event message log	(none)

After typing **t** *n*, press **Enter** twice to obtain the command prompt. Type **Ctrl-p** to return to the * prompt from inside any process.

The monitor process has no command prompt because instead of issuing commands in that process, you watch a running log of event messages. You can type **Ctrl-s** to pause scrolling, and **Ctrl-q** to resume it.

Subprocesses

When you are working inside the talk 6 or talk 5 processes, some commands change the input prompt and provide you a new command menu that is specific to a functional area. For example,

- Typing **protocol dlsw** under talk 6 moves you to the Config subprocess for configuring Data Link Switching. The command prompt becomes `DLSw config>`.
- Typing **perf** under talk 5 moves you to the Console subprocess for viewing CPU utilization statistics The command prompt becomes `PERF Console>`.

You can also move from one subprocess into another subprocess. For example, typing **ban** from the DLSw Config subprocess moves you to the Boundary Access

Node Config subprocess. You have gone one nesting level deeper in the menu system; you must return through the DLSw subprocess.

The following navigation rules apply:

- To enter a subprocess, type the specific command that takes you there. Type **?** at any menu to see the available commands. You know you have entered a subprocess when the command prompt changes.

- To leave any subprocess and return to the next higher level menu, type **exit**.

- To leave any subprocess and move immediately to the * prompt, type **Ctrl-p**. This also takes you out of the current process.

- To resume a subprocess after having typed **Ctrl-p**, type **t** *n* (where *n* is the process number you left), then **Enter** twice. You resume that process in the subprocess where you typed **Ctrl-p**.

Entering Commands

You type commands to enter processes, enter and leave subprocesses, and to perform tasks. Some task commands prompt you for parameter values, while others do not require any input other than the command name.

Forming Commands

A command is a sequence of one or more key words, optionally followed by typed-ahead parameter values. The following guidelines apply to forming a command:

- You must type a complete command before the system takes the action or prompts you for input parameters. If you type only part of a valid command (not enough key words), the system responds with `Command not fully specified`.

- You can type **?** at any process or subprocess prompt, or after any incomplete command, to see a menu of command keywords available from that point. You can use this to find or complete a command, as shown in this abbreviated example:

```
Config>?
ADD (device, user)
BOOT and load file functions
CHANGE (device, password, user)
    :              < other commands not shown>
Config>add
Command not fully specified
Config>add ?
DEVICE
NAMED-PROFILE
PPP-USER
TUNNEL-PROFILE
USER
Config>add user
Enter user name:  []?  <enter>
No user was added
Config>
```

In the example, **add** was not a complete command, but **add user** was. After the user typed the complete command, the system prompted for an input parameter value.

- You can abbreviate any command keyword to the minimum number of characters that uniquely select it from the menu on which it appears. For example, you can type **t 6** instead of **talk 6**, and **p appn** instead of **protocol appn**. In the example above, the user could have typed **a u** instead of **add user**.

- You can work with previously-entered commands in both talk 6 and talk 5 using the following keys:

  Ctrl-B   to scroll backward through previously entered commands

  Ctrl-F   to scroll forward through the list of previously entered commands

  Ctrl-U   to clear a retrieved command off the command line

  Backspace to edit a retrieved command starting from the end

  The command history buffer is shared by talk 6 and talk 5.

Entering Command Parameter Values

Some of the commands that perform a task require you to supply values for input parameters. You can either let the system prompt you for these input values, or type them ahead on the command line following the command name.

If you do not type parameter values ahead:

- You type only the command name and press **Enter**.

- The system prompts you for each parameter in turn, supplying the default value for that parameter inside square brackets. Some defaults are fixed, but most are the last value you assigned to that particular parameter.

  - To accept the default value, press **Enter**
  - To supply a new value, type the value and press **Enter**
  - If the brackets are adjacent, as in [], there is no default and you need to supply a value

  The system performs a validity check on your response before prompting you with the next value.

- When you have responded to the final parameter prompt, the system takes the action specified by the command.

If you want to type parameter values ahead:

- You type the command name following by one or more parameter values separated by blanks, then press **Enter**.

- The system parses the command line and supplies the first value to the first parameter, the second value to the second parameter, and so on. You must supply the values in the order expected.

  The system performs a validity check as it assigns each value to the corresponding parameter.

- If the command requires more parameters than you have supplied values for, the system prompts you for the additional values as outlined above.

- When the system has supplied a valid value to each parameter, it takes the action specified by the command.

Typing values ahead can be a convenient short-cut for experienced users. You need to be careful that you provide valid parameters in the right order.

You should be alert for cases where you type **?** following a full command, and the command treats the "?" as a typed-ahead value for its first input parameter. If this happens, abort or undo the command and try again.

Common Error Messages

The following table explains several standard error messages from the command line interface:

Error Message	Explanation and Corrective Action
`Command error`	The command you typed does not exist on the current menu. You may have a typo, or be in the wrong place to issue this command, or not have typed enough characters to identify the command from the menu.
	Look at your prompt to verify where you are, and type **?** to see the available commands. Correct the command or move to the right place.
`Command not fully specified`	The command keywords you typed do not form a complete command.
	Type **Ctrl-b** to retrieve the command, then add " **?**" to the end of it to see your choices for the next keyword. Pick the next keyword to add and re-issue the command replacing **?** with that keyword.
	You may also want to consult the appropriate MAS command line reference manual for the command you are trying to enter.
`Command syntax error`	You typed an improper form of a valid command. You may have supplied an invalid or unexpected parameter.
	Try the command again with no parameter values, or consult the appropriate MAS command line reference manual entry.
`Feature <name> available but not enabled`	Under talk 5, you tried to enter the Console subprocess for a feature that is supported in your software load but is not actively running. Your current configuration either did not enable the feature, or is missing key values required in order to activate the feature.
	If you are using the Configuration Program, look on the Navigation Panel for **?**s, indicating required parameters not set. Follow the **?** trail to the panel(s) with field names in red that are not set.
	If you are doing configuration from the command line, consult the example configurations in this book and in the MAS reference manual chapter for this feature. Look for the key parameters that are shown as base parameters for enabling the function.
`Protocol <name> available but not configured`	The same as described above for `Feature available but not enabled`, but applied to a protocol.

Key User Tasks

This section organizes common user tasks into groups and provides tables with a quick reference to the commands to perform each task.

Configuring Physical Adapters and Interfaces

Task	How to do it
Add an interface at initial configuration	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **add dev ?** to see a list of supported adapter types. 3. Type **add dev** *type*, where *type* is the keyword from the list for the adapter type you want. 4. Enter the physical slot and port number (if asked) of the interface you are configuring. Slots are 1 and 2 from left to right. LAN ports are numbered on the adapter face, and WAN ports are numbered on the cable connectors. 5. Note the new logical interface (net) number the Network Utility assigns to this interface. 6. Type **net** *logical interface number* to enter the Config subprocess for the particular interface type. Use the commands in that subprocess to verify or change from the default settings for the interface. 7. Type **exit** to return to the `Config>` prompt. 8. Type **write** to save this configuration, then **reload** followed by **yes** to reboot with it.
Enable the dynamic addition of interfaces after initial configuration	Before you can add an interface dynamically, the active Network Utility configuration must have "spare interfaces" defined. 1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **set spare** and enter the number of spare interfaces you want. 3. Type **write** to save this configuration, then **reload** followed by **yes** to reboot with it.

Task	How to do it
Add an interface dynamically after initial configuration	1. Verify that you have active spare interfaces: a. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. b. Type **int** and verify that you have NULL interfaces. c. Type **Ctrl-p** to return to the * prompt. If you have no spare interfaces, you must follow the procedure above to add some to your configuration and reboot. 2. From the * prompt, type **talk 6** and press **Enter** twice to reach the Config> prompt. 3. Use **add dev** and **net** commands to configure a new interface, as described in the initial configuration procedures. Note the new logical interface number assigned by the **add dev** command. 4. Use the **protocol** and **feature** commands to move to Config subprocesses and configure protocol information relating to the new interface. 5. Type **Ctrl-p**, **talk 5**, and press **Enter** twice to reach the + prompt. 6. Type **activate int** and give the new logical interface number. The system activates the new interface dynamically. 7. If you want to save the new interface configuration so it will survive a reboot, go back to talk 6 and type **write** to write the modified configuration to disk. Or, make the corresponding changes at the Configuration Program and download the revised configuration to the Network Utility.
Dynamically change interface configuration	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the Config> prompt. 2. Type **list dev** to see the logical interface number for the interface you want to change. 3. Type **net** *logical interface number* to enter the Config subprocess for the specific interface type. 4. Enter commands to change the configuration of the interface. 5. Type **Ctrl-p**, then **talk 5** and press **Enter** twice to reach the + prompt. 6. Type **reset ?** and enter the number of the interface you just reconfigured. Network Utility takes the interface down and brings it back up using the modified configuration. 7. If you want to save these configuration changes so they will survive a reboot, go back to talk 6 and type **write** to write the modified configuration to disk. Or, make the corresponding changes at the Configuration Program and download the revised configuration to the Network Utility.

Managing Physical Adapters and Interfaces 4-7

Task	How to do it
Look at the status of an interface	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **config** to see information about the software and, at the end, the current state of all interfaces. If the display output pauses with --More-- displayed, press the space bar to see the next screen of output. 3. Type **stat** to see packet and byte statistics for interfaces. 4. Type **err** to see error counts for interfaces interfaces. 5. Type **net** *logical interface number* to enter the Console subprocess for the specific interface type. Use the commands in that subprocess to display type-specific interface status information.
Recycle (disable/enable) an interface	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **int** to see the logical "net" number for the interface you want to recycle. 3. Type **disable int** *logical interface number* to take the interface offline dynamically. 4. Type **test** *logical interface number* to bring the interface back up.
Recycle (disable/enable) an adapter	Note: If you intend to remove the adapter while it is disabled (the standard "hot plug" procedure), you should also refer to the "Removal and Replacement Procedures" chapter in the 2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual. 1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **disable slot** *slot number*, where 1 is the left-hand slot and 2 is the right-hand slot. This disables all interfaces on the adapter in that slot. 3. Type **enable slot** *slot number* to activate all interfaces on the adapter in that slot.

Basic IP Configuration and Operation

Task	How to do it

Add an IP address to a network adapter	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt.
	
	 2. Type **prot ip** to reach the IP Config subprocess.
	
	 3. Type **li addr** to see currently configured IP addresses.
	
	 4. Type **add addr** to add an IP address. Supply the logical interface (net) number of the interface, the IP address, and the address mask.
	
	 5. If you want to activate this and other IP configuration changes in the running Network Utility:
	
	    a. Type **Ctrl-p**, then **talk 5** and press **Enter** twice to reach the + prompt.
	    b. Type **prot ip** to reach the IP Console subprocess.
	    c. Type **int** to see currently active interface IP addresses.
	    d. Type **reset ip** to activate the new address.

Set the IP address of the PCMCIA EtherJet adapter	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt.
	
	 2. Type **system set ip** and supply the following information (defaults are the current values of these parameters):
	
	    • IP address - the address to be used for the EtherJet adapter
	    • IP netmask - the network mask for that address
	    • IP gateway address - the address of the IP workstation you are likely to communicate with, or the router you use to reach that workstation.
	
	 Any changes you make take effect immediately, and are stored in Network Utility non-volatile memory. These addresses are not part of the Network Utility configuration.
	
	 You can also set the EtherJet IP address from the firmware. Follow the procedure below for EtherJet Ping, but select option 1 "IP Parameters", instead of option 3 "Ping".

Add a static route	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **prot ip** to reach the IP Config subprocess. 3. Type **li route** to see currently configured routes. 4. Type **add route** to add a static route. Supply the information requested.
Ping and traceroute from a network adapter	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **prot ip** to reach the IP Console subprocess. 3. To ping an address with default parameters, type **ping** *ip address*. To modify parameters, type only **ping** and respond to the prompts. Type **Ctrl-c** to end the ping. 4. To trace the route to an address with default parameters, type **trace** *ip address*. To modify parameters, type only **trace** and respond to the prompts. Type **Ctrl-c** to end the traceroute.

Task	How to do it
Ping from the PCMCIA EtherJet adapter	1. Use one of the procedures in "Boot Options: Fast Boot and Reaching Firmware" on page 4-12 to reach the firmware main menu.
	2. Bring up the panel from which you do a Ping
	a. Select option 4, "Utilities". b. Select option 11,"Remote Initial Program Load Setup". c. Select option 3, "Ping". d. Select the PCMCIA Ethernet interface.
	3. Enter the IP addresses you want to use for the ping (these temporarily override the configured addresses) and press **Enter**.

Managing the Command Line Configuration

Task	How to do it
Erase the configuration for a protocol, or for all protocols	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **clear ?** to see a list of sets of configuration information you can clear with a single command. 3. Type **clear** *protocol name* to clear information for a particular protocol, or **clear all** to clear information for all protocols (but not device information). These commands change the current configuration in memory but do not affect the operational state of the Network Utility.
Erase the configuration for an interface, or for all interfaces	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **del int** if you want to delete the configuration for a particular interface, including all protocol configuration related to that interface. 3. Type **clear dev** if you want to delete the configuration for all interfaces. This command does not clear associated protocol information, so you would normally use it with **clear all** to completely erase a configuration. These commands change the current configuration in memory but do not affect the operational state of the Network Utility.
Activate the entire current talk 6 configuration	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **write** to write the current configuration in memory to disk in the next available configuration file position of the active bank. 3. Type **reload** then **yes** to reboot Network Utility and activate that configuration. If you activate a configuration with no protocol or no device information, the Network Utility will enter config-only mode. You will have to define one protocol and one interface and reboot before the Network Utility can be fully operational.

General Status Monitoring

Task	How to do it
Look at CPU utilization	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **perf** to reach the performance monitoring Console subprocess. 3. Type **list** and verify that the CPU Monitor State is ENABLED. This is the default setting for Network Utility. If the state is not ENABLED, type **enable cpu**. 4. Type **report** to see recent CPU utilization statistics. The most current snapshot is the value "Most recent short window". 5. If you want CPU utilization to be reported every so often as an event message you can monitor with talk 2, type **enable t2**. Type **Ctrl-p** and **talk 2** to watch CPU utilization messages being generated. Type **Ctrl-p** to exit talk 2. 6. If you want the talk 2 CPU reporting to be continued after your next reboot, move to talk 6 and repeat the above commands. Or, configure the same settings on the CPU Utilization panel from the Configuration Program, and transfer the updated configuration to the Network Utility.
Look at memory utilization	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **mem** to see current global memory statistics. This command reports the total physical installed memory, and details about the part of memory used by the routing function. The routing function includes all network protocols and features except APPN and TN3270 server. 3. If you are running APPN or TN3270 server, type **p appn** to reach the APPN Console subprocess. Type **mem** to see current APPN memory statistics and threshold states. TN3270 server usage is included in these statistics, even if you are running only subarea TN3270 host attachment.
Turn on default ELS messages	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **event** to reach the event logging Console subprocess. 3. Type **disp sub all** to activate the STANDARD level of logging for all defined subsystems. This includes error messages and uncommon informational messages. 4. Type **Ctrl-p** then **talk 2** to watch any messages being generated, and **Ctrl-p** to exit talk 2. 5. If you want these settings to be maintained after your next reboot, move to talk 6 and repeat the above commands. This will make the settings part of your configuration.

Boot Options: Fast Boot and Reaching Firmware

Task	How to do it
Minimize boot time in a test environment	1. Type **talk 6** and then **boot** to reach the boot Config subprocess. 2. Type **en fast** to enable the fast boot option. The next time you reboot the Network Utility, it will boot more quickly by skipping some of the power-on diagnostics. This option is not recommended for production environments. You can use **dis fast** to go back to the normal full diagnostic mode.
Reach the firmware if you have a directly connected terminal console	1. Make sure your terminal emulation screen size is set to 24 rows by 80 columns. 2. From the * prompt, type **reload**, then **yes** to the confirmation message. Start watching the boot status messages closely. 3. When you see the message `Starting Boot Sequence` followed by `Strike F1 key now to prematurely terminate Boot`, type **Ctrl-c** or **F1** immediately. 4. Within a few seconds, you should be either at the firmware main menu or at a prompt for a supervisory password. If neither of these appear and you see disk load messages, you waited too long and missed the time window for typing **Ctrl-c** or **F1**. Wait for the boot sequence to complete, then repeat steps 2 and 3 of this procedure. Or, use the dial-in procedure to ensure you will stop in the firmware without having to press a key at the right time. 5. If the system prompts you for a supervisory password, enter the current password, originally set to "2216" at the factory. The system then presents the firmware main menu.
Reach the firmware if you have a dialed-up terminal console	1. Make sure your terminal emulation screen size is set to 24 rows by 80 columns. 2. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 3. Type **boot** to reach the boot Config subprocess. 4. Type **disable auto-boot** to select the mode where a boot sequence will always stop at the firmware. 5. Type **Ctrl-p** to reach the * prompt, then **reload yes** to reboot Network Utility. The reboot causes you to lose your dial connection. 6. After a few minutes, dial back in and you should be either at the firmware main menu or at a prompt for a supervisory password. 7. If the system prompts you for a supervisory password, enter the current password, originally set to "2216" at the factory. The system then presents the firmware main menu. Unless you want to stop in the firmware every time, do an **enable auto-boot** the next time you reach the operational code.

Task	How to do it
Boot from the firmware into the operational code	1. From within the firmware menu structure, press **\<Esc\>** as required to reach the firmware main menu. 2. If you want to continue the current boot sequence up into the operational code, press **F9** (Start OS). If you want to completely reboot starting from power-on diagnostics press **F3** (Reboot). This will cause you to lose your connection if you are dialed into the Network Utility PCMCIA modem or system card service port. 3. Dial back in if necessary, or just monitor the disk load messages. Press the space bar to obtain the command prompt if the system asks you to do so.

Chapter 5. A Guided Tour through the Command-Line Interface

This chapter is a tutorial to walk users who are new to IBM routing products through the concepts and basic navigation of the Network Utility command line interface. It covers:

- Basic concepts of adapter and port numbering
- How to move to different parts of the system and what each is for
- Example tasks and commands from different processes
- How to navigate menus and issue commands
- How to configure, query status, and watch the system log
- Basic concepts of saving and activating configuration changes
- What firmware is, how to get to it, and a few things you can do with it

The tutorial text makes the most sense if you follow it from beginning to end with the same Network Utility.

If you already have experience with the IBM 2216, you will find the Network Utility interface to be nearly identical. IBM 2210 users will find familiar prompts and menu navigation, but differences in areas including configuring adapters, saving configurations, and rebooting the product.

Prompts and Processes

If you followed one of the initial configuration procedures in Chapter 3, "Performing the Initial Configuration" on page 3-1, you have configured your Network Utility and booted it into normal operating mode. Your user console should show the command prompt '*'.

In normal operating mode, the routing function in Network Utility is running. You as the operator can use the command line interface to look at and modify the configuration, view the active system status, look at the message log etc.. You navigate to different parts of the command line interface to perform these different tasks, and the * prompt is the root of the navigation tree.

Type **?** from the * prompt to see the commands available from this point:

```
*?
DIAGS hardware diagnostics
DIVERT output from process
FLUSH output from process
HALT output from process
INTERCEPT character is
LOGOUT
MEMORY statistics
RELOAD
STATUS of process(es)
TALK to process
TELNET to IP-Address <this terminal type>
*
```

Although each of these commands has its purpose, you will use two of them far more than any of the others:

talk   Attaches your console to one of various processes, or ways of viewing the system

reload   Reboots the Network Utility (as we have seen already)

To use the **talk** command, type **t** *n*, where *n* (a *process id*) is usually one of the following:

6   To look at and modify the configuration (the *Config* process)

5   To look at current system status, actively control the state of the running system, and activate dynamic configuration changes (the *Console* process)

2   To look at a rolling log of informational and status messages (the *Monitor* process)

To undo the **talk** command and move from inside any process directly back to the * prompt, type **Ctrl-p**.

In the following three sections, we will visit each of the major processes and get familiar with some of the tasks you can perform inside each process. Along the way, we will get familiar with moving around between processes and menus, and entering commands.

Configuring (using talk 6, the Config process)

From the * prompt, type **t 6** to enter the command line process for configuring the Network Utility:

```
*        <enter>
*t 6
Gateway user configuration
Config>   <enter>
Config>
```

Now that you are inside the Config process, the command prompt has changed from * to `Config>`. Both the Config and Console processes have unique prompts so you can tell at a glance which process you are in. The status message "Gateway user configuration" only shows up the first time you enter the Config process following a reboot ("gateway" is used as a synonym for "router" in various places in the system).

When you have been in a given process before and re-enter it using the **talk** command, the system gives you a blank line instead of an immediate command prompt. Just press **Enter** and you are restored to where you were the last time you were inside that process:

```
Config>   <ctrl-p>          <---- leave Config and go back to *
*         <enter>
*t 6                        <---- go back into Config
          <enter>
Config>                     <---- we're back at the main Config prompt
```

When you are working inside the Config process, you are changing how the Network Utility is configured to operate. With a few exceptions, these changes have no effect on the running state of the router. In order to activate talk 6 changes you must either:

1. Issue one of several commands to activate a set of changes, or
2. Save the changes to the hard disk and reboot the system

As you follow this tutorial you will see examples of both methods.

Command Overview

At the main `Config>` prompt, type **?** to see an alphabetical list of the commands available to you:

```
Config>?
ADD (device, user)
BOOT and load file functions
CHANGE (device, password, user)
CLEAR configuration information
DELETE (interface, user)
DISABLE (interface, console-login, etc)
ENABLE (interface, console-login, etc)
EVENT logging system and messages
FEATURE (non-protocol and network features)
LIST (devices, configuration, patches, users)
LOAD (add, delete, list)
NETWORK interface configuration
PATCH global configuration parameters
PERFORMANCE monitor
PROTOCOL configuration
QCONFIG (quick configuration)
SET system-wide parameters
SYSTEM
TIME of day parameters
UNPATCH global configuration parameters
UPDATE
WRITE
Config>
```

Some of the above commands are for actually configuring the functions of the box, and others are for configuration management and system administration. To give you a feel for the types of things you do under talk 6, the following list groups key commands by user task:

- Configuring adapters and ports

add device	Configures a single adapter slot and port
change device	Moves or copies a slot configuration to another slot
delete interface	Deletes a single interface (adapter port) and associated protocol information
disable/enable interface	Controls whether a specific interface will be activated
list device	Shows all configured interfaces
net *interface number*	Goes to the subprocess to configure the specified interface, below the protocol level
set data-link	Changes a newly-added WAN adapter port from the default of PPP to Frame Relay, SDLC, SDLC Relay, or X.25
system set/display ip	Sets/shows the IP parameters for the PCMCIA LAN adapter

- Configuring protocols and features

protocol *name*	Goes to the subprocess to configure the specified protocol
feature *name*	Goes to the subprocess to configure the specified feature

- Managing configurations and software loads

boot	Goes to the subprocess to manage transfer and usage of configuration files and software loads on disk
clear	Can clean out all device, all protocol, or specific pieces of the current configuration in RAM
write	Saves the current configuration in RAM to the hard disk

- Configuring to monitor the box

event	Goes to the subprocess to configure which ELS messages are active
performance	Goes to the subprocess to configure CPU utilization monitoring

- Administering the system

add/change/delete/list user, change password
> Administer user id's for controlled console access

disable/enable console-login
> Control remote access to console

set host/prompt/contact/location
> Set up a host name, prompt prefix, contact person, or location

time	Sets the time and time format, or whether to get the time from a remote host

- Servicing the software

disable/enable dump, reboot
> Control dumping and rebooting if the box crashes

patch, unpatch	Control specialized software functions to get around problems in specific user environments
system retrieve	Sends a compressed system dump off the router to a server

Let's actually use some of these talk 6 commands to perform basic configuration tasks. As you work through the following examples, you will get experience not only with the tasks shown, but also generally with moving around through menus and issuing commands. We begin with a task that may already be familiar, if you used the command line procedure for initial configuration.

Example: configuring a port on an adapter

In the running example used throughout this tutorial, the user first booted a Network Utility with the following configuration:

- ESCON adapter in slot 1, IP not configured
- Token ring adapter in slot 2, port 2 configured with IP address 192.1.1.8

So we can start the example from scratch, we use **clear dev** to erase the device configuration, then use **add dev** and **del int** to get back to exactly the same device configuration.[1]

From the Config> prompt, type **list device** (or **li dev**, abbreviated) to see the adapters and ports that are defined in the current configuration. If you have no configuration, or no adapter ports defined, **li dev** gives no output but simply re-issues the user prompt. Since we have cleared all devices, let's add one. Type **add dev ?** to see a list of all the adapter types you can add:

```
Config>clear dev
You are about to clear all Device configuration information.
Are you sure you want to do this? ? [No]: yes
Device configuration cleared
Config>li dev
Config>add dev ?
ATM            1-port 155 Mbps ATM adapter
EIA-232E       8-port EIA-232E/V.24 adapter
ESCON Channel  1-port ESCON Channel adapter
ETHERNET       2-port Ethernet adapter
ETH100         1-port 10/100 Mb Ethernet adapter
FDDI           1-port FDDI adapter
HSSI           1-port HSSI adapter
PCA            1-port Parallel Channel adapter
TOKEN-RING     2-port Token Ring adapter
V35/V36        6-port V.35/V.36 adapter
X21            8-port X.21 adapter
Config>
```

We use the **add dev** command to configure a single port on a single adapter. For a multi-port adapter, you must specify which port you are adding to the configuration, and re-issue the command for each port you want to have active. Here we add a single-port ESCON adapter, and both ports of a 2-port token-ring adapter:

```
Config>add dev esc
Device Slot #(1-2) [1]? 1
Adding ESCON Channel device in slot 1  port 1 as interface #0
Use "net 0" to configure ESCON Channel parameters
Config>add dev tok
Device Slot #(1-2) [1]? 2
Device Port #(1-2) [1]? 1
Adding Token Ring device in slot 2  port 1 as interface #1
Use "net 1" to configure Token Ring parameters
Config>add dev tok
Device Slot #(1-2) [1]? 2
Device Port #(1-2) [2]? 2
Adding Token Ring device in slot 2  port 2 as interface #2
Use "net 2" to configure Token Ring parameters
Config>li dev
Ifc 0     ESCON Channel                      Slot: 1   Port: 1
Ifc 1     Token Ring                         Slot: 2   Port: 1
Ifc 2     Token Ring                         Slot: 2   Port: 2
Config>
```

---

[1] Normally, you use **clear dev** only in conjunction with **clear all**, which clears out protocol information.

To specify the adapter type, you type on the same line as **add dev** the first few characters of the words in the left-most column of the **add dev ?** output list (enough characters to distinguish the adapter type you want). When prompted, you must supply the slot and (for multi-port adapters only) the port number. Slot and port numbering is fixed as follows:

- The two adapter slots on a Network Utility are numbered 1 and 2, from left to right as you look at the front of the box

- Ports on multi-port LAN adapters are numbered 1 and 2 and are labelled on the adapter face

- Ports on multi-port WAN adapters are numbered starting with 0 and are labelled on the connectors at the end of the adapter cable

The **add dev** command makes sure you do not try to add two different adapters in the same slot, add an adapter to a slot that does not exist, or specify a port number that does not exist on a given adapter. It does *not* validate the device type you select against the adapters that are physically installed in your Network Utility. This allows you to configure adapters you have not yet installed, or produce a configuration for a different Network Utility. The box validates device configuration only when you boot up with a particular configuration. The system reports mismatches through the LEDs on the front of your adapter, which will not come up, as well as from a locally viewable event log. You can also type commands to see adapter status, as you will see later in this tutorial.

Logical Interface Numbers

In response to your **add dev** command, the Network Utility assigns a logical *interface number* or *net number* to the port you have just added. This is the key number by which you refer to this interface on every other command in the system. Only the **add dev** command uses physical slot and port numbers; all other commands use the logical interface number. When you subdivide a physical ("base") port such as ESCON or Frame Relay into multiple virtual interfaces, each virtual interface also has an interface number. As shown above, you can use the **li dev** command to see the interface number for every physical and virtual interface.

Example: deleting an Interface

If you make a mistake and want to undo the **add dev** command, or want to delete adapter/port configuration for any reason, use the **delete interface** command. (It is not named "delete device", because it deals with logical interface numbers and not adapter slot/port numbers.) To continue our example, let's say we decided we only wanted to use port 2 of the token-ring adapter. We delete port 1 (which happens to be interface 1) as follows:

```
Config>li dev
Ifc 0     ESCON Channel                        Slot: 1   Port: 1
Ifc 1     Token Ring                           Slot: 2   Port: 1
Ifc 2     Token Ring                           Slot: 2   Port: 2
Config>del int
Interface number? 1
Interface being deleted... please be patient.
The router must be restarted
Interface 1 deleted successfully
Config>li dev
Ifc 0     ESCON Channel                        Slot: 1   Port: 1
Ifc 1     Token Ring                           Slot: 2   Port: 2
Config>
```

Note that token ring port 2 has now become logical interface 1. If there had been
other interfaces with numbers higher than 1, these numbers also would have been
decremented by 1. If you want to delete every interface in a configuration, just
delete interface 0 repeatedly until there are no more interfaces.

In addition to device configuration itself, it is normal to have protocol configuration
that is associated with a particular interface. When you delete an interface using
the **del int** command, the system also deletes all protocol configuration associated
with that interface, and renumbers all protocol configuration associated with
renumbered interfaces.[2] You need to reboot Network Utility for a **del int** operation
to take effect in the running system.

Example: setting the host name, using menus

To look more closely at how to issue commands in general, let's try something
simple, like using the **set** command to set up a name ("host name") for this
Network Utility. First, try the command by itself:

```
Config>set
Command not fully specified
```

This error message reports that the **set** command is backed by a menu of
additional keywords, and you need to type more keywords until you form a
complete command that will perform an action. Anytime you are at a menu (as we
have seen already), you can type **?** to see the available commands or keywords to
type. If you are just trying to remember a command, it is usually much faster to
move around typing **?**, than it is to look up the command in a manual. In this case,
we see that our options are:

```
Config>set ?
CONTACT-PERSON
DATA-LINK
DOWN-NOTIFY
GLOBAL-BUFFERS
HOSTNAME
INACTIVITY-TIMER
INPUT-LOW-WATER
LOCATION
PACKET-SIZE
PROMPT
```

---

[2] The **clear dev** command does not perform this function, so you should use it only when you are also clearing protocol information
by hand.

```
RECEIVE-BUFFERS
SPARE-INTERFACES
```

As you can see, the **set** menu includes a mix of data items: some for system administration, some for node tuning, etc.. In Network Utility, node tuning options are defaulted for you and you should not have to change them.

Back to our task, the keyword we want is clearly "hostname". We can abbreviate any menu item (command name or keyword) to the number of characters needed to make it unique, so we choose to shorten "hostname" a bit:

```
Config>set host
Host name for this node []? rtp01
Host name updated successfully
rtp01 Config>
```

By default, the system inserts the new host name in front of all command prompts. Many users like this so they can telnet into a number of routers from a single work station and easily distinguish one router console from another. If you want to choose a different prompt prefix, you can use the **set prompt** command to do so. To reset either to a null value, use the **clear host** or **clear prompt** command and reboot the Network Utility. To look at the current values, use **list config**.

Note that **set host** is an exception to the normal talk 6 rule in that it took effect immediately and did not require you to issue some sort of "activate" command, or to reboot the Network Utility. Very few talk 6 commands behave this way, but this one is useful so you can immediately see its effect on the user prompt.

Example: typing ahead

Suppose you don't like the new prompt and want to change the host name from "rtp01" to "RTP01". You can do this in a single command, as follows:

```
rtp01 Config>set host RTP01
Host name updated successfully
RTP01 Config>
```

The system did not prompt you for the host name because you typed it on the original command line. This illustrates another general rule: when a full command prompts you for input parameters, you have the option of typing them on the original command line and skipping the prompts. The system parses the command line and passes the first parameter to the first prompt, the second parameter to the second prompt, and so on. If you choose to skip prompts, you need to be careful to type parameters in the right order.

Example: setting a port parameter using "net"

Now that we've configured our host name, let's try something a bit more complex. Suppose you noticed when you rebooted from Config-only mode, that your newly configured token-ring adapter port 2 did not come up. Let's see what ring speed it is configured for, and change that value. This sort of low-level device-specific configuration parameter is what you use the **net** command for, as shown here:

```
RTP01 Config>li dev             <------ what were those i/f numbers again?
Ifc 0     ESCON Channel                       Slot: 1   Port: 1
Ifc 1     Token Ring                          Slot: 2   Port: 2
RTP01 Config>         <enter>
RTP01 Config>net 1              <------ let's configure interface 1
Token-Ring interface configuration
RTP01 TKR config>   <enter>   <------ note the new subprocess prompt
RTP01 TKR config>?             <------ what are the commands here?
EXIT
FRAME
LIST
LLC
MEDIA
SET
PACKET-SIZE bytes
SOURCE-ROUTING
SPEED Mb/sec
RTP01 TKR config>li            <------ show me what we have now
Token-Ring configuration:

Packet size (INFO field): 2052
Speed:                  4 Mb/sec        <----- aha! we want 16Mb/sec
Media:                  Shielded

RIF Aging Timer:        120
Source Routing:         Enabled
MAC Address:            000000000000
RTP01 TKR config>speed
Speed (4 or 16) [4]? 16        <------ we change the speed here
RTP01 TKR config>li            <------ let's verify the new value
Token-Ring configuration:

Packet size (INFO field): 2052
Speed:                  16 Mb/sec       <----- looks good now!
Media:                  Shielded

RIF Aging Timer:        120
Source Routing:         Enabled
MAC Address:            000000000000
RTP01 TKR config>ex            <------ exit the subprocess
RTP01 Config>                  <------ we're back at the main T 6 menu
```

This change to the ring speed does not take effect immediately, but requires a talk 5 command or reboot to activate it. In "Dynamic Reconfiguration" on page 6-5, we cover the basics of activating configuration changes without a reboot. Normally, you use the **net** command immediately after **add dev**, to look at the default settings for the new interface and make any necessary changes before activating the port for the first time.

In this example, when you typed **net 1**, you moved into a subprocess for configuring token-ring interfaces. The base menu changed and the prompt also changed to let you know you were no longer at the main Config> menu, but one level deeper. In order to leave any subprocess menu and return to the next higher one, type **exit**. Remember also that **Ctrl-p** immediately takes you all the way out to the * prompt, and when you return to that process you re-enter where you were last:

```
RTP01 Config>          <enter>        <------ start here
RTP01 Config>net 1                    <------ enter a Config subprocess
Token-Ring interface configuration
RTP01 TKR config>    <ctrl-p>         <------ jump out
RTP01 *              <enter>
RTP01 *t 6                            <------ go back to Config
                     <enter>
RTP01 TKR config>    <enter>          <------ you're back to the subprocess
RTP01 TKR config>ex                   <------ exit the subprocess
RTP01 Config>                         <------ we're back where we started
```

Let's try two more examples in the Config process, and then move on to the
Console process. The first example shows how to reduce the time to reload the
box, and the second shows how to change parameters associated with a box
protocol.

Example: enabling "fast-boot"

From the Config> prompt, type **boot** to reach the subsystem for managing
configurations, code loads, and boot options. Chapter 7, "Handling Configuration
Files" gives the full background on this subsystem, so we will not look at all the
commands here. Look under the **enable** command and try out the "fastboot"
option:

```
RTP01 Config>boot                        <----- enter subprocess
Boot configuration
RTP01 Boot config>   <enter>             <----- note new prompt
RTP01 Boot config>en ?                   <----- list "enable" options
AUTO-BOOT-- set Unattended mode
FAST-BOOT-- bypass diags
RTP01 Boot config>en fast                <----- try out "fast-boot"
FastBoot mode is now enabled.

Operation completed successfully.
RTP01 Boot config>ex                     <----- exit the boot subprocess
RTP01 Config>
```

If you watched the console bootup messages when you powered on your Network
Utility or typed the **reload** command, you may have noticed that the system runs
through a number of power-on diagnostics when it is booting. While this is
desirable for a production router that is rebooted infrequently and whose hardware
should be validated, it does lengthen the boot time. If you are actively configuring
and repeatedly rebooting a given router, you may wish to reduce the boot time by
skipping these diagnostics. You have just done this with the **enable fast-boot**
command. The next time you do a **reload**, it will proceed more quickly. You
should undo this change using **disable fast-boot** before placing the Network Utility
into production.

Note that the fast-boot mode can only be controlled by the command line and not
from the Configuration Program. The system's boot mode is stored in non-volatile
memory on the box, and is not part of the configuration file.

Example: changing an interface IP address

For our final Config process example, let's use the menus and commands of the IP protocol subprocess to change an interface IP address. As noted on page 5-4, this example started with a Network Utility that had an IP address configured on Interface 1 (port 2 on the token-ring adapter in slot 2).

```
RTP01 Config>li dev              <----- what are the intfcs again?
Ifc 0      ESCON Channel                      Slot: 1   Port: 1
Ifc 1      Token Ring                         Slot: 2   Port: 2
RTP01 Config>p ip                <----- short for "protocol ip"
Internet protocol user configuration
RTP01 IP config>   <enter>       <----- now in IP Config subprocess
RTP01 IP config>li addr          <----- list configured IP addresses
IP addresses for each interface:
   intf     0                                 IP disabled on this interface

   intf     1   192.1.1.8       255.255.255.0    Local wire broadcast, fill 1
RTP01 IP config>change addr
Enter the address to be changed []? 192.1.1.8
New address [192.1.1.8]? 192.7.7.7
Address mask [255.255.255.0]?   <enter>
RTP01 IP config>li addr          <----- verify the change
IP addresses for each interface:
   intf     0                                 IP disabled on this interface

   intf     1   192.7.7.7       255.255.255.0    Local wire broadcast, fill 1
RTP01 IP config>ex               <----- exit IP config
RTP01 Config>
```

This is our first example of using the **protocol** command to enter the subprocess for an individual protocol. IP is just one of many protocols we could have selected, and there is a similar list of features you can access using the **feature** command. Type **list config** from Config> for a full list of the protocols and features you can configure, or just **p ?** or **f ?** for a quick reminder. All protocols and features work the same way: you enter the subprocess for a protocol or feature, configure it using commands specific to that protocol or feature, then **exit** to the main Config> prompt.

For detailed command reference material on configuring any given protocol, see that protocol's chapter in one of the two volumes of the *MAS Protocol Configuration and Monitoring Reference*. Each of these chapters provides introductory material about the protocol, and a description of each configuration and monitoring console command for that protocol.

We have now completed the overview of the Config process and its commands. Let's move on to talk 5, the Console process. Remember, to leave any process you type **Ctrl-p** to reach the ∗ prompt, then you are ready to use the **talk** command to enter another process:

```
RTP01 Config>   <ctrl-p>
RTP01 *
```

Operating (using talk 5, the Console process)

From the * prompt, type **t 5** to enter the command line process for monitoring and controlling the Network Utility's active state:

```
RTP01 *   <enter>
RTP01 *t 5


CGW Operator Console

RTP01 +   <enter>
RTP01 +
```

Now that you are inside the Console process, the command prompt has changed from * to +. The Config and Console processes and their subprocesses have unique prompts so you can tell your position at a glance. The status message "CGW Operator Console" only shows up the first time you enter the Console process following a reboot. As we discussed with talk 6, if the system gives you a blank line when you type **t 5**, that means you have been in talk 5 before and need to press **Enter** to resume wherever you were last.

When you are working inside the Console process, you type commands to view and modify the active running state of the Network Utility. You cannot modify the Network Utility's configuration files from this process. Some talk 5 commands allow you to dynamically modify configuration parameters, but these changes are lost when you reboot the Network Utility. If you have made configuration changes under talk 6, however, you can dynamically activate some of them from talk 5 without rebooting the Network Utility.

Command Overview

At the main + prompt, type **?** to see an alphabetical list of the commands available to you:

```
RTP01 +?
ACTIVATE interface
BUFFER statistics
CLEAR statistics
CONFIGURATION of router
DISABLE interface or slot
ENABLE slot
ERROR counts
EVENT logging
FEATURE commands
INTERFACE statistics
MEMORY statistics
NETWORK commands
PERFORMANCE monitor
PROTOCOL commands
QUEUE lengths
RESET interface
STATISTICS of network
TEST network
UPTIME
RTP01 +
```

Some of the above commands are for viewing the status of the box, and some are operator commands for actively changing that status. In addition, under each protocol and feature there is a Console subprocess containing a mixture of these two command types. The following list groups key talk 5 commands by type:

- Viewing box status

buffer	Shows interface buffer allocation and in-use counts
configuration	Shows software identity, protocols/features, and interface status
error	Shows frame error counts for one or more interfaces
interface	Shows the interface number to slot/port mapping (the talk 5 equivalent of talk 6 **list dev**), plus self-test pass/fail counts
memory	Shows installed memory and in-use statistics for memory and global (non-interface) buffers
queue	Shows input and output buffer queue counts for one or more interfaces
statistics	Shows packet and byte counts for one or more interfaces
uptime	Shows elapsed time since the last reboot

- Controlling box status

activate	Enables a spare interface that you just configured under talk 6
clear	Resets counters for one or more interfaces
disable	Takes offline either a single interface, or all the interfaces in a slot
enable	Brings online all the interfaces in a specified slot
reset	Disables an interface and re-enables it using new configuration parameters you changed under talk 6
test	Verifies and brings a single interface online

- Accessing other console subprocesses

event	Go view counts and temporarily change which ELS messages are being logged
feature *name*	Go view and change status for the specified feature
network *interface number*	
	Go view and change status for the specified interface
performance	Go view CPU statistics and temporarily change how they are being collected and displayed
protocol *name*	Go view and change status for the specified protocol

Example: viewing box status

As we did from talk 6, let's try some of these talk 5 commands. Those for viewing box status are all quite simple; you simply type the one-word command and look at the output:

```
RTP01 +mem
Physical installed memory:        256 MB
Total routing (heap) memory:      228 MB
Routing memory in use:              3 %

                     Total   Reserve    Never      Perm     Temp     Prev
                                        Alloc     Alloc    Alloc    Alloc
Heap memory       239390720    26616 232309212  7029792    49828     1888

Number of global buffers: Total = 1000, Free = 1000, Fair = 194, Low = 200
Global buff size: Data = 4478, Hdr = 82, Wrap = 72, Trail = 7, Total = 4644
RTP01 +    <enter>
RTP01 +buff
                 Input Buffers        Buffer sizes                   Bytes
Net    Interface  Req Alloc Low Curr   Hdr  Wrap  Data Trail Total    Alloc
0      ESCON/0    255  255  20    0    86    72  4478     0  4636  1182180
1      TKR/0      250  250   7    0    85    72  2052     7  2216   554000
```

As you can see, **mem** shows box-level status, while **buff** gives interface-level information. For all the commands that give per-interface information (**buff**, **config**, **error**, **int**, **queue**, **stat**), you can specify a list or range of interface numbers you are interested in:

```
RTP01 +int 0-1
                                         Self-Test  Self-Test Maintenance
Net   Net'   Interface  Slot-Port          Passed     Failed      Failed
0     0      ESCON/0    Slot: 1   Port: 1       0          0           0
1     1      TKR/0      Slot: 2   Port: 2       0          0           0
RTP01 +stat 1
Net    Interface     Unicast  Multicast    Bytes   Packets    Bytes
                     Pkts Rcv  Pkts Rcv  Received    Trans    Trans
1      TKR/0               0         0         0        0        0
```

Clearly, the output of these various status commands needs some explaining. See the *MAS Software Users Guide* chapter "The Operating/Monitoring Process" for a description of the fields in each command's output.

Example: viewing interface status

The **config** command is particularly important, because at the end of the output is the status of all specified interfaces (this example output is edited to remove blank lines):

```
RTP01 +c
Multiprotocol Access Services
NetU-TX1 Feature 1001 V3.1 Mod 0 PTF 1 RPQ 0 MAS.DE1 netu_38PB

Num Name   Protocol
0   IP     DOD-IP
3   ARP    Address Resolution
11  SNMP   Simple Network Management Protocol
29  NHRP   Next Hop Resolution Protocol

Num Name   Feature
2   MCF    MAC Filtering
7   CMPRS  Data Compression Subsystem
8   NDR    Network Dispatching Router
10  AUTH   Authentication
```

```
2 Total Networks:
Net  Interface   MAC/Data-Link          Hardware                       State
0    ESCON/0     ESCON                  ESCON Channel                  Not present

1    TKR/0       Token-Ring/802.5       Token-Ring                     HW Mismatch
RTP01 +
```

The Network Utility from which this example output was captured in fact has an empty slot 1 and an Ethernet adapter in slot 2. In talk 6, it doesn't matter if what you configure does not match the installed adapters, but when you reboot with that configuration (as we have done), talk 5 will show you that your configured interfaces have not come up.

If we had configured correctly, the interface state would start with "Testing" then move to "Up", and we would be able to use the **net** command to enter an adapter-specific Console subprocess to get more detailed status information. As it is now, we get the following:

```
RTP01 +net 0
 Network interface is not available.
RTP01 +
```

Example: accessing an unconfigured protocol

To view and control what's currently going on with any given protocol, you use the **protocol** command to enter the Console subprocess for that protocol. As we saw previously, **p ?** will generate a quick list of the protocols supported in a given software load. Let's just pick Data Link Switching (DLSw) at random to try:

```
RTP01 +p dls                        <----- short for "protocol dlsw"
 Protocol DLSW is available but not configured
RTP01 +
```

DLSw is *available* because it is supported by this software load,[3] but it is *not configured* because we never went into talk 6 and entered the commands to enable DLSw. Now that we have booted the box without DLSw in the configuration, it is not running and there is no DLSw status to view or modify from talk 5.

Example: accessing a configured protocol

As noted on page 5-4, this example started on a Network Utility already booted with an IP configuration. IP is therefore actively running, so we can enter its Console subprocess and see what commands are available:

```
RTP01 +p ip                         <----- short for "protocol ip"
RTP01 IP>?
ACCESS controls
CACHE
COUNTERS
DUMP routing tables
INTERFACE addresses
PACKET-FILTER summary
PARAMETERS
PING dest_addr [src_addr size ttl rate]
```

---

[3] If it had not been supported, it would not have shown up under **p ?**, and the system would not have recognized the value "dls".

```
REDUNDANT Default Gateways
RESET
RIP
ROUTE given address
ROUTE-TABLE-FILTERING
SIZES
STATIC routes
TRACEROUTE dest_addr [src_addr size probes wait ttl]
UDP-FORWARDING
VRID
VRRP
EXIT
RTP01 IP>
```

If you compare the above command list to that generated in talk 6 by typing **?** at the **IP config>** prompt, you see that the talk 5 and talk 6 commands are quite different. In talk 5, for example, you can initiate a **ping** to see if you can reach a given IP address from the Network Utility. Since this is an active command immediately operating on an active network interface, it does not belong in talk 6. Other commands to view active status likewise are talk 5 commands and not talk 6 commands.

Example: dynamic reconfiguration

You may remember that while we were in talk 6, we changed the IP address of token-ring port 2 from 192.1.1.8 to 192.7.7.7. Let's see what value appears under talk 5:

```
RTP01 IP>int                          <----- short for "interface"
Interface  IP Address(es)   Mask(s)
  TKR/0     192.1.1.8        255.255.255.0
```

Our talk 6 change had no effect on the operational state of the Network Utility, because we have not yet activated it either by explicit command or by rebooting. Use the command **reset ip** to re-read the current talk 6 IP configuration and dynamically activate it in the running system:

```
RTP01 IP>reset ip
RTP01 IP>int
Interface  IP Address(es)   Mask(s)
  TKR/0     192.7.7.7        255.255.255.0
RTP01 IP>ex
RTP01 +
```

As you can see, our IP address change (and any other IP changes we might have made under talk 6) is now active. Most protocols have some mechanism for dynamic reconfiguration, but not every protocol has a **reset** command under talk 5. See "Dynamic Reconfiguration" on page 6-5 for more background on ways to do dynamic reconfiguration.

We have now seen how to issue talk 5 commands to actively query the status of the system. There is another, more passive mechanism available: viewing event messages that the Network Utility generates. To do this you use **talk 2**. As always, type **Ctrl-p** to leave the current process:

```
RTP01 +   <ctrl-p>
RTP01 *
```

Event Logging (using talk 2, the Monitor process)

From the * prompt, type **t 2** to attach your console to the process for viewing the Network Utility's local message log:

```
RTP01 *   <enter>
RTP01 *t 2
00:00:50   GW.001:

Copyright 1984 Massachusetts Institute of Technology,
Copyright 1989 The Regents of the University of California


00:00:50   GW.002: Portable CGW RTP01 Rel NetU-TX1 Feature 1001 V3.1 Mod 0 PTF 1
 RPQ 0 MAS.DE1 netu_38PB
 strtd
00:00:50   GW.005: Bffrs: 1000 avail 1000 idle   fair 194 low 200
00:00:50   DOLOG: .....Remote Logging Facility is now available.....
```

In this example, only four messages have been logged since the Network Utility was last booted. Each message has the format:

- Time stamp in the format HH:MM:SS

  All 4 of the above messages were logged in the same second, 50 seconds after the clock started.
- Message id in the format SUBSYSTEM.ID

  GW.001, GW.002, and GW.005 are ELS messages in the GW (GateWay) subsystem. DOLOG is a non-standard, unconditional type of message you will see from time to time.
- Message body

  The body of GW.001 is the two Copyright statements. The body of GW.002 is the software version statement. To look up the meaning of any particular ELS message, see the *Event Logging System Messages Guide* either on the web or in CD-ROM format.

Unlike the talk 6 and talk 5 processes, the talk 2 process has no user command prompt. That is because you do not type commands when you are inside talk 2; you simply watch messages roll by as the Network Utility generates them. You control what messages appear by enabling or disabling individual or groups of messages under the **event** subprocess of either talk 6 or talk 5. See "Monitoring Event Messages" on page 8-2 for an introduction to ELS concepts and controlling ELS messages.

Under talk 2, then, the only thing you would normally type is **Ctrl-p**, to return to the * and move to talk 5 or talk 6. If messages are scrolling by too quickly to read, you can use **Ctrl-s** to pause scrolling, and **Ctrl-q** to resume it. Other options for capturing fast-moving event messages include:

- Activating a log file from within a PC terminal emulation program that you are using for your console

- From a UNIX or AIX work station, telnetting into the Network Utility to get your console connection, and *tee*ing the telnet session into a local work station file.

- Using the Network Utility's capability to log ELS messages over the network to a remote host, rather than to the local talk 2 process.

These options are described in detail in the *MAS Software Users Guide* chapter "Using the Event Logging System (ELS)".

When you enter talk 2, the system displays all the messages that have been buffered up since the last time you entered talk 2. If the message buffer has been overrun or the system is currently generating messages faster than it can display them, you will see lines about "messages flushed" interspersed within the talk 2 scrolling output.

If you are about to enter talk 2 and you know that there is a backlog of old messages to be displayed before you can see the current messages you are interested in, just use the command **flush 2** from the * prompt before typing **talk 2**. The system discards the entire backlog and talk 2 displays only messages generated after you entered the **flush** command.

Now that you've had a look at a few ELS messages, type **Ctrl-p** to exit talk 2 and return to the * prompt.

Saving the Configuration and Rebooting

If you followed through the examples in this tutorial, you have made the following talk 6 configuration changes since you began:

- Added two interfaces
- Set the host name
- Changed an interface token-ring speed
- Changed an interface IP address

Note: You also enabled the "fast-boot" option, but this change is stored in NVRAM and is not relevant here.

On a Network Utility, talk 6 changes are actually made in a RAM copy of the configuration. If you want these changes to become permanent and be used with the next reboot of the Network Utility, you need to write them to the hard disk. Two different command sequences can accomplish this task:

```
RTP01 *t 6
                  <enter>
RTP01 Config>write
Config Save: Using bank A and config number 3

<boot messages start to appear>

RTP01 Config>   <ctrl-p>
RTP01 *reload
Are you sure you want to reload the gateway? (Yes or [No]): yes

<boot messages start to appear>

 ..... or .....

RTP01 *reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? (Yes or [No] or Abort): yes
Config Save: Using bank A and config number 3

<boot messages start to appear>
```

In the first sequence, the user knows that he/she has made talk 6 changes, and uses the `Config>`**write** command to commit those changes to disk before **reload**. In the second sequence, the user counts on the system to remember whether or not he/she has changed the configuration. If the user has not made any changes, the **reload** process will take off following the first "yes" confirmation. If the user has made configuration changes, the system asks whether to save those changes to disk before proceeding with the **reload**.

Which method you use is completely up to you. Many users prefer the second method because it involves less thinking and typing, but it may also be easier to forget what talk 6 changes you have made if you do not issue a **write** shortly after making them.

Firmware

Up until now, we have always booted the Network Utility all the way up to the operational software, at either the `Config (only)>` or `*` prompts. There is one other major console user interface that we have not yet visited, that of the *firmware*. You may not need to interact much with the firmware, but you should know about it for two reasons:

1. It provides an alternate way to load code and configuration files onto the hard disk, and may give you a way out of a difficult problem.

2. Occasionally, IBM ships PTFs that require an upgrade to the firmware. New MAS releases also require a firmware upgrade. The only way to upgrade the firmware to a new level is by using the current firmware.

Network Utility firmware is low-level software that drives the power-on and boot logic of the system. It resides in Flash memory rather than on the hard disk, so in the event of a failure such as corruption of your operational software load on disk, you can retrieve new software or configuration files and get back up and running.

To reach the firmware user interface, do a **reload** from the `*` prompt and look for these messages:

```
Starting Boot Sequence...
Strike F1 key now to prematurely terminate Boot
```

Look closely because these only appear for a few seconds each. Press **F1** when prompted, or **Ctrl-c** at either message to interrupt the normal boot sequence and move into the firmware.

After you interrupt the boot sequence, the system may prompt you for a supervisory password before you can see the firmware main menu. This password controls access to sensitive low-level firmware functions. Its initial value from the factory is "2216". You can change it only from the firmware itself, under the Utilities menu.

If you telnet in or dial into the Network Utility via modem to get your console and lose your connection on **reload**, you may not be able to connect back in time to press **F1**. In this case, go to the **boot** subsystem of the Config process and issue the **disable auto-boot** command:

```
*t 6
Gateway user configuration
Config>boot
Boot configuration
Boot config>dis auto                 <----- short for "disable auto-boot"
AutoBoot mode is now disabled.

Operation completed successfully.
Boot config>    <ctrl-p>
*rel y                                <----- short for "reload", "yes"

<boot messages appear>
```

With AutoBoot mode disabled, the system will stop the **reload** process at the firmware, without your having to press **F1**. Then when you connect back in, you will be at the main menu or the request for the supervisory password.

If you disable auto-boot in talk 6 to reach the firmware, remember to re-enable it when you reach the operational code, or you will stop in the firmware for every **reload**.

When you reach the firmware, you see the following main text menu at your user console:

```
Nways System Firmware
Version 3.00 built on 04/21/98 at 22:18:42 in cc3:paws_netu6e:cc3_6e
(C)Copyright IBM Corporation, 1996, 1998.  All rights reserved.
                              System Management Services


 Select one:
  1. Manage Configuration
  2. Boot Sequence Selection
  3. Select Device to Test
  4. Utilities




     Enter   -   Esc=Quit   -    F1=Help   -    F3=Reboot  -   F9=Start OS
   -------------   -------------   -------------   -------------   ------------
```

The firmware menu structure and its options are described in the *2216 and Network Utility Service and Maintenance Manual* in the chapter "Using 2216 Firmware". You don't type any commands, just move through a sequence of menus by selecting options. The key tasks you may need to perform from the firmware are:

- Transfer configuration files and operational software to disk

  These functions are equivalent to the **boot** subsystem functions under talk 6.
  You find them in the firmware menus under "Utilities", then "Change Management".

- Upgrade the firmware itself

  To do this, start with "Utilities" on the main menu, then "Update System Firmware".

You may want to move around the menus a little to get familiar with them. When you are done with any firmware task, press **Esc** to return back out to the main menu. Use one of the following options to go on:

F3=Reboot - starts the boot process all over. If you have auto-boot disabled, you will just stop in the firmware again. If you are dialed in, you will lose your connection again.

F9=Start OS - continues the boot process past the firmware up into the operational code.

You have reached the end of this Network Utility user interface tutorial. In the following chapters, we will cover a number of other important Network Utility concepts and methods, and assume you have the background provided in this chapter.

Chapter 6. Configuration Concepts and Methods

This chapter provides background information about configuring Network Utility, including:

- What it means to configure Network Utility
- The different ways configuration information is stored and transferred
- The different methods available for creating and changing configurations

Chapter 3, "Performing the Initial Configuration" introduced the basic methods of configuring Network Utility and provided guidance on choosing between them (see "Choosing Your Configuration Method" on page 3-1). This chapter gives additional details about each method and discusses using both of them together.

For specific procedures and commands dealing with configuration files, see Chapter 7, "Handling Configuration Files" on page 7-1. Some common configuration tasks are described in Chapter 4, "Quick Reference to the User Interface" on page 4-1.

Configuration Basics

A Network Utility configuration is a collection of data items that control how the software operates, including such elements as:

- What interfaces to activate
- What links to bring up
- What protocols and features to make active
- What functions in a given protocol or feature to make active
- What network addresses and names to use

When you boot up a Network Utility, the system reads its configuration information from a file on the hard disk, and activates interfaces and protocols according to the information in that file. You create the file in one of two ways:

- Using the command line interface from a user terminal console

  You type commands to create configuration data items in memory, then write the configuration to the Network Utility hard disk.

- Using a graphical configuration program that runs on a PC or workstation

  You create the configuration on the workstation, then transfer it to the Network Utility hard disk.

Once the system is up and running, you can use the command line interface to make the following types of configuration changes:

- Changes that take effect in the running system, but are not saved in a file and are therefore lost when you reboot
- Changes that take effect in the running system, are also saved in a file and are therefore maintained when you reboot
- Changes that do not take effect in the running system, but are saved in a file and become active only when you reboot

6-1

Configuration Files

The Network Utility hard disk is organized to contain two logical *banks*, one for each of two operational code (software) loads. This allows you to have the active code load in one bank, transfer a new load to the other bank, test it, and be able to back off to the original load if necessary. The two banks are referred to as Bank A and Bank B.

Each of the two banks has room for 4 configuration files. You can select to boot up the code load in Bank A with any of the 4 configuration files in Bank A. The same holds true for Bank B. To use a Bank A configuration file with the Bank B code load, you must first copy the Bank A configuration file to one of the 4 file positions in Bank B.

There are four ways to transfer a configuration file into a bank on the hard disk:

1. Use the talk 6 command **write** to store the current configuration in RAM out as a disk file

   Use this command if you are configuring your Network Utility with the command line talk 6 process, rather than with the Configuration Program.

   Note: If the term "talk 6" is unfamiliar to you, work through Chapter 5, "A Guided Tour through the Command-Line Interface" as a tutorial on the command line interface.

2. Use TFTP or XMODEM to transfer the configuration file from a local server (PC or workstation) directly onto the hard disk

   You can transfer a configuration file in, whether the file was created from the Configuration Program or was previously transferred from this or another Network Utility.

3. Use SNMP to transfer configuration data from the Configuration Program into RAM, and then onto the hard disk

   You initiate the file transfer from the Configuration Program. This method is available only from the Configuration Program.

4. Copy a configuration file from one bank to the other

   You initiate copies and other configuration file management operations from the Network Utility console under talk 6 in the **boot** subprocess.

See the section "Loading New Configuration Files" on page 7-4 in Chapter 7, "Handling Configuration Files" for specifics on these operations.

Configuration Methods

Command Line Interface

To use the command line interface, you must first bring up a local or remote console to a Network Utility. For details on how to do this and reach the * or `Config (only)>` prompts, see Chapter 2, "Bringing Up a User Console."

If you have an active console at the * prompt, you use **talk 6** to access the Config process. If you are at the `Config (only)>`, the Config process is the only process available to you. From the Config process, you navigate menus and issue

commands to configure interfaces and protocols, and write these changes to configuration files on the Network Utility hard disk.

In most cases, you use the command line interface to configure the very Network Utility to which you are attached. But you could easily use a single Network Utility to produce configuration files to be transferred into other Network Utilities. Simply use the **write** command under talk 6 to store a configuration to a disk file, then use **tftp put** under the boot subprocess to transfer the file off the Network Utility. From then on, you have a file that can be loaded into the target Network Utility just as if it had come from the Configuration Program.

One option available only from the command line is Quick Config. As described in step 2 on page 3-3, Quick Config guides you through an initial configuration of a subset of the protocols in Network Utility. The system asks you questions, instead of the normal mode where it waits for you to type commands.

The ability to dynamically activate configuration changes without rebooting the Network Utility is also exclusive to the command line interface. In "Example: dynamic reconfiguration" on page 5-16, we saw one example of using talk 5 to activate an IP address change made under talk 6. "Dynamic Reconfiguration" on page 6-5 gives more background on the dynamic reconfiguration capabilities of Network Utility.

Configuration Program

Network Utility is supported by the same graphical configuration program that you can use to configure the 2216-400. You run this program on a PC or workstation and send the configurations you produce to one or more 2216s or Network Utilities. A version of the 2216/Network Utility Configuration Program is available for each of the following operating systems:

- Microsoft Windows 95 or Windows NT

- IBM AIX

- IBM OS/2

IBM distributes major releases of the Configuration Program on CD-ROM and on the Web. Regular maintenance PTFs are available only on the Web. The publication *Configuration Program User's Guide* describes system requirements and contains instructions for installing and using the program.

Support for Network Utility and 2216-400

When you start a new configuration with the Configuration Program, it presents a drop-down list for you to select whether the new configuration is for a 2216-400 or for a Network Utility. Your choice affects the following:

- The number of adapter slots you can configure
- The types of adapters you can configure (Network Utility supports a subset of the full list of 2216 adapters)
- The protocols and features you can configure (Network Utility supports a subset of the full MAS function)
- The default value for a variety of tuning parameters (Network Utility is pre-set for its intended applications).

Configurations for the 2216-400 and Network Utility are not interchangeable.

Configuration File Formats

The Configuration Program deals with three different formats of configuration files:

- .CSF files: contain data in a format native to the Configuration Program

  You use this format with the *Configure* pull-down commands **Open**, **Save**, **Save as**, and **Delete**. Contents are software release-independent; the Configuration Program automatically migrates data items when you do an **Open**.

- .CFG files: contain data in a format native to the router

  You use this format when you want to create a file to transfer to the router, or when you want to read in a file you have transferred from a router.

- .ACF files: contain data in an ASCII flat file format

  You can write your configuration out into an ASCII flat file, make changes to it with a text editor, and read it back in.

Transferring and Activating Configurations

There are two ways to transfer a configuration from the Configuration Program to a Network Utility:

1. Create a router-format (.cfg) file, transfer it (using FTP, perhaps) to a server near the Network Utility, then retrieve it with XMODEM or TFTP onto the Network Utility's hard disk. The configuration becomes active when you select it and reboot the Network Utility.

2. Initiate a Configuration Program "send" operation. The Configuration Program uses SNMP to send individual data items (not a true file) into the Network Utility. The Network Utility clears the active memory copy of its current configuration, receives these data items, then writes them to disk in a new file. Before you do the "send", you select at the Configuration Program whether the Network Utility should be rebooted with the new configuration, and if so when. The configuration you sent becomes active only upon reboot.

Note that with each method, you transfer and activate an entire Network Utility configuration. There is no mechanism for the Configuration Program to dynamically send a small configuration change and activate it at the Network Utility without requiring a reboot of the Network Utility. You can only perform this type of dynamic reconfiguration using the command line interface.

Other Configuration Program Features

Following are some of the features of the Configuration Program, a few of which we have touched on in the description above:

- Timed Restart

  When you use the Configuration Program's facility to send a configuration to a router, you can specify the date and time you want the router to restart and use the configuration.

- Multiple Router Send

  You can create a list of target routers to receive configuration files, with the same or different configuration files, restart times, etc., for each router.

- Command Line Facility

You can use the workstation operating system command line, from which you start the Configuration Program, to automate configuration operations that are available in the program. You place arguments on the original command line or in an argument file, and the Configuration Program uses them to direct its operation.

From AIX, it is not necessary to have the operating system graphical environment (e.g., Xwindows) installed to use this facility. You start the Configuration Program using the **headless** command.

- ASCII file support

  You can use the Configuration Program to create and read configuration files in ASCII format. You can also convert configuration files from one format to another. An ASCII configuration file may be useful if you need to alter many configurations at the one time without having to load configurations into the graphical user interface. This feature is not intended to be used to create new configurations or to make major modifications to existing configurations.

- On-line Help

  The Configuration Program supports an extensive set of help files. Press **F1** when you are positioned on any data item, and you will see a pop-up window describing the item and giving its default value and allowable range.

Dynamic Reconfiguration

As mentioned earlier, the ability to dynamically modify configuration parameters without rebooting the Network Utility is available only from the command line interface. Table 6-1 summarizes the different ways you can change configuration parameters from the command line, whether a change affects the running system before a reboot, and whether the change is active following a reboot. The column "Choose Write to Disk?" indicates whether one has issued the **write** command from the main talk 6 menu to save the configuration to disk, or has requested a disk save after issuing the **reload** command.

Table 6-1. Dynamic Reconfiguration Options			
Method	**Choose Write to Disk?**	**Affects Running System**	**Active After Reboot**
Change in talk 6	Yes	No (1)	Yes
	No	No (1)	No
Change in talk 5	Not applicable	Yes	No
Change in talk 6, then activate in talk 5 (3)	Yes	Yes (2)	Yes
	No	Yes (2)	No

Notes:

1. The Network Dispatcher feature is an exception to this rule; its talk 6 changes take effect immediately.

2. The change takes effect when you do the activate command, not when you change the parameter (unlike a direct talk 5 change).

3. The APPN protocol is an exception to this rule; you activate its talk 6 changes from talk 6 instead of talk 5.

As you can see, the general rule is that talk 6 changes become active following reboot or a talk 5 command to activate them. Talk 5 commands become active immediately but are lost upon reboot.

Not every configuration data item can be changed in all of the above ways. It depends on the part of the system (protocol, interface, etc.) to which a given data item belongs. For example, DLSw, SNMP, and ELS configuration all support most of the same commands in talk 6 and talk 5. You can make a change in either place depending on the permanence you want for the change. There is no talk 5 command to activate talk 6 changes, because a talk 5 command exists to make the same change.

In IP, however, there are no talk 5 commands corresponding to talk 6 commands. You use **reset ip** in talk 5 to activate the current talk 6 configuration. Interface reconfiguration is also activated using a single talk 5 command, because it involves taking the interface down and up.

See "Configuring Physical Adapters and Interfaces" on page 4-5 for a few examples of common dynamic reconfiguration tasks involving adapters and interfaces.

Combining Configuration Methods

If you decide to use only the command line interface for configuration, you never need to use the Configuration Program. If you use the Configuration Program, you still need to use the command line Config process for several reasons:

- For some protocols, talk 6 is the only way to view the Configuration Program configuration on an active Network Utility

- There are a few configuration items, such as ELS messages and the PCMCIA EtherJet addresses, that are only accessible by talk 6 and not from the Configuration Program

- The command line is the only way to make dynamic configuration changes.

To use a combination of the Configuration Program and talk 6, you must keep the ".csf" file at the Configuration Program synchronized with the configuration information at the Network Utility. A typical scenario might be:

- Do the initial configuration at the Configuration Program

- Transfer this configuration to the Network Utility, either using SNMP, or by a creating a .cfg file and transferring it manually

- Activate, debug, and tune the configuration at the Network Utility using the command line interface

- Retrieve the configuration back into the Configuration Program either using SNMP or by reading in a .cfg file

- Regularly retrieve the configuration from the Network Utility, as you need to make dynamic configuration changes

- Make planned network changes from the Configuration Program and send the new configurations to the Network Utility

See the next chapter for specific procedures to transfer configuration files.

Chapter 7. Handling Configuration Files

This chapter describes specific procedures for:

- Viewing and managing configuration files on the hard disk of a Network Utility
- Transferring configuration files from outside Network Utility onto its hard disk
- Transferring configuration files from the Network Utility hard disk

For background information on configuring Network Utility, see Chapter 3, "Performing the Initial Configuration" and Chapter 6, "Configuration Concepts and Methods."

For details on the individual commands introduced in this chapter, see the following chapters in the *MAS Software Users Guide*:

- "Using BOOT Config to Perform Change Management"
- "Configuring Change Management"

Managing Configuration Files on Disk

All the commands to list and manage configuration files on the Network Utility hard disk are located in the boot Config subprocess. The following example shows how to reach this subprocess and list the available commands:

```
*t 6
                <enter>
Config>boot
Boot configuration
Boot config>?
ADD description
COPY software
DESCRIBE software VPD
DISABLE boot choices
ENABLE boot choices
ERASE software
LIST software status
LOCK Config File
SET boot information
TFTP software
TIMEDLOAD software
UNLOCK Config File
EXIT
Boot config>
```

Listing Configurations

The **list** command is the starting point for viewing what configuration files are present in the four positions of each of the two code load banks. This same display is integrated into a number of the commands on the menu.

```
Boot config>li
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 03 Aug 1998 10:04 |
| CONFIG 1 - AVAIL       |                                | 04 Aug 1998 13:50 |
| CONFIG 2 - ACTIVE  *   | example config 1               | 04 Aug 1998 13:52 |
| CONFIG 3 - AVAIL       |                                | 04 Aug 1998 06:41 |
| CONFIG 4 - AVAIL       |                                | 04 Aug 1998 09:43 |
```

7-1

```
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - PENDING        |                               | 05 Aug 1998 03:41 |
| CONFIG 1 - PENDING *    |                               | 31 Jul 1998 12:59 |
| CONFIG 2 - AVAIL       |                               | 31 Jul 1998 09:50 |
| CONFIG 3 - AVAIL       |                               | 31 Jul 1998 09:52 |
| CONFIG 4 - AVAIL       |                               | 31 Jul 1998 12:50 |
+-----------------------+-------------------------------+-------------------+
   * - Last Used Config     L - Config File is Locked

 Auto-boot mode is enabled.    Fast-boot mode is enabled.

Time Activated Load Schedule Information...

The load timer is not currently activated.
Boot config>
```

Image (code load) and configuration states are defined as follows:

ACTIVE   The file was used for the current boot of the Network Utility

AVAIL   This is a valid file that can be made ACTIVE.

CORRUPT The file is not usable. Normally, this is because a file transfer to this position did not complete successfully.

LOCAL   The file will be used only on the next load or reset. After the file is used, it will be placed in the AVAIL state.

NONE   No file is present in the position (the initial state).

PENDING The file will be used on the next reload, reset, or power-up of the Network Utility.

To remind yourself of what is in a particular configuration file, use the **add** command to enter a brief description.

Making a Configuration Active

To make a particular configuration file active, you make it the PENDING configuration file in the bank with the ACTIVE or PENDING code load, then reboot Network Utility. You do this either after the file exists or when you create it, as follows:

- If the file is already on disk, use the **set** command to designate the bank and configuration file position to be used for the next boot.

  You can specify whether the new setting of source bank and configuration is just for the next boot (the state becomes LOCAL) or for all future boots (the state becomes PENDING).

  You normally use the **set** command after transferring a file to the disk using TFTP or XMODEM.

- If you create a new file using the talk 6 **write** command, it automatically becomes the PENDING configuration in the ACTIVE bank.

  When you do a **write**, the system writes the configuration in active memory to the next unlocked position in the ACTIVE bank, rotating in round-robin fashion. You do not pick the file position. If you want to prevent a particular file from being over-written, use the **lock** command.

  Because the new file becomes PENDING, you can do a **write** followed by a **reload** without paying attention to the particular position used, and without having to issue the **set** command.

- If you create a file implicitly by typing **reload** and choosing to save configuration changes, the new file becomes the PENDING configuration before the reboot proceeds.

  The following sequence works the same way as issuing the **write** command:

  *`**rel y**`
  `The configuration has been changed, save it? (Yes or [No] or Abort):` **yes**

- If you create a file by using the Configuration Program **Communicate** option to directly transfer a configuration, the new file becomes the PENDING configuration

  This also works the same way as issuing the **write** command. If you request a reboot from the Configuration Program, this configuration becomes active when the reboot occurs.

Delayed Activation

There are two ways to cause a timed, presumably unattended, activation of a configuration:

- If you are using the Configuration Program and transfer your configuration using the **Communicate** option, you can specify the date and time for the Network Utility to reboot and activate the configuration.

- No matter what method you use to create a configuration file on the Network Utility hard disk, you can used the **timedload** command in the boot Config subprocess to schedule a date and time for the Network Utility to reboot and activate a specified code load and configuration.

  If you choose the current code load and configuration, this function simply becomes a scheduled reboot operation.

File Utilities

The boot Config subprocess provides a number of utility commands for managing configuration files (and code loads) on disk:

add　　to enter a short description of a configuration

copy　　to copy a configuration between banks and/or file positions

erase　　to remove a configuration file and return the position status to NONE

lock　　to prevent the file from being over-written by one of the file creation methods

unlock　to allow a file position to be used again for a new file

Firmware Change Management

Most of the configuration management functions in the boot Config subprocess are also available from Network Utility firmware menus. To access them, select the following sequence starting from the firmware main menu:

- Option 4, "Utilities"
- Option 12, "Change Management"

Loading New Configuration Files

Table 7-1 summarizes the ways you can transfer a configuration from outside the Network Utility to its hard disk. SNMP involves a direct transfer from the Configuration Program to the Network Utility, while TFTP and XMODEM require the configuration file to be on a workstation that acts as a file server to the Network Utility.

Which method you choose to transfer it into the Network Utility depends on how you can attach to the Network Utility, whether you are using the Configuration Program, what software you have on your workstation, and your own preferences. Network Utility configuration files are typically small enough that transfer times over low-speed modems are reasonable.

Table 7-1. Loading Configurations

Physical Attachment	Line Protocol	Transfer Protocol	Tool	Default IP Addresses
Service port + null modem Service port + ext modem PCMCIA modem	Async terminal	XMODEM	Firmware	Not applicable
	SLIP	TFTP	Op-code	Network Utility=10.1.1.2 Workstation=10.1.1.3
		SNMP	Cfg pgm	
PCMCIA EtherJet Ethernet LIC (10 Mbps) Token-Ring LIC	IP	TFTP	Op-code Firmware	Network Utility=10.1.0.2 Workstation=10.1.0.3
		SNMP	Cfg pgm	
Any IP network interface	IP	TFTP	Op-code	No defaults
		SNMP	Cfg pgm	

The following sections summarize each of the possible configuration transfer procedures, grouping them by the tool from which you start the transfer.

Using the Configuration Program

There are two ways to transfer a configuration from the Configuration Program to a Network Utility.

1. Create a router configuration file and then use the Network Utility operational code or firmware as the tool from which to do the transfer.

2. Use SNMP to transfer the configuration to Network Utility memory and hard disk

Exporting a Router Configuration File

After you have started the Configuration Program and created a Network Utility configuration, move to the Navigation Window and:

1. Bring up the **Configure** drop-down menu and select **Create router configuration**.

2. Choose the directory path and filename on the workstation where you are running the Configuration Program, where you want the router configuration file (.cfg) to be stored.

3. Click on **OK** and the Configuration Program writes this file to disk

4. Do a **Save as** under **Configure**, so you also save the configuration in .csf format, the preferred format for archiving.

It is then your responsibility to load the file onto your Network Utility, using either the operational code or firmware to do the loading. You can follow any of the procedures described in "Using the Operational Code" on page 7-6 or "Using the Firmware" on page 7-7.

If your Configuration Program PC or workstation cannot be the TFTP or XMODEM server for the file transfer in these procedures, you must first move the .cfg file to a workstation that can be the server. You can use any file transfer method, such as FTP, to move the file between the workstations.

Directly Sending Using SNMP

In order to use SNMP transfer, you must configure the Network Utility with an IP address and enable SNMP with a read-write community name. Each of the example configurations in Part 2 of this book shows how to configure an IP address and SNMP for this communication, in both the Configuration Program and from talk 6.

If you want to use SNMP to download a Network Utility's very first configuration, see "Configuration Program Procedure for Initial Configuration" on page 3-5.

After you have created a Network Utility configuration at the Configuration Program, do the following to transfer that configuration to Network Utility using SNMP:

1. Bring up the **Configure** drop-down menu and select **Communications**.

2. From the pop-up, select **Single router** if you only want to send the current configuration to one Network Utility, or **Multiple routers** if you want to send any saved configuration to any number of target routers.

3. From the next single-router panel, or multiple-router list panel, select the **Send** option and enter the necessary addressing information for the router(s).

   You can also enter a date and time for the router to be restarted with this configuration, if you wish. It is not necessary to synchronize this date and time with the Network Utility, or even set a date and time at the Network Utility. The Configuration Program translates the date and time you set to an time interval and sends that value to the Network Utility.

4. Click on **OK**, and the Configuration Program starts sending configuration data items to the specified router(s) using SNMP. Sending starts immediately, regardless of whether you specified a later date and time for the target router(s) to reboot.

5. The Configuration Program provides status and result messages about the transfer. If you have problems and are sending to a single router, you may want to try the **Query router information** button instead of **Send**. This option retrieves a short amount of information from the router, so you can use it to see whether you have an SNMP communication path to the router.

When a given router begins to receive a configuration through SNMP, that configuration replaces any talk 6 changes made since the last reboot. When the transfer is complete, the Network Utility writes the received configuration to disk and activates it based on what you selected when you initiated the send operation.

Using the Operational Code

You can use the operational code to pull in a configuration file that was created in one of two ways:

- Exported from the Configuration Program using step 1 on page 7-4
- Previously transferred from this or another Network Utility

As Table 7-1 on page 7-4 shows, the configuration transfer procedures you can initiate from the op-code all use TFTP as the file transfer protocol.

Using TFTP

The op-code procedure for using TFTP to transfer a configuration file to the Network Utility hard disk is:

1. Place the configuration file on a workstation that has TFTP server software installed and IP network physical connectivity to the Network Utility.

2. Configure the IP addresses you will be using

   If you are using a standard network interface including an Ethernet or Token-ring adapter, use the Configuration Program or talk 6 to configure an IP address for the interface in the normal way. (From talk 6, you use **add address** in the IP subprocess.) Activate this configuration change before proceeding.

   If you are using the PCMCIA EtherJet card, use **system set ip** to set the following addresses:

   - IP address: the IP address for the EtherJet card
   - Netmask: the mask for the subnet attached to the EtherJet card
   - Gateway address: the IP address for the TFTP server workstation

   If you are using SLIP, you cannot change the IP addresses but must use those given in Table 7-1 on page 7-4.

3. Transfer the files

   From the * prompt, follow this sequence:

   ```
   *t 6
   Config>boot
   Boot configuration
   Boot config>tftp get config
   ```

   Respond to the prompts as follows:

   - Server IP address: Put the address of the TFTP server workstation.
   - Remote directory: Put the path name to the directory on the server workstation where the configuration file is. Use slashes in the direction expected by your server. Upper versus lower case only matters if it matters to your server.
   - Destination bank: Select bank A or bank B.
   - Destination configuration: Select an unlocked position between 1 and 4.

   Based on the server IP address and the configured Network Utility interface IP addresses, the Network Utility selects which of its interfaces to use to reach the server. The Network Utility gives success or failure status messages as appropriate.

4. Reboot or schedule a reboot to use the configuration

   To activate the new configuration immediately, do the following from the Boot config> prompt:

a. Use the **set** command to select the new configuration so it will be used for the next reboot.

b. Type **Ctrl-p** and then **reload** to reboot the Network Utility

To activate the new configuration later, type **timedload activate** from the Boot config> prompt to select the bank and new configuration, and to specify the date and time for the Network Utility to reboot. You can answer "no" to the questions about loading, because you already did this step.

See the *MAS Software Users Guide* chapter "Configuring Change Management" for more information on the commands in the above procedure.

Using the Firmware

You can use the firmware to pull in a configuration file that was created in one of two ways:

- Exported from the Configuration Program using step 1 on page 7-4
- Previously transferred from this or another Network Utility

As Table 7-1 on page 7-4 shows, the firmware supports both XMODEM and TFTP file transfer protocols.

Using XMODEM

The firmware procedure for using XMODEM to transfer a configuration file to the Network Utility hard disk is:

1. Place the configuration file on the workstation with the terminal emulation software supporting your current user console session.

2. Access the firmware main menu using the procedures described in "Boot Options: Fast Boot and Reaching Firmware" on page 4-12.

3. Make the following sequence of menu selections:

    a. System Management Services (main menu): Option 4, "Utilities"
    b. System Management Utilities: Option 12, "Change Management"
    c. Change Management Software Control: Option 12, "XMODEM software"
    d. Select Type: "Config"
    e. Select Bank: choose Bank A or Bank B
    f. Select Config: choose an unlocked position

    The firmware tells you when to start the file transfer.

4. Go to your terminal emulation package and start the transfer of the file from your workstation server, using whatever name you like. When the transfer starts, the status of the file position changes to CORRUPT, to indicate that it does not contain a complete configuration file. When the transfer completes, the status of the file position changes to AVAIL. You can verify that this has happened using option 7, "List Software", from the firmware Change Management menu.

5. Boot the Network Utility using the configuration you just loaded

    Use Option 9 "Set Boot Information" to select the current op-code bank and the new configuration. Press **esc** to reach the main menu, then **F9** to boot the Network Utility with the new configuration.

Using TFTP

The firmware procedure for using TFTP to transfer a configuration file to a Network Utility hard disk is:

1. Place the configuration file on a workstation that has TFTP server software installed and IP network physical connectivity to the Network Utility.

2. Access the firmware main menu using the procedures described in "Firmware" on page 5-19.

3. Configure the IP addresses you will be using:

   Follow the menu sequence:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 11, "Remote Initial Program Load Setup"
   c. Network Parameters: Option 1, "IP Parameters"

   Set the following addresses:

   - Client IP address: an IP address for the Network Utility LAN card. This is a temporary address that need not be related to the Network Utility operational address for that interface.
   - Server IP address: the IP address of the workstation's LAN adapter
   - Gateway IP address: the IP address of any intermediate router, or repeat the workstation's IP address if there is none
   - Netmask: the mask for the subnet attached to the Network Utility LAN card

4. Initiate the transfer through these menu selections:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 12, "Change Management"
   c. Change Management Software Control: Option 10, "TFTP software"
   d. Select Type: "Config"
   e. Select Bank: choose Bank A or Bank B

5. Enter the path and filename of the configuration file on your workstation

6. If prompted, select the interface through which you want the firmware to do the file transfer

   The firmware transfers the configuration file and gives status messages. On completion, you will be back at the Change Management menu.

7. Boot the Network Utility using the configuration you just loaded

   Use Option 9 "Set Boot Information" to select the current op-code bank and the new configuration. Press **esc** to reach the main menu, then **F9** to boot the Network Utility with the new configuration.

Transferring Configuration Files from Network Utility

You may want to transfer a configuration file *from* a Network Utility for any of the following reasons:

- You are using command line configuration and you want to back up your configuration somewhere other than on the Network Utility hard disk

- You are using command line configuration and you want to export the configuration file to another Network Utility

- You are using both Configuration Program and command line configuration and you want to update the Configuration Program file with recent talk 6 dynamic reconfiguration changes

For the operational code procedures to transfer a configuration to a Network Utility, there is a reverse procedure for transferring a configuration from a Network Utility. The steps are virtually identical, so we list only the essential differences below.

- Import a .cfg file into the Configuration Program

  Transfer the .cfg file to the Configuration Program workstation. Do a **Read router configuration** instead of a **Create router configuration**.

- Use SNMP to transfer a configuration into the Configuration Program

  Do a **Retrieve configuration** instead of a **Send configuration**.

- Use operational code TFTP to send a configuration from the Network Utility

  Type **tftp put config** instead of **tftp get config**.

There are no firmware-based procedures to transfer a configuration from a Network Utility.

Chapter 8. Management Concepts and Methods

In this book, we use the term *managing* to mean all the ways you can monitor and control what is going on with an active Network Utility. These ways include:

- Typing commands on a local or remote console, to query status and change the state of interfaces and protocols
- Monitoring a running log of event messages, either through the same console or at a server for remote logging
- Using an SNMP MIB browser (these terms are defined below) to query the status of interfaces and the functions of the box that have associated SNMP MIB support
- Using an SNMP-based management product and its applications, to monitor and control interfaces and the functions of the box that have associated SNMP MIB support
- Using SNMP-based topology applications, to monitor a protocol-specific (e.g., APPN or DLSw) view of your network and its resources
- Using an SNMP-based management product to monitor SNMP traps sent by the box to report error conditions
- Using an SNA alert focal point product (such as NetView/390) to monitor SNA alerts sent by the box to report error conditions

This chapter gives an overview of these methods and introduces some of the other products you can use to manage your Network Utility.

Console Commands

To enter commands to query and change box status, you must first bring up a local or remote console attachment to an active Network Utility. For details on how to do this and reach the * prompt, see Chapter 2, "Bringing Up a User Console."

Once you have an active console, you use talk 5 to access the Console process.[1] From there, you navigate menus and issue commands to query the status of interfaces and protocols, and to make dynamic operator changes such as:

- Disabling and enabling interfaces
- Recycling connections
- Activating configuration changes

See "Operating (using talk 5, the Console process)" on page 5-12 for an overview of talk 5 commands and the types of status you can view and change from the operator console. Full details on the top-level talk 5 commands are provided in the *MAS Software Users Guide* chapter "The Operating/Monitoring Process (GWCON - Talk 5) and Commands".

By using the talk 5 commands **net**, **protocol**, and **feature**, you can move down in the menu structure and use commands for monitoring and controlling interfaces and particular protocols and features. Interface-level talk 5 commands are documented

---

[1] If you attach to a Network Utility that has never been configured before, you will be in Config-only mode and will not be able to go into the talk 5 Console process. Follow the instructions in Chapter 3, "Performing the Initial Configuration" on page 3-1 to configure your Network Utility for the first time and boot into normal operating mode.

in the chapters of the *MAS Software Users Guide* devoted to the different interface types. Protocol and feature talk 5 commands are described in various chapters of the two-volume *MAS Protocol Configuration and Monitoring Reference.*

Monitoring Event Messages

Why monitor events?

Talk 5 commands provide a snapshot of Network Utility status, but cannot produce a log or trace of events happening inside the box. For this you use ELS, the Event Logging System. By activating the right ELS messages and monitoring the event log, you can follow in real time such events as:

- Interfaces going through test phases, coming up, and going down
- Packets for a particular protocol being sent and received
- DLC links coming up and going down
- CPU utilization changing in response to network activity
- Higher-level protocol connections (e.g., DLSw partner and circuit connections) coming up and going down

By monitoring ELS messages, you can start to answer some basic questions like "is anything happening?", "why isn't the link coming up?", or "is my protocol seeing the traffic I am sending?". It is a powerful tool for debugging basic configuration problems.

Specifying which events to log

To use ELS messages, you first tell the system which of its thousands of pre-defined events you want it to report to you. You can specify the set of active messages using the following criteria:

Subsystem name   Using the short pre-defined name of a software component such as IP, TKR, or DLS, you can refer to all possible messages from that component.

Event number   You can turn individual messages on or off, or specify a range of event numbers. It is sometimes useful to activate all the messages in a subsystem, then turn off a few particularly frequent ones within that subsystem, to avoid obscuring more critical messages.

Logging level   You can specify the severity level of the messages you want to see. For example, you may want to see only unusual error messages, or only trace messages, or include simple informational messages.

Group name   You can specify the name you chose previously when you defined a group of messages.

In addition, you can set up filters on logical interface number, so that for any active set of messages, only those relating to a particular interface appear in the log.

Specifying where to log events

When you activate messages, you choose one of the following destinations for the message:

1. the Monitor process

   You view messages sent to this process using the **talk 2** command from the *
   prompt. See "Event Logging (using talk 2, the Monitor process)" on page 5-17
   for an introduction to using the Monitor process.

2. a remote logging server

   You can set up any PC or workstation that supports a standard *syslog* facility to
   receive a flow of event message packets and save them in a file. The Network
   Utility sends each message in a UDP/IP packet out through a standard network
   interface. Because log message flow can be heavy, a log server is normally
   LAN-attached to the Network Utility.

3. an SNMP trap, sent to an SNMP management station

   The Network Utility packages the event message in an IBM enterprise-specific
   SNMP trap, and sends it in a UDP/IP packet out through a standard network
   interface.

Activating event logging

From the command line, you can use either talk 6 or talk 5 to select the events you
want to log and where to log them. From either process, enter the **event**
subprocess to proceed. If you activate events under talk 6, the changes do not
take effect until you write them to disk and reboot the Network Utility. The
messages for those events will be continuously active from the first reboot on.

If you activate events from talk 5, the system immediately begins to deliver
messages for those events to the destination you specify (talk 2, the log server, or
SNMP management station). When you reboot the Network Utility, those
messages cease to be active. Using talk 5 to activate events is a good way to
debug an immediate problem you may be having. You turn on the events, quickly
jump to talk 2 to see what is happening, and so on. When you reboot later, the
events are deactivated without you having to enter any new commands.

Another useful debug technique is to use the talk 5 event subprocess to view
statistics of how many times any event has been encountered. These statistics are
available even for events that have not been activated.

There is no talk 5 command to activate the current talk 6 ELS configuration. If you
want immediate activation, you must repeat the same commands in talk 5 that you
entered in talk 6.

From the Configuration Program, you can only set up the Network Utility to do
remote logging to a host. You cannot configure which ELS events are active or
direct ELS events to a particular destination. The Configuration Program does
preserve this configuration information, however, if you retrieve a configuration from
a Network Utility, modify other parts of the configuration using the Configuration
Program, and write the configuration back out.

From an SNMP management station, you can use SETs to control most ELS
configuration functions using an enterprise-specific ELS MIB.

For an introduction to some of the key commands to activate and control ELS events, see "Monitoring Events" on page 9-1. For a detailed explanation of ELS concepts and the associated talk 6 and talk 5 commands, see the *MAS Software Users Guide* chapter "Configuring and Monitoring the Event Logging System (ELS)". For a description of every individual ELS message, see the *Event Logging System Messages Guide* either on CD-ROM or on the Web.

Simple Network Management Protocol (SNMP) Support

Background

SNMP is an industry-standard protocol that management stations use to query and set configuration, control, and status information in a managed node. In the Network Utility context, the management station would normally be a PC or workstation with an SNMP management software product installed on it. The managed node would be the Network Utility.

SNMP requests and replies flow inside UDP packets through an IP network between the management station and the managed node. In general, the management station initiates communication by sending requests for information and requests to set data items to new values. The managed node simply carries out these requests and replies to them. A managed node can, however, send an unsolicited message called a *trap* to report an event. A Network Utility might send a trap to report such events as a box reboot or an interface going down.

A *Management Information Base (MIB)* is a virtual information store defining the data items in the managed node that can be accessed from the management station. MIBs are defined in strictly formatted description files which can be read both by people and by management station software.

A managed node product *supports* a MIB when its software can field SNMP requests for the data items documented in the MIB, and retrieve or set its corresponding internal data items. The MIB description file defines for each data item whether the management station can only read it, or can modify its value. Sometimes, a product chooses only to allow read-access to a data item that the MIB documents as write-able. You should consult product documentation to understand the level of access a particular product has implemented.

Most industry-standard protocols and interface types have an associated IETF standard MIB with an RFC number. Standard MIBs define data items that are common to most implementations of the associated protocol or interface type. Vendors cannot always wait for a MIB to reach standard RFC status within the IETF, and sometimes ship support for a pre-standard *Internet Draft* version of the MIB.

In addition to the standard MIBs, many product vendors develop their own MIBs to define data items that are unique to their products. For example, Network Utility supports MIBs that give access to memory and CPU utilization information, for which there is no standard MIB. In SNMP parlance, these vendor MIBs are called *enterprise-specific* MIBs.

MIB Support

IBM Network Utility supports a comprehensive set of standard and enterprise-specific MIBs for monitoring and managing resources. The current list numbers somewhere between 40 and 50 MIBs.

You can find a "README" file documenting Network Utility MIB support by accessing the appropriate software release directory on the World Wide Web at URL:

```
ftp://ftp.networking.raleigh.ibm.com/pub/netmgmt/netu
```

In the same directory, you can find the MIB description files themselves, ready to be retrieved using FTP and loaded into a management station. Whenever possible the files are compiled into SNMP Version 1 format, to make them compatible with the widest possible variety of management station software.

For the standard and Internet Draft MIBs, the compilation process strips out introductory explanatory text and page formatting that helps make a MIB more readable. To get the full pre-compiled version of an RFC or Internet Draft MIB, retrieve it from an IETF FTP site as you would any RFC or Internet Draft. You can start at the following URL and follows links to the RFC or Internet Draft repository:

```
http://www.ietf.org
```

Getting Started

At the Network Utility

Before an SNMP management station can communicate with your Network Utility, you must first configure SNMP in the Network Utility with the appropriate access enabled. You can use either the Configuration Program, talk 6, or talk 5 to enable SNMP and set up a *community name* that grants access to one or more management stations. From talk 6 or talk 5, use **protocol snmp** to access the Config and Console subprocesses for working with SNMP. As shown in step 2 on page 3-3, you can also use Quick Config to enable SNMP and set up a read or a read-write community name.

See the *MAS Protocol Configuration and Monitoring Reference Volume 1* chapters "Using SNMP" and "Configuring and Monitoring SNMP" for more background information, and for a description of the SNMP talk 6 and talk 5 commands.

At the Management Station

Before a management station can provide any significant support of a managed node, it must know what MIBs that managed node supports. If you are using any of the IBM products described in "IBM Nways Manager Products" on page 8-7, you do not have to do anything to set this up. Each of them has the MIBs that Network Utility supports already compiled in.

If you are using some other management product, you may have to set up this knowledge. Management stations typically provide a facility for loading compiled MIB modules into the station. When you are preparing a management station to manage Network Utility, set it to read in all the MIBs from the appropriate directory under the URL given in "MIB Support."

If you intend to send traps from Network Utility to the management station, you may also need to set up the management station to issue messages or take specific actions on receipt of a trap.

SNA Alert Support

IBM's Systems Network Architecture (SNA) defines a rich set of protocol flows for the purpose of managing network products. A key part of that architecture is the ability for the managed node to send an unsolicited error or event report, called an *alert*, to an SNA management station. An alert contains a sequence of submessages that enable the management product to report to an operator such things as:

- the identity of the node that built the alert
- the error or event that prompted the alert
- several possible causes for the problem
- possible corrective actions

The SNA management product most commonly used to receive alerts is NetView/390. In SNA architecture, such a product is called an alert *focal point*. A product in the network that can receive and forward alerts on behalf of other products is called an *entry point*.

When you are using Network Utility as an APPN network node, it has the capability to establish LU6.2 sessions with alert focal points and send native SNA alerts to report error conditions in the box and in the network. The following are some of roughly 30 pre-defined events that trigger an alert from Network Utility's APPN function:

- Session setup failure
- Invalid XID received, XID protocol error
- HPR or DLUR configuration or protocol error
- CP-CP session failure
- Out of resources
- Subcomponent protocol error

If one of these events occurs and Network Utility has no current focal point session on which to send the alert, it queues the alert for later transmission. You can configure the depth of this "held alert" queue. You cannot configure which of these events will trigger an alert.

The LU6.2 session on which alerts flow can be established either by the focal point or by the Network Utility. You don't have to configure any special parameters at your APPN Network Utility, to enable it to accept a session from an alert focal point and send alerts. If you want the Network Utility to actively set up sessions and forward alerts, you configure the name of one or more *implicit* focal points as part of your APPN configuration. If the primary focal point cannot be reached, Network Utility attempts to reach the other configured names.

In addition to sending alerts for events it detected, Network Utility can serve as an SNA entry point and forward alerts on behalf of other SNA nodes with which it has sessions. No configuration is required to enable this function.

Getting Started

You can use the Configuration Program or talk 6 to configure focal point names if you want the Network Utility to activate the focal point sessions. From talk 6, use **protocol appn** to access the Config subprocess for working with APPN, then use the **add focal-point** command.

For more background, see the *MAS Protocol Configuration and Monitoring Reference Volume 2* "APPN" chapter sections "Entry Point Capabilities for APPN-related Alerts", "Configurable Held Alert Queue", and "Implicit Focal Point". The command names are given in the "Router Configuration Process" section in the same chapter.

Network Management Products

Both SNMP and SNA management flows require a product separate from Network Utility, to manage a display of the network and the Network Utility, to query the Network Utility for status, or to receive unsolicited event reports from the Network Utility. This section lists some of the products you might use to perform these tasks.

SNMP MIB Browsers

A *MIB browser* is a small PC or workstation application that can load MIB definitions, query or set data items in a managed node, and decode returned values and results into a easily readable form. In SNMP terms it is a management station, but a MIB browser lacks the power and sophistication of a full-fledged SNMP management platform such as those described below. MIB browsers are frequently packaged as part of such platforms, but can also be stand-alone products.

IBM Nways Manager Products

The following IBM SNMP network management products are specifically intended to manage Network Utility and a wide variety of other IBM and non-IBM networking products. Each of them provides a graphical topology view of your network resources, with a color-coded status of resources and overall status of each network. Each supports automatic discovery of network resources, and automatic updates to a network map in response to network changes.

IBM Nways Manager for AIX

Designed for managing medium to large network environments, you install this product on a workstation running AIX, IBM's version of UNIX. Nways Manager for AIX runs on top of Tivoli TME 10 NetView, which was formerly known as "Netview for AIX" and "NetView/6000". Tivoli TME 10 NetView provides general network management platform capabilities such as the management of LAN topologies, fault and event recording, and error logging. When combined with IBM's SNA Server for AIX, Tivoli TME 10 NetView can also map SNMP traps to SNA alerts. For Network Utility, this permits an SNA alert to flow for virtually any defined ELS event.

Nways Manager for AIX provides the following capabilities on top of the base Tivoli TME 10 NetView functions:

- A Network Utility-specific management application

When you select a Network Utility from the network topology view, you see a graphic of the front panel of the Network Utility, with color-coded interface status. A navigation window on the side allows you to access all SNMP MIB information provided by Network Utility, either in graphical or tabular form. Included in this application are the ability for you to:

– View or change adapter and interface status
– Display statistics at a component or interface level
– Receive real-time, color-coded status at a glance
– Define and monitor performance thresholds
– Define and monitor real-time and historical statistics
– Monitor real-time events

From the Network Utility application, you can launch:

– the 2216/Network Utility graphical Configuration Program, to configure the box

– a telnet session to the Network Utility, so you can use the command line interface to configure, monitor, and control the Network Utility

Because the Network Utility management application is Java-based, you need not be at the workstation running Nways Manager to use it. You can bring up the application from a PC or workstation running a JDK-compliant Web browser, connected over your intranet or the Internet to the main Nways Manager workstation. For details on which web browsers and versions of JDK are required, see the Nways Manager product pre-requisites at:

    http://www.networking.ibm.com/netmgt

Java management support includes viewing real-time status of the Network Utility and the ability to do performance management. For security reasons, you cannot launch the Configuration Program from a Java web browser.

- Distributed Intelligent Agents

  To provide support for larger networks, you can use boxes other than the Nways Manager workstation to poll the managed nodes in your network. Offloading polling from the manager workstation frees its processor to do other tasks, and it frees network bandwidth as you place the polling closer to the devices being polled. These "agents" of the manager can be configured to notify Nways Manager when thresholds are exceeded.

  The intelligent agent software is Java-based and is downloaded from Nways Manager. The agents can be placed in any Java-enabled (Java virtual machine) workstations in the network. Nways Manager can also use the distributed polling capabilities provided by the TME 10 Mid-Level Manager.

- APPN topology support

  Nways Manager for AIX provides an APPN-level view of the topology of your network. You can discover participating APPN resources, view them, and view their status as color-coded icons. APPN protocol performance and error events (data and graph) are also provided. This application does not present Branch Extender or Extended Border Node topologies.

- DLSw topology support

  Nways Manager for AIX can also show you a DLSw topology view of your network, including DLSw connectivity, resources, and color-coded status. The

topology is refreshed as new nodes are discovered. This application does not present the topology of DLSw IP multicast groups.

- VLAN, ATM, and RMON support

  Nways Manager for AIX has comprehensive support for products implementing virtual LANs, for ATM networks, and for collecting, correlating, and displaying data from RMON and ECAM probes.

The first version of Nways Manager for AIX with specific support for Network Utility is the 3Q98 PTF of Version 1.2.2.

For more information about Nways Manager for AIX including specifications and system requirements, start with the World Wide Web at URL:

    http://www.networking.ibm.com/cma/cmaprod.html

The pages at this URL describe the separately-priced components of Nways Manager for AIX, and which components perform which of the above functions.

IBM Nways Workgroup Manager for Windows NT

Designed for managing small to medium network environments, Workgroup Manager is a 32-bit native Windows NT application that operates on NT Version 4.0. Unlike Nways Manager for AIX, Workgroup manager is self-contained and does not use an underlying network management platform. It must therefore provide a number of platform functions itself.

Key features of Nways Workgroup Manager for Windows NT include:

- Automatic discovery of your IP network
- Real-time, graphical views of network topology
- Ability to browse, update, and compile MIBs
- Color-coded and aggregated network and device real-time status
- Trouble ticketing
- Trap management, including specifying trap severities
- Trap compiler
- Polling configuration and notification
- Performance threshold configuration and notification
- Inventory management
- Collection and presentation of real-time and historical statistics

Nways Workgroup Manager for Windows NT supports exactly the same Network Utility-specific Java management application described above for Nways Manager for AIX. You can run the Network Utility management application from a Java-capable web browser. Nways Workgroup Manager for Windows NT also supports Distributed Intelligent Agents.

Nways Workgroup Manager for Windows NT does not support the APPN and DLSw topology applications that Nways Manager for AIX does. Nways Workgroup Manager for Windows NT's topology display is based on IP connectivity between the managed nodes.

The first version of Nways Workgroup Manager for Windows NT with specific support for Network Utility is the 3Q98 PTF of Version 1.1.2.

IBM Nways Manager for HP-UX

Designed for managing medium to large network environments, you install this product on a workstation running HP-UX, Hewlett Packard's version of UNIX. Nways Manager for HP-UX runs on top of HP's *Network Node Manager* management platform software, previously known as "HP OpenView".

In this environment, network node manager provides the base management platform functions, including topology display, trap management, etc.. Unlike Nways Manager for AIX, you associate IBM network devices with the appropriate Nways Manager for HP-UX management application.

From Nways Manager for HP-UX, you can launch the same Network Utility-specific Java management application described above for Nways Manager for AIX. Nways Manager for HP-UX also supports Distributed Intelligent Agents.

Nways Manager for HP-UX does not support the APPN and DLSw topology applications that Nways Manager for AIX does.

The first version of Nways Manager for HP-UX with specific support for Network Utility is Version 1.2 (August 1998).

NetView/390

NetView/390 is a host-based management product for managing medium to large SNA networks. There are several ways you can use NetView/390 to manage a Network Utility and the SNA products it can connect to the host:

* Controlling SNA resources (activating and deactivating links, PUs, and LUs)

    – When Network Utility is running DLSw, NetView/390 can control the the links that DLSw is representing, and the PUs and LUs in remote SNA end stations.

    – When Network Utility is running TN3270 server support, NetView/390 can control the local PUs and LUs represented in the Network Utility.

    – When Network Utility is running DLUR for downstream nodes, NetView/390 can control the PUs and LUs the Network Utility is serving, and the links between Network Utility and those nodes.

    – When Network Utility is bridging SNA end station traffic, NetView/390 can control the end station PUs and LUs.

    – When Network Utility is running APPN, DLSw, or bridging SNA traffic, NetView/390 can control adjacent links between the host and Network Utility.

    – When Network Utility is running LSA direct gateway function, NetView/390 can control the LAN links that appear to be local to VTAM, as well as the PUs and LUs of the attached SNA end stations.

* Monitoring network errors and topology

    – NetView/390 can be the alert focal point when Network Utility is serving as an APPN node, both for the alerts that Network Utility generates and those it forwards from other nodes.

    – When Network Utility is running DLSw, DLUR, or bridging SNA traffic, NetView/390 can receive alerts, response time information, or any other SSCP-PU flow from a downstream PU.

– NetView/390 can be the alert focal point for Network Utility traps that have been converted to alerts by Tivoli TME 10 NetView and SNA Server for AIX.

– Through related products *SNA Topology Manager*, *APPN Accounting Manager*, and *APPN Topology Integrator*, NetView/390 can acquire and monitor the topology of an APPN network including Network Utility and other SNMP-capable APPN products.

Chapter 9. General Management Tasks

This chapter gives procedures and commands for important Network Utility operations. It serves as a supplement to some of the concept presentations in previous chapters.

Monitoring Events

This section supplements the background information on event logging and viewing provided in "Event Logging (using talk 2, the Monitor process)" on page 5-17 and "Monitoring Event Messages" on page 8-2. It introduces the commands that control what events are logged, and where they are logged.

Accessing the Event Logging System

You must use the command line interface to activate event logging. From the Configuration Program, you can only configure general remote logging parameters.

From either the main talk 5 or talk 6 prompts, type **event** to enter the ELS Console or Config subprocess, respectively. You see essentially the same commands whether you are working under talk 5 or talk 6. Talk 5 ELS commands take effect immediately and are quite useful for turning on messages to debug a particular flow in a running system. From talk 6, you configure the events you want to be logged all the time, so you do not have to activate them each time you reboot the Network Utility.

Commands to Control Event Logging

There are six basic commands for activating and deactivating event logging, two for each of the three possible destinations of log messages:

- **disp** and **nodisp** control which events are locally logged to talk 2

- **trap** and **notrap** control which events generate SNMP traps

- **remote** and **noremote** control which events are remotely logged to a syslogd-capable host

All of these commands use the same method for specifying which events are to be activated or deactivated. Following the name of the command on the command line, you normally type one of the following (there are other options):

- **event** *subsystem.event#*, to specify a single pre-defined event

  subsystem is the name of a functional component as known to ELS, such as "dls" for DLSw or "esc" for ESCON. You can type **li sub** to get a list of ELS subsystem names.

  event# is the number of a pre-defined event, typed with leading zeros. You can type **li sub** *subsystem* to get a quick list of the events in a particular subsystem.

- **sub** *subsystem logging_level*, to specify some set of the pre-defined events in an ELS subsystem

  subsystem is the ELS subsystem name described above. The value "all" selects all subsystems.

logging_level is optional and defaults to "standard", which includes all error and unusual informational messages. The value "all" selects all messages in the subsystem.

The following list gives a few examples of these commands:

disp sub all
Enables logging to talk 2 of all error and unusual informational messages in all ELS subsystems. This is a good general setting to configure in talk 6.

rem sub dls
Enables remote logging of all error and unusual informational messages in the DLS subsystem. Separately, you need to configure the destination host for remote logging.

disp sub sdlc all
Enables logging to talk 2 of all messages in the SDLC subsystem. You might enable all messages when trying to trace a error situation.

nodisp ev sdlc.008
Disables logging to talk 2 of a particularly chatty SDLC message, which may be interfering with seeing more important messages in the error log.

trap ev dls.475
Enables sending an SNMP trap when a particular DLSw QLLC error event occurs.

For detailed information about these commands, how to configure remote logging, what the logging levels are, and more, see the chapter "Using the Event Logging System (ELS)" in the *MAS Software Users Guide*.

Monitoring Memory Utilization

This section describes how Network Utility memory is used, and how you can monitor its status.

Network Utility Memory Usage

A Network Utility currently ships with 256MB of main memory. When you boot the system, it loads operational code from disk into this memory, taking a certain amount of memory space for each load module. Once the operational code is loaded, the system splits up the remaining memory between APPN/TN3270 (if configured) and the routing function. The routing function includes IP, DLSw, TCP, channel gateway; in short, every function except APPN and TN3270 server.

When you configure APPN either from the Configuration Program or the command line, you can specify the amount of memory to be reserved for APPN. In Network Utility, this value is pre-set to the memory required for a maximum TN3270E server configuration. This value should be reasonable for non-TN3270 APPN applications as well, so you should not need to change it. If your configuration does not enable APPN, Network Utility ignores the configured value and does not reserve memory for APPN. If your configuration enables APPN, Network Utility allocates the specified amount of memory to APPN, then allocates all remaining memory to the routing function.

You can monitor memory utilization in a running Network Utility either from a command line console, or from an SNMP management station. Either way, you

look separately at the status of APPN memory and the status of routing function memory. Once the system is loaded, these memory partitions are fixed and are managed independently.

Monitoring Memory from the Command Line

To monitor routing function memory from the command line,

1. From the * prompt, type **talk 5** and press **Enter** to reach the + prompt.
2. Type **mem** to see summary and detailed statistics on current memory utilization. The output uses the term *heap* to refer to the memory being used by the routing function.

To monitor APPN/TN3270 memory from the command line,

1. From the * prompt, type **talk 5** and press **Enter** to reach the + prompt.
2. From the + prompt, type **p appn** to reach the APPN Console subprocess.
3. Type **mem** to see summary and detailed statistics on APPN memory utilization. The output breaks down APPN memory into various pieces and shows the state of each piece.

Monitoring Memory using SNMP

Network Utility supports IBM enterprise-specific MIBs that provide access to memory utilization information both for the routing function and for APPN/TN3270.

The Nways Manager products discussed in "IBM Nways Manager Products" on page 8-7 provide full statistical support for both the APPN and routing function memory partitions. For either partition, you can view real-time and historical utilization information. You can set up alarm thresholds for either utilization percentage, so you can be notified when memory utilization reaches a certain level.

You can also configure Network Utility from the command line to send an SNMP trap when available routing function memory falls below a given threshold. From the talk 6 prompt `Config>`, type the command **patch mosheap-lowmark**, and give the percentage value if you want to change it from the default value of 10%.

Monitoring CPU Utilization

This section describes how to control CPU monitoring, and get reports from talk 5 or direct periodic messages to talk 2.

Accessing Performance Monitoring

From either the main talk 5 or talk 6 prompts, type **perf** to enter the Performance Monitoring Console or Config subprocess, respectively. From talk 6 and from the Configuration Program, you can enable or disable CPU utilization monitoring and set its operating parameters as part of your Network Utility configuration. From talk 5, you can make the same changes take effect immediately, and you can get reports on the CPU utilization in a running Network Utility.

Console Commands to Monitor CPU Utilization

Once you are at the `PERF Console>` prompt, the following commands are available to you:

report

Give a summary of current CPU utilization, high water marks, and historical distribution of values.

enable cpu, disable cpu
   Control the overall gathering of CPU utilization information. By default,
   Network Utility runs with CPU utilization enabled, with negligible impact to
   system performance. If you are running TN3270 server functions with Network
   Dispatcher, it is particularly important to leave CPU utilization enabled.

enable t2, disable t2
   Control the generation of a periodic ELS message in talk 2 showing current
   CPU utilization. If you enable this message, you can avoid having to
   repeatedly type the **report** command to monitor how CPU utilization is
   changing.

set, list, clear
   Set the time window for statistics gathering. View the current values of all
   settings. Reset statistics.

All the same commands or parameters are available from talk 6 and the
Configuration Program, except for **clear** and **report**.

For more information on these commands and examples of their output, see the
chapter "Configuring and Monitoring Performance" in the *MAS Software Users
Guide*.

Monitoring CPU Utilization using SNMP
Network Utility supports an IBM enterprise-specific MIB that provides access to
current and historical CPU utilization information.

The Nways Manager products discussed in "IBM Nways Manager Products" on
page 8-7 provide full statistical support for Network Utility CPU utilization. You can
view both real-time and historical utilization information. You can set up alarm
thresholds for the utilization percentage, so you can be notified when it reaches a
certain level.

Chapter 10. Software Maintenance

This chapter describes what you need to know to receive and install fixes for Network Utility software problems, and to upgrade to new software releases containing new function.

This information includes:

- how the software is named and packaged
- how to download new software versions from the World Wide Web
- how to load software onto Network Utility
- how to call for product service and support

Software Versions and Packaging

Version Naming

The software that operates Network Utility is called *Multiprotocol Access Services*, or MAS. MAS also operates the IBM 2216-400, but there are different, separate packages of MAS for each product. The MAS packages for Network Utility are characterized by:

- Pre-set configuration defaults, to tune Network Utility for its intended applications.

- Specialized function packaging oriented toward the key uses of Network Utility. For example, some of the general multi-protocol routing functions of the 2216-400 such as IPX, Appletalk, Banyan Vines, and DECNet, are not available in the Network Utility packages.

Specific levels of MAS are identified by the following numbers:

Version   A new function release occasionally requires a new version number. Sometimes this is related to a price increase, but it could also be related to a shift in how IBM is distributing the software. A new version number does not mean that the release has any more new function than a release that only has a new release number.

Release   This number changes with every new function release.

Modifier   This number signals a new function release that is a small change to a larger base new function release. It follows the decimal point in the format "MAS Vv Rr.m PTF p".

PTF   This number represents a maintenance level, described below.

The initial code base for Network Utility is: MAS V3R1.0 PTF 1. Because IBM is using the same release numbering as for the 2216-400 packages of MAS, you can easily correlate the function and maintenance level of software on the two products.

To see the software level of the code that is actively running in your Network Utility, move to the base talk 5 menu and type **c** (for "configuration"). The software version part of the output of this command uses the format "MAS Vv.r Mod m PTF p".

To see the software level of the code loads on the Network Utility hard disk, move to the base talk 6 menu, type **boot** to enter the boot Config subprocess, then type **describe**

Maintenance Levels

When you access the World Wide Web pages that contain recent versions of Network Utility software, you see some of the following terms for different maintenance levels of the Network Utility packages:

GA Level The software level first made "generally available" to IBM customers. This is the level initially shipped on the hard disk of new Network Utility boxes. GA level software undergoes an extensive product-level and system-level test before it is released. General availability normally corresponds to a new Version or Release of the software (the initial release of Network Utility in a PTF is an exception to this rule).

PTF A major maintenance release ("program temporary fix") that accumulates a large number of fixes and undergoes a regression test of most major software functions. After a Release has been deployed for some time, IBM will typically start to ship a stable PTF on the hard disk of new products.

EPTF A small maintenance release ("emergency PTF") that comes out on a more frequent basis, involves fewer fixes, and undergoes a regression test of the specific areas affected by the fixes.

PTFs and EPTFs are cumulative, in that each supersedes all previous PTFs and EPTFs. You only need to install the latest PTF or EPTF to obtain all previous fixes.

Feature Packaging

There are two feature packages of Network Utility software, corresponding to the two different models of Network Utility:

Model	Feature Number	Description
TX1	1001	Base code, including DLSw, APPN, and IP
TN1	1021	Base code plus TN3720e server function

Based on the model you purchased, your Network Utility comes pre-loaded with the proper software package in both banks of the hard disk. When you load a new maintenance level of software, you load the same package that is already on the Network Utility. Your userid and password to the Network Utility Web page (discussed below) gives you access only to packages priced the same as or less than the one you initially purchased.

Note that there is only one version of the Configuration Program, and it supports the software functions in all the software packages. If you configure functions that are not supported in the particular software package you have on the router, the router software ignores that part of the configuration.

From the command line, you cannot configure or monitor software functions that are not present in the software load you are running.

Getting Web Access to the Software

To update your Network Utility software, you must first download the appropriate maintenance level from the World Wide Web. To find the new software, start with the main Network Utility product page at URL:

```
http://www.networking.ibm.com/networkutility
```

Click on "Downloads" to move to a page containing the following:

- General information about accessing the software
- Detailed procedures for downloading and installing the software
- Links to the latest maintenance levels of the Configuration Program, with associated README files
- Links to the latest maintenance levels of MAS, with associated PTF or EPTF content files

When you follow a link to a particular maintenance level of the Configuration Program, you move to a page with packed binary versions of the 2216/Network Utility Configuration Program for each of its supported operating systems. Anyone can download these files. The associated README file gives instructions for unpacking and installing the new version of the Configuration Program.

When you follow a link to a particular maintenance level of MAS, you move to a page with compressed packed binary versions of each of the Network Utility software features listed above. You currently need a user ID and password to be able to download these files. Some user IDs give access to higher-priced features, while others give access only to the base feature.

To obtain a user ID and password, follow the instructions in the letter that came packaged in the box with your new Network Utility. In general, the current process (subject to change) is:

1. You send e-mail to IBM requesting an ID/password form, using a keyword from the above-mentioned letter
2. IBM returns an electronic order form with certain fields filled in
3. You complete the form and send it electronically to IBM
4. IBM assigns a user ID and initial password and notifies you of it via e-mail

You should request a user ID in advance of actually needing it, to give time for this process to complete.

See the general information part of the "Downloads" web page for further information about IDs and passwords, including how to change your password. If the password/ID process changes, the revised process will be posted on this page.

Downloading and Unpacking Files

The web page for downloading a particular MAS maintenance release contains files for each of the supported software features. Each file contains a complete set of software for Network Utility. When you install a maintenance level of Network Utility software, you completely replace all the existing software with the new level.

To download the software in a particular file and transfer it to the router, you:

1. Use your web browser to download the complete file in binary to your workstation.

2. Transfer the file to the workstation from which you will load it into the router. We call this the "server workstation" because it acts as a file server to the router. You can use FTP or any other file transfer method for this step.

3. At the server workstation, unpack the single downloaded file into a number of router software files. These files are called "load modules" and have the file extension "ld" (or "LD" if your file system does not support mixed case).

4. Using TFTP or XMODEM, you transfer the load modules into the router.

The web page contains two files for each software feature, each constructed a different packing utility. Choose the version that your server workstation software can unpack. Normally you would choose as follows:

Server Operating System	File Format	Unpack Command
DOS, Windows, or OS/2	.zip	pkunzip
UNIX or AIX	.tar	tar -xvf

When you unpack the router software, make sure that all ".ld" files are in the same directory, and have file system permissions to give appropriate "read" access. Do not change the names of any of the ".ld" files. Do not mix files between different Network Utility feature packages, or between different maintenance levels of the same package. Keep each package distinct and separate with a different path name on your server workstation.

Loading New Operational Code

Operational code (op-code, for short) is the software that runs the normal packet forwarding and system services functions of Network Utility. Op-code includes the base operating system, protocols, features, diagnostics, and the command line interface code. The vast majority of software changes in PTFs and EPTFs are changes to the operational code.

To load and activate new operational code, you must:

1. Transfer the unpacked load modules from your server workstation into one of the two op-code banks on the Network Utility hard disk

2. Set the router to boot from the bank with the new op-code

3. Reboot the router, or schedule it to reboot at a later date and time

Table 10-1 on page 10-5 summarizes the different ways you can transfer operational code from a server workstation to a Network Utility hard disk. Which method you choose depends on how you can attach the workstation to the router, what software you have on your workstation, and your own preferences. Here are some important points to consider:

- The size of all the .ld files combined is over 10MB. If you can possibly use a LAN or network interface instead of the service port or modem, you should do so to avoid hours of file transfer time.
- The TFTP-based methods from the op-code and firmware automatically transfer all .ld files in a single operation. With XMODEM, you must manually specify the name of each of the roughly 20 .ld files that make up a software load.

Table 10-1. Loading Operational Code				
Physical Attachment	**Line Protocol**	**Transfer Protocol**	**Tool**	**Default IP Addresses**
Service port + null modem Service port + ext modem PCMCIA modem	Async termimal	XMODEM	Firmware	Not applicable
	SLIP	TFTP	Op-code	Network Utility=10.1.1.2 Workstation=10.1.1.3
PCMCIA EtherJet Ethernet LIC (10 Mbps) Token-Ring LIC	IP	TFTP	Op-code Firmware	Network Utility=10.1.0.2 Workstation=10.1.0.3
Any IP network interface	IP	TFTP	Op-code	No defaults

There are detailed step-by-step installation instructions for all procedures on the Network Utility Code Downloads web page referenced earlier. The following sections summarize these procedures without providing every response to every prompt.

Using the Operational Code

As Table 10-1 shows, the transfer procedures you can initiate from the op-code all use TFTP as the file transfer protocol.

Using TFTP

The op-code procedure for using TFTP to transfer op-code files to a Network Utility hard disk is:

1. Configure the IP addresses you will be using

   If you are using a standard network interface including an Ethernet or Token-ring LIC, use the Configuration Program or talk 6 to configure an IP address for the interface in the normal way. (From talk 6, you use **add address** in the IP subprocess.) Activate this configuration change before proceeding.

   If you are using the PCMCIA EtherJet card, use **system set ip** to set the following addresses:

   - IP address: the IP address for the EtherJet card
   - Netmask: the mask for the subnet attached to the EtherJet card
   - Gateway address: the IP address for the TFTP server workstation (not actually used in this procedure)

   If you are using SLIP, you cannot change the IP addresses but must use those given in Table 10-1.

2. Transfer the files

   From the * prompt, follow this sequence:

   ```
   *t 6
   Config>boot
   Boot configuration
   Boot config>tftp get load mod
   ```

   Respond to the prompts as follows:

   - Server IP address: Put the address of the TFTP server workstation.
   - Remote directory: Put the path name to the directory on the server workstation where all the .ld files are. Use slashes in the direction

expected by your server. Upper versus lower case only matters if it matters to your server.

- Destination bank: Select bank A or bank B. You cannot select the currently active bank.

Based on the server IP address and the configured Network Utility interface IP addresses, the router selects which of its interfaces to use to reach the server. The router gives success or failure status messages as appropriate.

3. Reboot or schedule a reboot

To activate the new load immediately, do the following from the Boot config> prompt:

a. If necessary, use the **copy config** command to place the configuration you want into the bank where you just placed the new code load.
b. Use the **set** command to select the bank you just loaded to boot next, and to select the configuration you want.
c. Type **Ctrl-p** and then **reload** to reboot the router

To activate the new load later, do the following from the Boot config> prompt:

a. If necessary, use the **copy config** command to place the configuration you want into the bank where you just placed the new code load.
b. Type **timedload activate** to select the bank and configuration, and to specify the date and time for the router to reboot. You can answer "no" to the questions about loading the bank, because you already did this step.

See the *MAS Software Users Guide* chapter "Configuring Change Management" for more information on the commands in the above procedure.

Using the Firmware

As Table 10-1 on page 10-5 shows, you can use either XMODEM or TFTP from the firmware to transfer op-code to the Network Utility hard disk. XMODEM is not recommended because modem speeds are too slow for these large op-code files and XMODEM requires regular interaction. TFTP over LAN interfaces is the preferred transfer method when you are working from the firmware. Nevertheless, this section summarizes all the possible procedures in case you need to use them.

Using XMODEM

The firmware procedure for using XMODEM to transfer op-code files to a Network Utility hard disk is:

1. Access the firmware main menu using the procedures described in "Firmware" on page 5-19.

2. Make the following sequence of menu selections:

a. System Management Services (main menu): Option 4, "Utilities"
b. System Management Utilities: Option 12, "Change Management"
c. Change Management Software Control: Option 12, "XMODEM software"
d. Select type: "Load Image"
e. Select bank: choose Bank A or Bank B

The firmware tells you when to start the file transfer.

3. Go to your terminal emulation package and start the transfer of the file "LML.ld" from your workstation server.

4. After transferring LML.ld, you must transfer every other ".ld" module on your workstation server, one by one. LML.ld must be first, but after that the order does not matter. You must include Firm.ld. When the file transfer begins, the status of the bank changes to CORRUPT, to indicate that it does not contain a complete valid code load. When the Network Utility has received the last load module, the status of the bank changes to AVAIL. You can verify that this has happened using option 7, "List Software", from the firmware Change Management menu.

Using TFTP

The firmware procedure for using TFTP to transfer op-code files to a Network Utility hard disk is:

1. Access the firmware main menu using the procedures described in "Firmware" on page 5-19.

2. Configure the IP addresses you will be using:

   Follow the menu sequence:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 11, "Remote Initial Program Load Setup"
   c. Network Parameters: Option 1, "IP Parameters"

   Set the following addresses:

   - Client IP address: an IP address for the Network Utility LAN card. This is a temporary address that need not be related to the router operational address for that interface.
   - Server IP address: the IP address of the workstation's LAN adapter
   - Gateway IP address: the IP address of any intermediate router, or repeat the workstation's IP address if there is none
   - Netmask: the mask for the subnet attached to the Network Utility LAN card

3. Initiate the transfer through these menu selections:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 12, "Change Management"
   c. Change Management Software Control: Option 10, "TFTP software"
   d. Select Type: "Load Image"
   e. Select Bank: choose Bank A or Bank B
   f. Select Load Type: "Modules"

4. Enter the path on your workstation to the directory with all the load modules

5. If prompted, select the interface through which you want the firmware to do the file transfer

   The firmware transfers each load module in turn and gives status messages. On completion, you will be back at the Change Management menu.

6. Boot the router using the op-code you just loaded

   Use Option 9 "Set Boot Information" to select the new op-code bank (and configuration) to boot from. Press **esc** to reach the main menu, then **F9** to boot the Network Utility with the new op-code.

See the instructions on the Network Utility Code Downloads web page for more details on the above procedures and the errors you may encounter.

Upgrading Firmware

Overview

Firmware is low-level software that drives the power-on and boot logic of Network Utility. It resides in non-volatile flash memory rather than on the hard disk, so in the event of a failure such as corruption of your operational software load on disk, you can retrieve new software or configuration files and get back up and running. To *upgrade* the firmware means to write a new version of it to flash, replacing the old version.

You need to upgrade the firmware under two conditions:

1. IBM ships a PTF or EPTF that you need to fix a problem, and that PTF or EPTF requires a firmware upgrade.

   The content document associated with each PTF or EPTF states whether firmware upgrade is required or not.

2. You want to install a new MAS functional release

   Moving to a new release almost always requires a firmware upgrade.

On the Network Utility Code Download web page, there are no separate files containing new versions of the firmware. Instead, the firmware is one of the load modules packed inside the .zip and .tar files along with the operational code load modules. The firmware load module has the file name "Firm.ld". Every PTF and EPTF contains a new Firm.ld file, even if its contents are the same as an older maintenance level.

When you follow the procedures described in "Loading New Operational Code" on page 10-4, you are downloading a new version of firmware from the web and transferring it to Bank A or Bank B of your hard disk. Placing Firm.ld into a disk bank and rebooting from that bank has absolutely no effect on the active firmware, which is running from flash memory. In order to upgrade to new firmware, you must write the new firmware to flash memory.

There are two ways to upgrade the firmware in flash memory. You initiate both of these from the current firmware that you are replacing. There is no command in the operational code to upgrade firmware.

1. Write a copy of Firm.ld that is already on disk, to flash memory

2. Transfer Firm.ld from a server workstation onto disk using TFTP or XMODEM, then write it from disk to flash memory

Of these two methods, the first is the simplest and is recommended whenever you have the new level of firmware on disk already.

To expand on the second method, Table 10-2 on page 10-9 summarizes the different ways you can transfer firmware from a server workstation to flash memory. Unlike transferring operational code, Firm.ld is a small enough file that using the service port or PCMCIA modem is a practical approach.

Table 10-2. Loading Firmware				
Physical Attachment	**Line Protocol**	**Transfer Protocol**	**Tool**	**Default IP Addresses**
Service port + null modem Service port + ext modem PCMCIA modem	Async terminal	XMODEM	Firmware	Not applicable
PCMCIA EtherJet Ethernet LIC (10 Mbps) Token-Ring LIC	IP	TFTP	Op-code Firmware	Network Utility=10.1.0.2 Workstation=10.1.0.3

Procedures

Using Local Disk Copy
The firmware upgrade procedure using local disk copy is:

1. Access the firmware main menu using the procedures described in "Firmware" on page 5-19.

2. Make the following sequence of menu selections:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 7, "Update System Firmware"
   c. F/W Update Options: Option 3, "Use a Local Image File"

   The firmware asks for a local file name. Enter one of:

   > **c:\sys0\firm.ld** for Bank A
   > **c:\sys1\firm.ld** for Bank B

3. Respond "yes" to the question "Do you want to continue?" The firmware then starts writing the new firmware to flash memory.

4. Wait and do not turn off the system while the update proceeds.

5. On completion, press **Enter** to restart the system. The new firmware boots up to the current operational code.

Using XMODEM
The firmware upgrade procedure using XMODEM is:

1. Access the firmware main menu using the procedures described in "Firmware" on page 5-19.

2. Make the following sequence of menu selections:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 7, "Update System Firmware"
   c. F/W Update Options: Option 2, "XMODEM a Remote Image File"

   The firmware asks for a local file name. This is the name of a temporary file (for example, firmtemp.ld) on the hard disk. Do not give a path name. After you enter the file name, the firmware tells you when to start the file transfer.

3. Go to your terminal emulation package and start the transfer of the file "Firm.ld" from your workstation. It should be in the directory where you unpacked the .zip or .tar file you downloaded from the web.

4. After XMODEM completes, respond "yes" to the question "Do you want to continue?" at the firmware console. The firmware then starts writing the new firmware to flash memory.

5. Wait and do not turn off the system while the update proceeds.

6. On completion, press **Enter** to restart the system. The new firmware boots up to the current operational code.

Using TFTP

The firmware upgrade procedure using TFTP is:

1. Access the firmware main menu using the procedures described in "Firmware" on page 5-19.

2. Configure the IP addresses you will be using:

   Follow the menu sequence:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 11, "Remote Initial Program Load Setup"
   c. Network Parameters: Option 1, "IP Parameters"

   Set the following addresses:

   - Client IP address: an IP address for the Network Utility LAN card. This is a temporary address that need not be related to the router operational address for that interface.
   - Server IP address: the IP address of the workstation's LAN adapter
   - Gateway IP address: the IP address of any intermediate router, or repeat the workstation's IP address if there is none
   - Netmask: the mask for the subnet attached to the Network Utility LAN card

3. Initiate the transfer with the following sequence of menu selections:

   a. System Management Services (main menu): Option 4, "Utilities"
   b. System Management Utilities: Option 7, "Update System Firmware"
   c. F/W Update Options: Option 1, "TFTP a Remote Image File"

   Enter the following:

   - Local file name: choose a name for a temporary file to be stored in the root directory of the Network Utility hard disk. Do not give a path name. Use a file name extension of 3 characters or fewer.
   - Remote file name: the path and file name (which must be "Firm.ld") of the firmware load module on your workstation. It should be in the directory where you unpacked the .zip or .tar file you downloaded from the web.

   After you select the adapter and port the firmware should use, the router initiates the TFTP get operation.

4. After TFTP completes, respond "yes" to the question "Do you want to continue?" at the firmware console. The firmware then starts writing the new firmware to flash memory.

5. Wait and do not turn off the system while the update proceeds.

6. On completion, press **Enter** to restart the system. The new firmware boots up to the current operational code.

See the instructions on the Network Utility Code Downloads web page for more details on the above procedures and the errors you may encounter.

How to Call for Service and Support

If you bought your Network Utility from an IBM Business Partner or Reseller, contact that party to find out how to receive service and support.

If you bought your Network Utility from IBM, the following forms of assistance are available:

* Hardware or code problem service and support

  For telephone support:

  – Inside the U.S. - call 1-800-IBM-SERV
  – Outside the U.S. - contact your local IBM service representative for the phone number in your country

  Before you call, have the machine type, model, and serial number from the backplate of the Network Utility available. If you have a software problem, you may need to have a TFTP server and Internet connection available to transfer a memory dump from the Network Utility and send it to IBM support personnel.

  You can also access IBM Service and Support via the World Wide Web at:

  `http://www.networking.ibm.com/support`

  Select the Network Utility product to reach product technical hints, tips, FAQ's, and code updates. In addition, you can subscribe to receive notification of future code updates.

* Configuration help and how-to questions for initial installations

  – Inside the U.S. - call 1-800-IBM-SERV. This is a free service.
  – Outside the U.S. - contact your local IBM service representative. This service may not be free outside the U.S..

* Service and support contracts for network design, planning, or problem determination

  – Inside the U.S. - call 1-800-IBM-SERV
  – Outside the U.S. - contact your local IBM service representative

Configuration and Management Specifics

Chapter 11. Overview

This chapter is an introduction to the part of the book titled "Configuration and Management Specifics." It gives an overview of possible applications for Network Utility, and describes how the other chapters document some of these applications.

Major Network Utility Functions

Using IBM's Multiprotocol Access Services software technology, Network Utility supports a variety of networking functions. The Network Utility is specifically designed for CPU and memory-intensive functions at network positions requiring a small number of physical interfaces.

Key applications of Network Utility by model include:

- Model TN1 - Network Utility TN3270E Server

  - TN3270E Server

    This function provides SNA host application access to IP desktop users.

    One or more Network Utilities can be positioned at a regional office or host data center, to provide access for medium to large numbers of TN3270 clients distributed throughout an IP network.

  Network Utility model TN1 also supports all the functions of model TX1.

- Model TX1 - Network Utility Transport

  - Data Link Switching (DLSw)

    DLSw provides native SNA end station (workstation, controller, FEP, or host) connectivity across IP backbone networks. It also performs DLC type conversion like that done in FRAD and X.25 PAD products.

    One or more Network Utilities can be positioned at a regional office or host data center, to terminate TCP connections from smaller DLSw routers in many branch offices.

  - Advanced Peer to Peer Networking (APPN)

    APPN provides native SNA end station (workstation, controller, FEP, or host) connectivity across SNA backbone networks. The *Enterprise Extender* feature allows this same connectivity across IP backbone networks.

    Network Utilities can be positioned wherever a high-capacity APPN network node is required. You can place one at the edge of an IP network to receive traffic from other Enterprise Extender products. A Network Utility could also provide extended border node function when connecting two different APPN networks.

  - Channel Gateway

    Network Utility supports both ESCON (fiber-optic cable) and Parallel Channel (bus and tag cable) adapters. Using one of these adapters, A Network Utility can serve as a gateway routing both SNA and IP traffic from a S/390 host to local LANs, an ATM network, or to a high-speed serial line.

  - Network Dispatcher

This function allows a number of IP-based application servers (for example, TN3270 servers, HTTP web servers, FTP servers) to present a single IP address appearance to client workstations on a intranet or on the Internet. The network dispatcher function fields TCP connection requests from these clients and routes them to an available server. It provides both load balancing among the servers, and high "logical server" availability by bypassing failed physical servers.

The Network Utility can be placed at a host data center in front of hosts providing these server functions, or in front of multiple Network Utility Model-TN1s that are providing TN3270E Server function.

– High-speed media conversion

Network Utility can serve as a high-speed bridge between interfaces on its supported adapters.

In this book, we have selected a key subset of the above functions for expanded discussion and example configurations. The chapters that follow cover:

* TN3270E Server, optionally with Network Dispatcher in front of multiple servers

* Channel Gateway, for both SNA and IP traffic

* Data Link Switching, with both TCP termination and local DLC conversion

For help in understanding and configuring Network Utility functions other than these, consult the core software publications:

* *MAS Protocol Configuration and Monitoring Reference Volume 1*
* *MAS Protocol Configuration and Monitoring Reference Volume 2*
* *MAS Software Users Guide*

You may also find configuration help in the following IBM Redbooks. Although they are specific to the IBM 2216 Model 400, some of the configuration scenarios may apply.

* *IBM 2216 Nways Multiaccess Connector Description and Configuration Scenarios Volume 1* (SG24-4957)
* *IBM 2210 Nways Multiprotocol Router and IBM 2216 Nways Multiaccess Connector Description and Configuration Scenarios Volume 2* (SG24-4956)

Chapter Layout and Conventions

Chapters 12 through 18 of this book are organized as follows.

Chapter Layout

Each of the three key functions (TN3270E Server, Channel Gateway, and Data Link Switching) is covered by two chapters:

* An introductory chapter that:

  – Summarizes the supported function
  – Discusses example network configurations
  – Introduces how to manage the function

* An "Example Configuration Details" chapter containing:

  – Labelled diagrams of key example configurations

– Matching tables with configuration parameters for both Configuration Program users and command line users

The configurations shown and described in tables are actual working configurations. Binary configuration files matching these tables are downloadable from the World Wide Web. To access these files, follow the `Downloads` link from:

> `http://www.networking.ibm.com/networkutility`

In addition, Chapter 18, "Sample Host Definitions" provides detailed examples for configuring IBM host software products to match Network Utility configurations.

Example Configuration Table Conventions

The configuration parameter tables used in the three "Example Configuration" chapters all follow the same format. Table columns and conventions are as follows:

Configuration Program Navigation
> The sequence of folder and panel names to follow until you reach the panel where you enter parameter values.

Configuration Program Values
> Parameter names and their values.
>
> If the configuration program panel shows parameters not listed in the table, we used their default values. *Your configuration must be for a Network Utility and not a 2216-400, to have the correct default values.*

Command-Line Commands
> The commands you type to configure the same parameters using the command line interface, as follows:
>
> - Command sequences start from the talk 6 prompt `Config>`. Where needed, the initial command shows how to get to the right place in the menu system and the resulting command prompt.
>
> - Commands without parameters specified will either ask for the input values, or have no parameters. Parameter prompts from the system are shown in `this font`.
>
> - Where the value prompts and values you type are self-explanatory, the details are not shown.
>
> - "(Accept other defaults)" means that there are other parameter prompts you should accept the default values for (by pressing **Enter**).

Notes
> Numbers that reference comments at the bottom of each table.

Chapter 12. TN3270E Server

Overview

This section introduces TN3270 and summarizes the TN3270E server function implemented in Network Utility.

What is TN3270?

Many companies today are considering the consolidation of their WAN traffic onto a single IP-only backbone. At the same time, other companies are simplifying their workstation configuration and attempting to run only the TCP/IP protocol stack at the desktop. However, most of these companies still require access to SNA application hosts.

TN3270 meets these requirements by allowing you to run IP from the desktop over the network and attach to your SNA host through a TN3270 server. The clients connect to the server using a TCP connection. The server provides a gateway function for the downstream TN3270 clients by mapping the client sessions to SNA dependent LU-LU sessions that the server maintains with the SNA host. The TN3270 server handles the conversion between the TN3270 data stream and a SNA 3270 data stream.

To deploy a TN3270 solution, you install TN3270 client software on desktop workstations[1] and TN3270 server software in one of several places discussed below. Client software is available from IBM and many other vendors, and runs on top of the TCP/IP stack in the workstation. A given client product provides one of two possible levels of standards support:

- Base TN3270 client

  These clients conform to RFC 1576 (TN3270 Current Practices) and/or RFC 1646 (TN3270 Extensions for LU name and Printer Selection).

- TN3270E client

  These clients are compliant with RFC 1647 (TN3270 Enhancements), and RFC 2355 (TN3270 Enhancements).

A server implementation that can support TN3270E clients is called a TN3270E server.

Placement of the TN3270 Server Function

The TN3270 server function can be placed in a variety of products and positions within a network, including:

- In the SNA host itself

  IBM and several other vendors provide host TN3270 server software that sits on top of the host TCP/IP stack, and connects within the host to VTAM.

- In a router or Network Utility in the network

---

[1] You can also find small, dedicated TN3270 client products that represent printers.

IBM and other vendors provide TN3270 server function in networking hardware products. You can place these products directly adjacent to the SNA host, or at any position in the network where you have SNA connectivity to the host. If you are using IBM routers (2210 or 2216) or Network Utilities, and your host is running APPN, you can use Enterprise Extender technology to place the server at any position where you have IP connectivity to the host.

- In a software product in the network

IBM and other vendors provide TN3270 server software products that you install on mid-range servers using operating systems such as AIX, OS/2, or Windows/NT. You can place these products at any position in the network where you have SNA connectivity to the application host.

The choice of TN3270 server product and network position is a complex one, involving such factors as:

- Host capacity and cycle impact
- Price for performance and capacity
- Availability
- Impact of server failure
- Scalability

Network Utility provides a high-performing TN3270E server implementation that scales to large networks. By combining it with the Network Dispatcher feature, you can implement server redundancy and load sharing in large TN3270 installations. You can also place a Network Utility out into an SNA or IP network away from the data center, and get the same advantages of scalability, incremental addition, and reduced impact of server failure.

Network Utility TN3270E Server Function

Standards Compliance
The Network Utility implementation of TN3270E server supports these RFCs:

RFC 1576 - TN3270 Current Practices
RFC 1646 - TN3270 Extensions for LU names and Printers
RFC 1647 - TN3270 Enhancements
RFC 2355 - TN3270 Enhancements (obsoletes RFC 1647)

It can handle both base TN3270 and TN3270E clients, at the same time.

Host Connectivity
As mentioned above, the path from a TN3270 client to the SNA host consists of two pieces:

- A TCP connection over IP from the client to the server
- An SNA LU-LU session from the server to the host

The form of the SNA connection from the server to the host depends on how the server represents PUs and dependent LUs. When you are using Network Utility as your TN3270 server, you can configure either of two different ways to establish links and represent PUs and LUs to VTAM:

- Using SNA subarea links

You set up Network Utility this way when you are not running APPN at the host. You configure a separate DLC-layer link to the host for every PU

(maximum of 253 LUs). Multiple PUs require multiple parallel host links. SNA frames arriving at Network Utility on one of these links flow directly to the corresponding internal PU.

Subarea host links must be a single DLC-layer hop to the host. They can traverse bridges or other DLC-layer forwarding mechanisms (such as protocol converters or external DLSw routers). Network Utility supports the following link types for subarea host attachment:

  – Token-ring: physical, ATM LAN emulation, or channel LSA
  – Ethernet: physical, ATM LAN emulation, or channel LSA
  – FDDI: physical only
  – Frame relay: routed RFC 1490 format only

* Using an APPN Dependent LU Requester (DLUR) link

You set up Network Utility this way when you are running APPN with its Dependent LU Server (DLUS) function at the host. You configure one DLC-layer link to the host to carry the DLUR-DLUS "pipe", even if you are defining multiple local PUs. SNA frames arriving at Network Utility on this link flow to the DLUR function, which redirects them to the correct internal PU.

When you are using DLUR, you can route through an APPN network using either ISR or HPR routing to reach the host. Network Utility supports the following link types as the "first hop" APPN link to the host:

  – Token-ring: physical, ATM LAN emulation, or channel LSA
  – Ethernet: physical, ATM LAN emulation, or channel LSA
  – FDDI: physical only
  – Frame relay: bridged or routed RFC 1490 formats
  – ATM (native, not LAN emulation): HPR only
  – Channel MPC+: HPR only
  – PPP
  – SDLC: ISR only
  – X.25: ISR only
  – DLSw: ISR only
  – IP (Enterprise Extender): HPR only

Note especially that when using DLUR and HPR routing, you can place a Network Utility TN3270E server across an IP network from the SNA application host. Enterprise Extender maintains session-level class of service and transmission priority across the IP network.

General TN3270E Server Configuration

This section covers general information about configuring Network Utility TN3270 server support, before specific example configurations begin on page 12-5.

Configuring TN3270 Subarea under the APPN Protocol

In the Network Utility implementation of TN3270 server, all SNA functions are bundled within the APPN protocol. This means that *even when you are configuring SNA subarea attachment and your SNA host is not running APPN*, you must use the configuration and console services of the APPN protocol. In particular:

* You must go through the APPN protocol at the command line and the Configuration Program to configure ports, links, and TN3270 server functions

- You must go through the APPN protocol at the command line to use TN3270 monitoring commands
- You must still configure APPN at the node level

When you configure SNA subarea support, Network Utility does in fact still function as an APPN network node, but only on links to other APPN nodes. If the *only* ports and links you configure are those for SNA subarea host attachment, then the APPN function is rendered moot.

Configuring in the APPN Environment

APPN and TN3270 server are fully configurable both from the Configuration Program and from the command line. From the Configuration Program, the TN3270 configuration parameters are always available. If you create a TN3270 configuration and download it to a Network Utility model TX1, which does not support TN3270 server function, the Network Utility ignores the TN3270 part of the configuration. If you are working from the command line on a model TX1, the commands for configuring and monitoring TN3270 simply do not appear on the APPN menus.

To change an APPN/TN3270 configuration from the Configuration Program, you make the change, transfer the configuration to the Network Utility, and reboot for it to take effect.

To change an APPN/TN3270 configuration from the command line, you move to talk 6, type **p appn**, then issue the commands to make the change. You have two options for activating the change:

- Write the configuration to disk and reboot Network Utility to activate it

- Issue the talk 6 APPN **activate** command to dynamically activate the modified APPN/TN3270 configuration

  Depending on the configuration items you changed, APPN either makes the change immediately, or restarts APPN (but not the entire Network Utility) to activate the change. For the latter case, if you move to talk 5 and type **p appn** while APPN is restarting, you get the message APPN is not currently active. You can poll with talk 5 commands to see when the restart is complete.

  You can recycle the entire TN3270 server function in this way, by disabling and enabling it with the talk 6 command **set tn**, and activating each of these configuration changes dynamically.

Implicit and Explicit LU Naming and Mapping

When you configure Network Utility's TN3270 server function, you create a local LU name for every one of the potential concurrent client sessions the Network Utility is intended to support. The LU name you define in the Network Utility need not have any relation to LU names in VTAM.

When a TN3270 client connects to a server over TCP, it can request a specific LU name, or it can place a generic request for any LU of a certain type. If you are configuring a client to request a specific name, you specify one of the local names defined at the server (Network Utility), not a VTAM LU name.

Because a single Network Utility may support thousands of LUs with similar characteristics, it does not require you to individually configure each LU. Rather,

you can create a large pool of *implicit* LUs to satisfy clients that don't request a particular LU name. You then add a small number of *explicit* LUs to satisfy the clients that do request a particular name.[2]

As you will see in the example configurations, you define implicit LUs in groups as you define each local PU. You specify an LU name mask and number of LUs, but no NAU address. To configure an explicit LU, you specify an LU name and an NAU address (2-254). When the Network Utility activates the configuration, it reserves the NAU addresses for explicit LUs, then generates names for the implicit LUs using the group name mask and one of the available NAU addresses.

When a TN3270 client connects in and does not explicitly request an LU name, Network Utility attaches the client to any available implicit LU. At this point, the server function treats all the implicit LUs as being in one large pool without regard to PU boundaries.

Example Configurations

Network Utility as a TN3270E server can be deployed in several configurations. For example, it can be placed either in the remote branch or in the data center. It can attach to the host via a traditional SNA subarea connection or it can use APPN. In the data center, it can be placed in a channel-attached configuration or it can it can be a stand-alone server that resides on the campus LAN (or ATM cloud) using the channel-attached connection provided by an existing 3745/46, 2216-400, 3172, an OSA adapter, or an OEM gateway.

One of the most important elements of a TN3270 implementation is scalability. The Network Utility solution can scale to very large configurations while providing high availability and redundancy.

The following scenarios show you how to effectively utilize the Network Utility as a TN3270e Server.

TN3270 via a Subarea Connection to an NCP

This scenario (shown Figure 12-1 on page 12-6) shows a traditional SNA subarea network with all host access occurring through an IBM 3745/46 Front End Processor (FEP) with the IBM Network Control Program (NCP). The Network Utility is installed to provide TN3270 server support for downstream workstations both in the local campus and in the remote sites. The Network Utility attaches to the host through the FEP via a normal subarea connection.

Up to 9,000 TN3270 sessions can be handled with a single Network Utility installed as shown in the diagram. As your network grows, the solution can be scaled simply by adding more TN3270e server capacity via additional Network Utilities. You can also set up automatic load balancing among your TN3270E servers by installing a separate IBM router or Network Utility to serve as a Network Dispatcher. (See "Highly Scalable, Fault Tolerant TN3270e" on page 12-9 for an example of how to scale the network.)

---

[2] The implicit/explicit distinction is solely within the Network Utility. A client can request an implicit LU name, and the Network Utility will satisfy the request if the LU is available. The key point is that the server function will never assign an explicit LU to a client unless the client specifically requests that LU name.
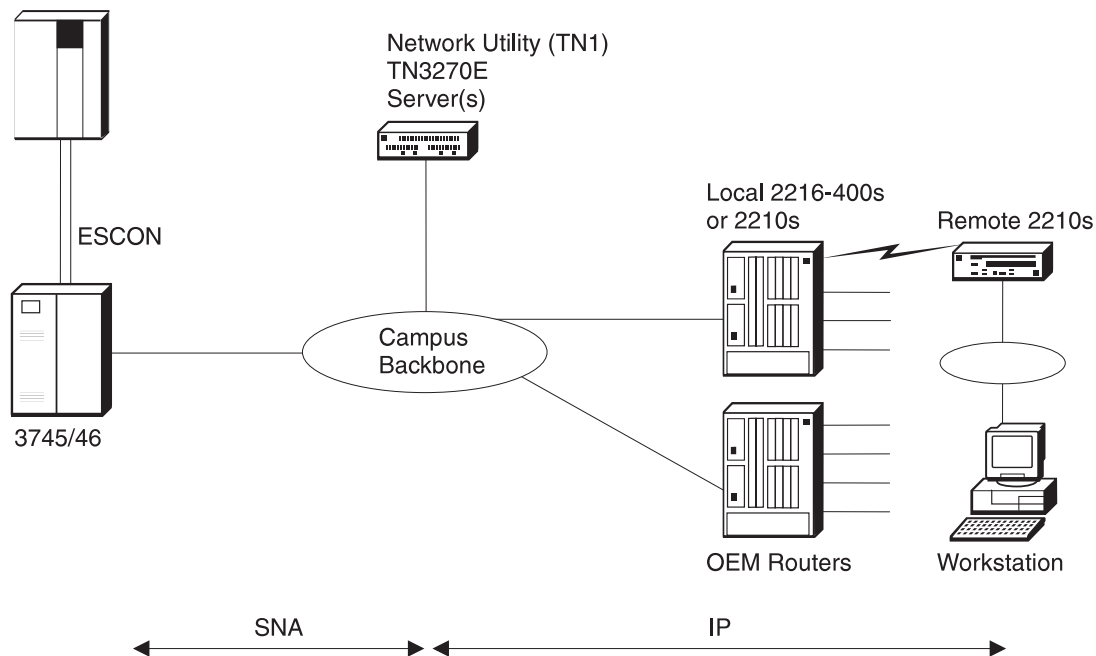
Figure 12-1. TN3270 via a Subarea Connection through a 37xx

Keys to Configuration

The configuration of the TN3270e server function is very straight forward in this scenario. However, the following points are worth noting:

- There is both an APPN and a subarea implementation of the TN3270E server. Both require APPN support to be installed on the Network Utility and both are configured within the APPN configuration process. This is true even though a pure subarea configuration does not use the APPN function. This is an implementation statement as the TN3270e server function uses the APPN SNA stack for both subarea and APPN connections to the host.

  Please also note these additional points relating to APPN and TN3270e Server configuration.

  – APPN support must be enabled.

  – You must define a port and one or more link stations to define the connection to VTAM.

  – For subarea configurations, defining a link station and specifying to solicit an SSCP session implicitly defines a PU on the Network Utility. This PU will support up to 253 downstream LUs. If you need more than 253 LUs, then you need to define more than one link station. Each link station needs to use a different Service Access Point (SAP) and a different Local Node ID (IDNUM).

- When configuring the parameters for the TN3270e server, the IP address of the server can be set either to the internal box IP address or to one of the interface

IP addresses. Keep in mind that the address you select for TN3270 may be unavailable for using normal IP telnet to manage the box.[3]

- The downstream LUs can be defined either as explicit or implicit.

  – Use explicit definitions when you need to ensure that the device will always use the same LU name. (For example, printers would normally use explicit definitions.)

  – Use implicit definitions when you have a large group of devices that can use a common pool of available LUs and do not need to use the same LU name every time.

For a complete look at the configuration parameters needed for this scenario, see Table 13-2 on page 13-3.

TN3270 via a Subarea Connection through a Channel Gateway

This scenario, shown in Figure 12-2 on page 12-8, is similar to the previous scenario except that here, the Network Utility attaches to the host through a LAN channel gateway such as an IBM 3172, an IBM 2216, an IBM 3746 with the Multi-Access Enclosure (MAE) or an OEM device. These gateways use External Communications Adapter (XCA) pass-through and do not provide the SNA boundary function normally provided by an NCP. With a gateway, this function is provided by VTAM.

If you have an existing gateway with a TN3270 server configured, you can use the Network Utility to offload the existing TN3270 workload or to provide additional TN3270 capacity as your network requirements grow.

An existing 2216 or a 3746 allows you to have multiple channel connections to the host while you can incrementally install Network Utilities for your TN3270e server requirements. The dynamic load balancing features of network dispatcher can be used to optimize efficiency.

---

3 If you need to use telnet at this same address, you can configure the TN3270e server to use another port (24 for example) so that telnet can use port number 23. This requires that the TN3270 client workstations be configured to use this same port.
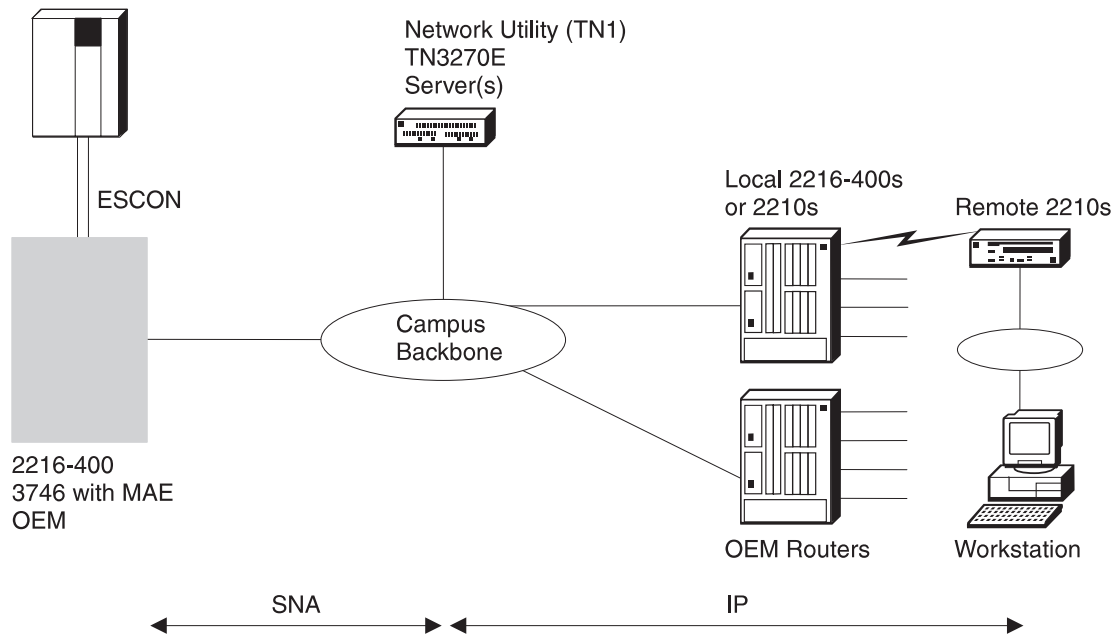
Figure 12-2. TN3270 via a Subarea Connection through a LAN Gateway

Keys to Configuration

From the Network Utility perspective, the configuration of this scenario is identical to the previous one. The host definitions are also identical. For both scenarios, you just have to define the switched major nodes for the PUs in the TN3270e server.

TN3270 through an OSA Adapter

This scenario is shown in Figure 12-3 on page 12-9. Here, the Network Utility attaches to the host through the S/390 Open Systems Adapter (OSA). Like the previous gateway scenario, the SNA boundary function is in the host.

While the TN3270 server function can reside on the host itself, many customers prefer to offload this function externally to another platform. The Network Utility meets this requirement well by providing scalable, cost effective TN3270e server function without changing your method of host attachment. This allows you to leverage your existing investments.
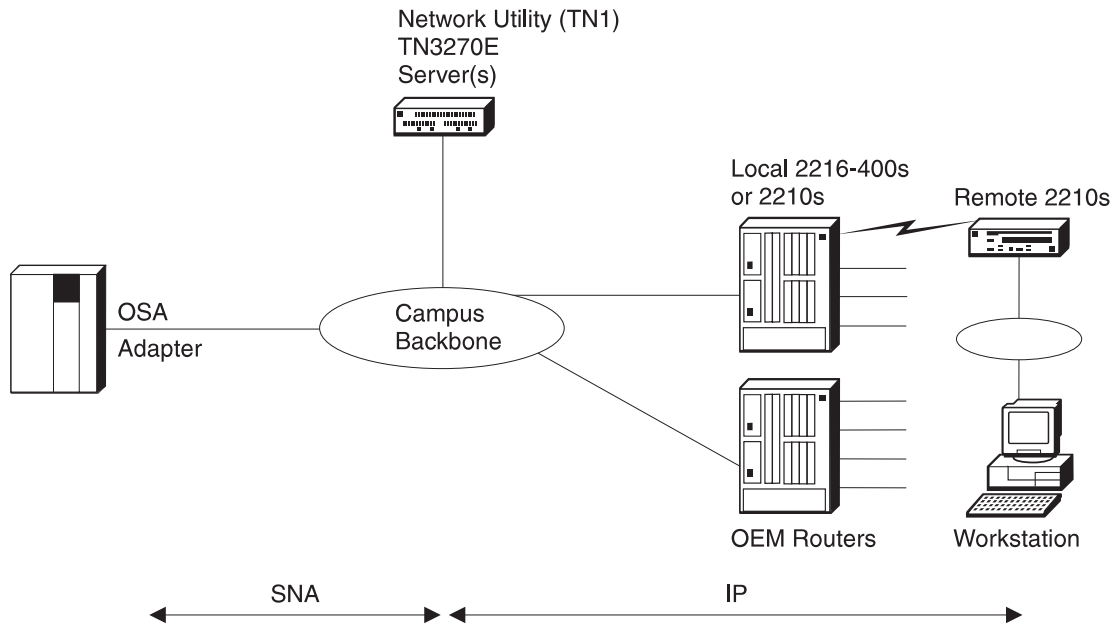
Figure 12-3. TN3270 via an OSA adapter

Keys to Configuration
From the Network Utility perspective, the configuration of this scenario is identical to the previous two.

Highly Scalable, Fault Tolerant TN3270e

This scenario, shown in Figure 12-4 on page 12-10, is an extension of the one discussed in "TN3270 via a Subarea Connection to an NCP" on page 12-5. Here, the solution is scaled with multiple Network Utility devices to provide TN3270e server support for large 3270 environments. Also, a separate Network Utility is configured as a network dispatcher and deployed to provide load balancing. The new Network Dispatcher Advisor for TN3270 allows the Network Dispatcher to collect load statistics from each Network Utility TN3270e server in real time to achieve the best possible distribution among the TN3270 servers.

The solution provides high availability in the event of a failure in one of the TN3270e servers. The server that the client session is dispatched to is transparent to the user. If a failure occurs, the sessions through that server are lost but the users simply log back on to the host through another Network Utility using the same destination IP address for the TN3270e server.

The Network Dispatcher function also utilizes redundant hardware with a second Network Utility configured as a Network Dispatcher and serving as a backup to the primary one.

With this configuration, you can scale your TN3270e support to any size simply by adding additional TN3270e server capacity. You can do this incrementally and non-disruptively as your network requirements grow.
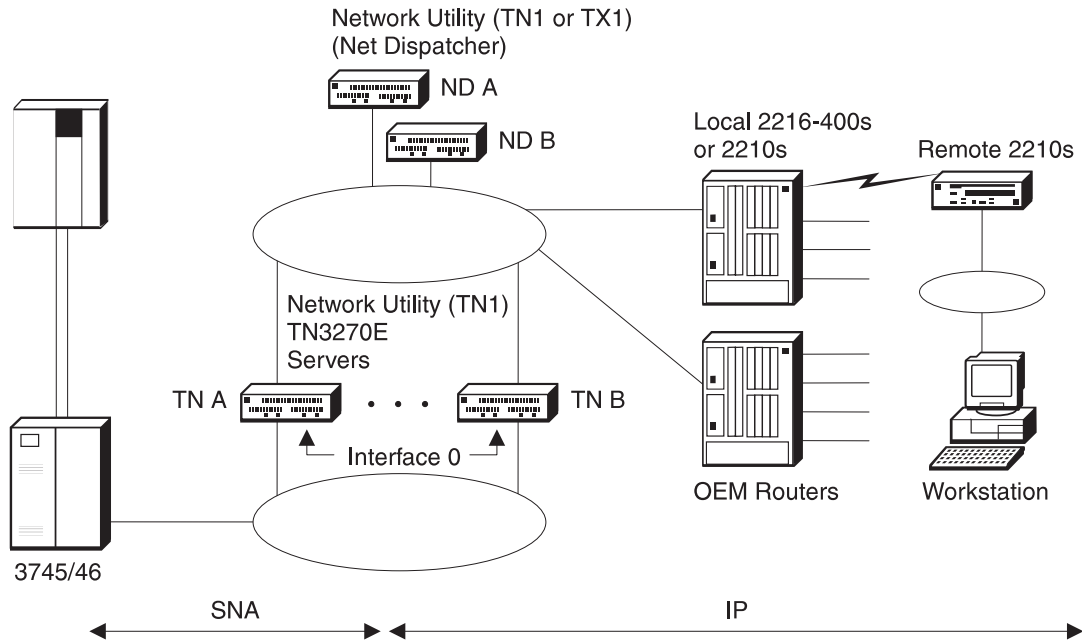
Figure 12-4. Highly Scalable, Fault Tolerant TN3270e

Keys to Configuration

As far as the TN3270e server is concerned, the configuration is the same whether you have a Network Dispatcher or not. In fact, the TN3270e server is unaware that the traffic from the clients is being dispatched through another machine. Please see "TN3270 via a Subarea Connection to an NCP" on page 12-5 for the basic configuration points for a TN3270e server. See Table 13-3 on page 13-9 for the complete set of configuration parameters for the TN3270e servers for this scenario.

However, the IP addressing needs special attention in this configuration for high availability. In "TN3270 via a Subarea Connection to an NCP" on page 12-5, the TN3270e server was configured with the same address as the router ID (also the same address as the LAN interface). In a Network Dispatcher environment, the IP addressing is somewhat different.

A Network Dispatcher and one or more TN3270e Servers form what is called a cluster. An IP address is defined for the cluster and workstations send their TN3270 packets to this IP address. The Network Dispatcher receives these packets and forwards them on to a server in the cluster for processing.

Since the Network Dispatcher does not alter the destination IP address of these packets, each TN3270e server also needs to be configured with this same IP address. However, you have to make sure that the TN3270e servers do not broadcast this address via OSPF or RIP to the network because you don't want these servers to respond to the cluster address. Only the Network Dispatcher should respond to the cluster address.[4]

---

[4] The cluster address can not be pinged. The Network Dispatcher does not respond to pings to the cluster address. It only processes telnet and FTP packets.

The router must know the TN3270e server's IP address in order to forward packets to the server function. One way to make this address known to the router is to specify it to an interface as a secondary address. Figure 12-5 on page 12-11 shows an example of this IP addressing scheme for the highly available, fault tolerant TN3270 configuration depicted in Figure 12-4 on page 12-10.

```
   TN3270e Server #1 (TNA):
        Internal address  172.128.252.3
        Interface 0       172.128.2.3      (2nd address: 172.128.1.100)
        Interface 1       172.128.1.3
        OSPF Router ID    172.128.1.3
        TN3270e Server    172.128.1.100    (same as cluster address)

   TN3270e Server #2 (TNB):
        Internal address  172.128.252.4
        Interface 0       172.128.2.4      (2nd address: 172.128.1.100)
        Interface 1       172.128.1.4
        OSPF Router ID    172.128.1.4
        TN3270e Server    172.128.1.100    (same as cluster address)

   Network Dispatcher #1 (NDA):
        Internal address  172.128.252.1
        Interface 0 addrs 172.128.1.1
        OSPF Router ID    172.128.1.1
        Cluster address   172.128.1.100
           Port 23
              Server 1    172.128.1.3
              Server 2    172.128.1.4

   Network Dispatcher #2 (NDB):
        Internal address  172.128.252.2
        Interface 0 addrs 172.128.1.2
        OSPF Router ID    172.128.1.2
        Cluster address   172.128.1.100
           Port 23
              Server 1    172.128.1.3
              Server 2    172.128.1.4
```

Figure 12-5. IP addressing for the Highly available, Fault Tolerant TN3270 Scenario

Note that the cluster address is assigned as a second IP address on interface 0 of the Network Utility machines. In this scenario, the LAN segment that interface 0 attaches to does not carry any IP traffic - only the SNA subarea traffic from the TN3270e server to the host.

The configuration of the Network Dispatchers is standard. The complete set of configuration parameters needed for this scenario can be found in Table 13-4 on page 13-14 for the primary network dispatcher. Differences from this configuration for the backup network dispatcher can be found in Table 13-5 on page 13-18.

Explicit LUs and Network Dispatcher

Special care has to be taken for explicit LU definition in a Network Dispatcher environment. A session request for either a implicit or a explicit LU can be dispatched to any server. This means that the explicit LU has to be defined in each server, since it is not known in advance to which server the session will be dispatched. The explicit LU in this environment, a printer, for example, is represented by two different LUs in VTAM. The PUs in the TN3270e servers that have the LU defined each have to have an unique Node ID (IDNUM), because VTAM does not allow duplicate PU or LU names to be active at the same time.

When a server has both explicit LUs and implicit LU pool(s) defined, if all the sessions of the pool are used, the server can not handle any more requests for sessions from this pool. But, the ND still dispatches sessions to this server, because this server does not report 100% load.

One way to handle explicit LUs is to define them on a separate TN3270e server that is outside the network dispatcher environment. This is normally acceptable because there are many fewer of them and hence do not require the dynamic load balancing feature of network dispatcher Also, devices that use explicit definitions (such as printers) often have a much lower acceptable availability requirement even in an otherwise fault-tolerant environment.

TN3270 Via DLUR over APPN

This scenario, shown in Figure 12-6 on page 12-13, uses APPN to communicate with the host. The Network Utility uses APPN High Performance Routing (HPR) and establishes a Rapid Transport Protocol (RTP) session with the host. HPR is used all the way from the TN3270e server to VTAM. In case of a failure, this assures nondisruptive session switching to an alternate path if you have parallel gateways. This is especially important in Parallel Sysplex environments.

In addition, HPR is supported over IP through the Enterprise Extender feature of the Network Utility. This is important if you want to place your TN3270e server in a remote location and use IP to transport the APPN traffic back to your data center.

The channel gateway is an APPN network node performing APPN Automatic Network routing (ANR) for the RTP session between the Network Utility and the host.
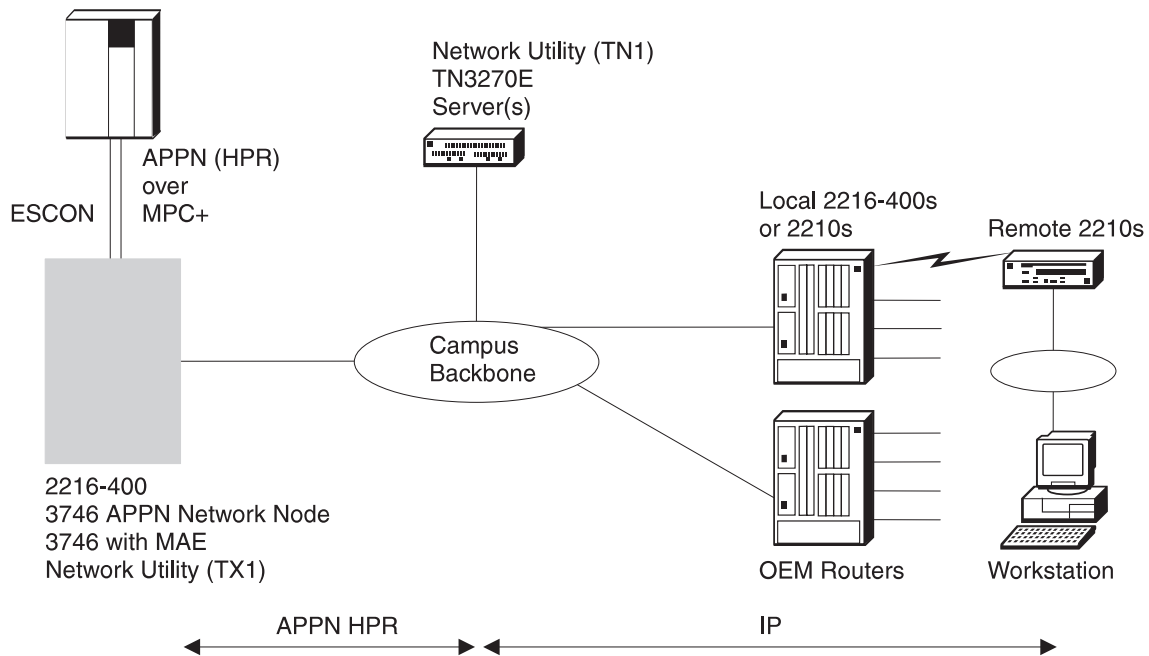
Figure 12-6. TN3270 Via DLUR over APPN

When connecting a TN3270e server to the host via APPN, you must configure
DLUR support on the Network Utility. The DLUR feature extends to APPN nodes
the support of T2.0 or T2.1 devices containing dependent LUs. The DLUR function
on an APPN network node works in conjunction with a dependent LU server
(DLUS). The DLUS function is usually provided by VTAM, although it may reside in
any part of a mixed APPN/subarea network.

The dependent LU flows (SSCP-PU and SSCP-LU) are encapsulated in a LU 6.2
pipe (CP-SVR) established between the DLUR APPN node and the DLUS SSCP.
The CP-SVR pipe is made up of a pair of LU 6.2 sessions using a new
CPSVRMGR mode between the DLUR and the DLUS. This pipe brings the SSCP
function (in the DLUS) to the DLUR APPN node where it can be made available to
attach T2.0/T2.1 nodes containing dependent LUs.

Keys to Configuration
From a downstream workstation perspective, the TN3270e server appears the
same whether that server is using SNA subarea or APPN to communicate with the
host on the uplink. At the Network Utility, you configure the base TN3270 server
parameters the same way as in the SNA subarea scenarios, but the way you
configure the local PUs differs. Instead of associating each PU with a subarea link,
you configure local PUs without any link association. The DLUR function is
responsible for routing traffic on the DLUS-DLUR pipe to and from these local PUs.

APPN requires Dependent LU Requester (DLUR) support to be configured in the
Network Utility. DLUR is quite simple to configure with the only required parameter
being the CP name of the Dependent LU Server (DLUS), which is VTAM.

You have to make some additional host definitions for APPN and DLUR support.
See Chapter 18, "Sample Host Definitions" on page 18-1 for an example of these
commands.

For a complete look at the configuration parameters needed for this scenario, see Table 13-6 on page 13-20.

Distributed TN3270e Server

The previous configurations showed how the Network Utility can be deployed in the data center to centralize the TN3270e server function in your network. This configuration, shown in Figure 12-7, shows just one example of how the Network Utility can also be placed in a remote location to provide distributed TN3270e server capability.

In this configuration, the Network Utility is providing TN3270e server service to workstations in the remote location. As always with a TN3270 configuration, the workstations are using IP to communicate with the TN3270e server. The TN3270e server is using Dependent LU Requester (DLUR) over an APPN connection back to the host in the data center.

In this example, the corporate WAN is a public frame relay network which carries IP traffic only. Therefore, the Network Utility is configured to use the Enterprise Extender feature which allows the APPN HPR traffic to be carried over the IP-only WAN.

The Enterprise Extender traffic is terminated at the host gateway which decapsulates the HPR traffic and then passes the APPN traffic through the network node onto the MPC+ path to the host. This is a very fast, low-overhead packet forwarding function, so a single gateway can handle a large amount of traffic.
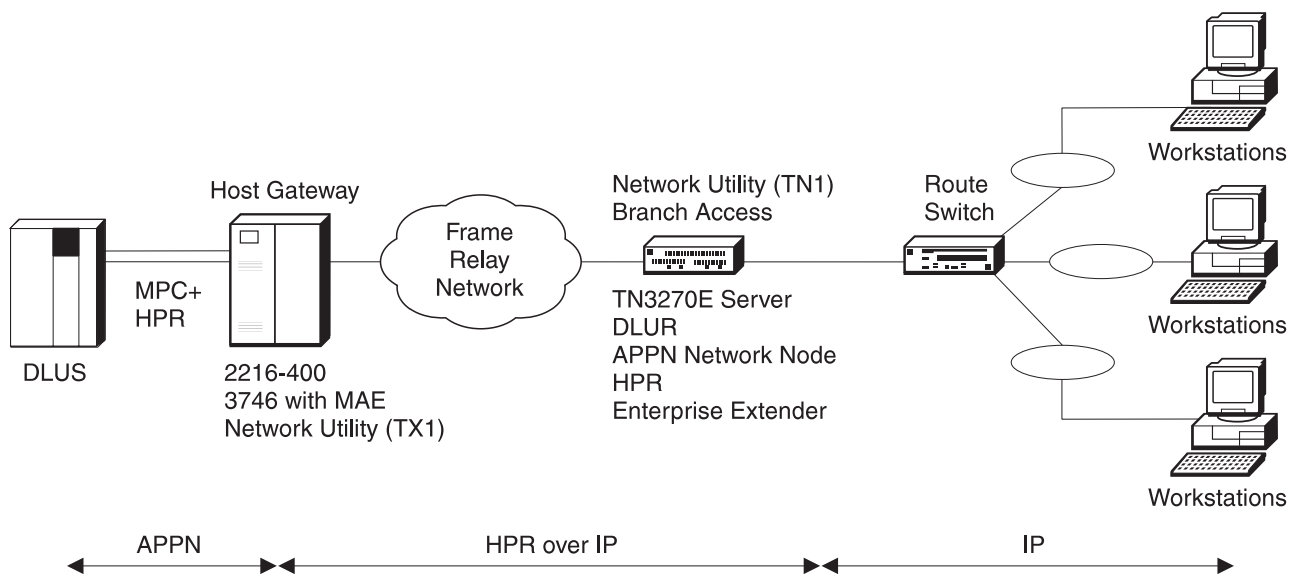
Figure 12-7. Distributed TN3270e Server

Keys to Configuration

From a downstream workstation perspective, the TN3270e server appears the same whether it is in the remote branch or in the data center regardless of whether the upstream connection to the host is using SNA subarea or APPN. Therefore, the TN3270e server function in the Network Utility is configured exactly the same as in the previous scenarios.

APPN and DLUR is configured the same as in "TN3270 Via DLUR over APPN" on page 12-12 with one exception which is the port definition for APPN over the frame relay IP link. When configuring APPN to use HPR over IP (the Enterprise Extender feature), you specify a port type of IP. Then when adding the link station for this port, instead of specifying the MAC address of the adjacent FEP as was done in "TN3270 Via DLUR over APPN" on page 12-12, you specify the IP address of the other end of the HPR over IP network, which is the host gateway in this example.[5] The IP network is responsible for the delivery of the traffic to the host gateway over the best path available. You are assured a reliable transport since the connection between the TN3270e Server and the host uses an RTP session.

Managing the TN3270E Server

This section introduces some of the ways you can monitor and manage the TN3270E server function.

Command-Line Monitoring

To view currently running TN3270 server status from the command line, move first to talk 5 and enter **p appn**. If you get the message *Protocol APPN is available but not configured*, you need to complete your base APPN configuration and reboot Network Utility to activate APPN. As discussed in "Configuring TN3270 Subarea under the APPN Protocol" on page 12-3, you need APPN to be active even if you are using only TN3270 subarea connectivity.

The key command for viewing TN3270-specific status is **list tn3270**, or just **li tn**. When you type this command with no parameters, it displays general status and configuration information about the server function. When you type this command followed on the same line by a client IP address, it displays detailed status information about that particular client.

The following list summarizes the possible outputs of this command:

- General status (**li tn**)

  If the system responds *TN3270E is not configured or not active*, you did not enable the TN3270 server function adequately in the currently active APPN configuration. If you get this error and you did configure the function, perhaps the TN3270 server IP address you chose is not active as an interface address or as the internal IP address. Consult the example TN3270 configurations in Chapter 13 for other possible reasons, then change your APPN/TN3270 configuration and activate it as described in "Configuring in the APPN Environment" on page 12-4.

  If the server function is active, it provides the following information:

  - Configuration information currently in use

    TN3270E IP Address
      The server IP address to which clients connect, also the cluster address if you are using Network Dispatcher
    TN3270E Port Number
      The TCP port to which clients connect

---

[5] The host gateway must also be configured with an HPR over IP port in much the same manner as is described here.

NetDisp Advisor Port Number
> The TCP port to which Network Dispatchers may connect to retrieve load information

Keepalive type
> Whether and how the server polls clients to see if they are still active. Possible values are:
>
> **None**
> > Server does not poll clients, and will only discover client absence when trying to send data
>
> **NOP**
> > Server polls clients at the TCP level, client software need not have capability to respond
>
> **Timing mark**
> > Server polls clients at the TN3270 level, and client software must respond within a certain time window

Automatic Logoff
> Whether or not the server disconnects clients after a period of inactivity (with no data flowing in either direction)

– Summary statistics

Number of connections
> The current number of active TCP connections from TN3270 clients

Number of connections in SSCP-LU state
> The number of currently active client TCP connections that have an associated LU in this state (received an ACTLU but not yet a BIND)

Number of connections in LU-LU state
> The number of currently active client TCP connections that have an associated LU in this state (received BIND, fully active)

- Specific client status (**li tn** *client IP address*)

If the system responds *No connections found for this client ip address*, the client whose IP address you specified does not currently have a TCP connection with the Network Utility server.

If there is a TCP connection with this client, the system provides information under the column headers listed below. If a given piece of information is not available because of the state of this particular LU, the system displays a blank field.

Local LU
> The LU name, configured at Network Utility, to which the server function has mapped this client TCP connection

Class
> The type of LU, as follows:
> > **IW** Implicit workstation
> > **EW** Explicit workstation
> > **IP** Implicit printer
> > **EP** Explicit printer

Assoc LU
> For a workstation LU, the name of any associated printer LU

Client Addr
> The IP address of the client

Status
> Whether the connection is in SSCP-LU state or LU-LU state

Prim LU
> The primary LU name as known to VTAM

Sec LU
> The secondary LU name as known to VTAM

Idle Min
> The number of minutes since this connection carried any user data

Besides the **li tn** command, a TN3270 server user needs to be able to query the status of other APPN or SNA resources on which the function depends. The following commands are of general use:

li port - to show interface status

li link - to show the status of logical links

If you are using DLUR for your host connection, the following commands are particularly useful:

li appc - to check the status of the DLUS-DLUR pipe
li local - to show the status of internal PUs used by the TN3270 server function

Event Logging Support

In general, APPN/TN3270 ELS messages are intended to capture debug and trace information for IBM support personnel. These functions have extensive logging and trace support, but the ELS messages themselves are tightly packed with low-level information.

Normally, you activate APPN/TN3270 tracing and logging under the direction of IBM support personnel. The general procedure is to enable some of a large list of possible traces as part of your APPN configuration. From the Configuration Program, see the APPN Node Services folder. From talk 6, use the **set trace** command. After you activate this configuration change, the output of these traces flows into a trace table in APPN memory, and also to ELS if you have APPN ELS messages active. Should you have a problem that requires activating traces, IBM support will provide detailed procedures to guide you in capturing debug information.

SNA Management Support

APPN generates SNA alerts for a variety of error conditions, and can forward alerts from other SNA devices. This support is described in "SNA Alert Support" on page 8-6. There are no alerts specific to the TN3270 server function, but alerts that the Network Utility itself generates may relate to SNA resources involved with TN3270.

From a VTAM or NetView/390 operator console, you can control the links, PUs, and LUs involved with TN3270 as described in "NetView/390" on page 8-10.

SNMP MIB and Trap Support

Network Utility supports an Internet Draft version of both of the forthcoming standard MIBs for TN3270 server function:

TN3270 Base MIB
TN3270 Response Time MIB

Network Utility support for these MIBs includes the ability to:

- View server configuration, status, and statistics
- Set up client groups for response time collection
- Map LU names from VTAM name to local name to client IP address
- Collect response time data for current client groups

In addition, Network Utility supports the following IETF MIBs relating to APPN and SNA functions:

RFC 2155, APPN
RFC 2051, APPC
RFC 2232, DLUR
RFC 2238, HPR

RFC 1666, SNA NAU
Internet Draft, Extended Border Node

Network Utility supports the following IBM Enterprise Specific MIBs relating to APPN functions:

APPN Memory
APPN Accounting
APPN HPR NCL
APPN HPR Route Test
APPN Peripheral Access Node (Branch Extender)

These MIBs provide a comprehensive view of APPN and SNA resources within Network Utility, including those being used for TN3270.

Network Management Application Support

The Nways Manager products discussed in "IBM Nways Manager Products" on page 8-7 provide specialized statistical support for TN3270 response time monitoring, as well as the ability to view TN3270 server resources. To initiate response time monitoring, you select a group of one or more clients using an IP address and mask. For each group you define, the manager collects response time statistics into pre-defined time buckets (less than 1 second, 1-2 seconds, etc.). Using the collected information, you can view aggregate historical response time by group, or create custom reports that present the data in different graphical formats.

To view TN3270 resources and their status, you use specific panels that combine information from different tables within the base TN3270 MIB. To view APPN and SNA resources in general, you use specific panels that access information from the APPN MIBs. You can also use integrated browser support to view the information in any of these MIBs.

Nways Manager for AIX provides an APPN-level view of the topology of your network. You can discover participating APPN resources, view them, and view their status as color-coded icons. APPN protocol performance and error events (data and graph) are also provided. This application does not represent Branch Extender or Extended Border Node topologies.

Chapter 13. TN3270E Server Example Configuration Details

This chapter contains diagrams and configuration parameter tables for several of the example TN3270E server network configurations in Chapter 12, "TN3270E Server." The parameter values shown are from real working test configurations.

For an explanation of the columns and conventions in the configuration parameter tables, see "Example Configuration Table Conventions" on page 11-3.

The Network Utility World Wide Web pages contain binary configuration files that match these configuration parameter tables. To access these files, follow the `Download` link from:

> `http://www.networking.ibm.com/networkutility`

The configurations documented in this chapter are:

Table 13-1. Cross-Reference of Example Configuration Information	
Configuration Description	**Parameter Table**
"TN3270 via a Subarea Connection to an NCP" on page 12-5	Table 13-2 on page 13-3
"Highly Scalable, Fault Tolerant TN3270e" on page 12-9, for the TN3270 server TN A	Table 13-3 on page 13-9
"Highly Scalable, Fault Tolerant TN3270e" on page 12-9, for the Network Dispatcher ND A	Table 13-4 on page 13-14
"Highly Scalable, Fault Tolerant TN3270e" on page 12-9, differences for the Network Dispatcher ND B	Table 13-5 on page 13-18
"TN3270 Via DLUR over APPN" on page 12-12	Table 13-6 on page 13-20

Figure 13-1. TN3270 Subarea

Table 13-2 (Page 1 of 5). TN3270 Subarea. See page 12-5 for a description and 13-2 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Devices
  Adapters
    Slots
	Slot1: 2 port TR	See "add dev" on next row	1

Devices
  Adapters
    Ports
	Slot 1/Port 1: Interface 0: TR	`Config>`**`add dev tok`**	2

Devices
  Interfaces
	Interface 0
	  Mac Address 400022AA0001
		`Config>`**`net 0`**
		  `set phy 40:00:22:AA:00:01`	

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System
  SNMP Config
    General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

System
  SNMP Config
    Communities
      General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`Config>`**`p snmp`**
		`SNMP Config>`
		  `add community`
		  `set comm access write`
			3

Protocols
  IP
    General
	Internal address: 172.128.252.2
	Router ID: 172.128.1.2
		`Config>`**`p ip`**
		`IP Config>`
		  `set internal 172.128.252.2`
		  `set router-id 172.128.1.2`	

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.2
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP Config>`**`add address`**	

Protocols
  IP
    OSPF
      General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**
		
		  (Accept other defaults)	

Protocols
  IP
    OSPF
      Area Configuration
        General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Table 13-2 (Page 2 of 5). TN3270 Subarea. See page 12-5 for a description and 13-2 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
 IP
  OSPF
   Interfaces
	Interface 0
	 OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		`    Interface IP address:` **`172.128.1.2`**
		`    Attaches to area:` **`0.0.0.0`**
		
		(Accept other defaults)	

Protocols APPN General	APPN network node (checked to enable) Network ID: NUBNODE Control point name: CPNU	`Config>`**`p appn`** `APPN Config>` **`set node`** `  Enable APPN` `  Network ID:` **`NUBNODE`** `  Control point name:` **`CPNU`** (Accept other defaults)	4

Protocols
 APPN
  Interfaces
	(highlight Interface 0 Token Ring)
	 (click on the configure tab)
	 Define APPN port (checked to enable)
	 Port name: TR3270
	 High performance routing (HPR) supported
	   (unchecked to disable)
	 Support multiple PUs (checked to enable)
		`Config>`**`protocol APPN`**
		`APPN Config>`**`add port`**
		`  APPN Port Link Type:` **`TOKEN RING`**
		`  Port name:` **`TR3270`**
		`  Enable APPN`
		`  Support multiple PUs`
		`  High performance routing:` **`No`**
		
		(Accept other defaults)
			5

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
 APPN
  Interfaces
	(highlight Interface 0 Token Ring)
	(click on the Link stations tab)
	STAT001 (new definition)
	 General-1 Tab:
	  Link station name: STAT001
	  Solicit SSCP session (checked)
	  Link support APPN functions (unchecked)
	 General-2 Tab:
	  MAC address of adjacent node:
	   400000003172
	  Node ID: 12244
	  Local SAP address: 04
	(click on "Add" to create the Link station)
	
	STAT002 (new definition)
	 General-1 Tab:
	  Link station name: STAT002
	  Solicit SSCP session (checked)
	  Link support APPN functions (unchecked)
	 General-2 Tab:
	  MAC address of adjacent node:
	   400000003172
	  Node ID: 12245
	  Local SAP address: 08
	(click on "Add" to create the Link station)
		APPN Config>**add lin**
		Port name for the link station: **TR3270**
		Station name: **STAT001**
		MAC address of adjacent node: **400000003172**
		Solicit SSCP Session: **Yes**
		Local Node ID: **12244**
		Local SAP address: **4**
		Does link support APPN function?: **No**
		
		  (Accept other defaults)
		
		APPN Config>**add lin**
		Port name for the link station: **TR3270**
		Station name: **STAT002**
		MAC address of adjacent node:**400000003172**
		Solicit SSCP Session: **Yes**
		Local Node ID: **12245**
		Local SAP address: **8**
		Does link support APPN function?: **No**
		
		  (Accept other defaults)
			6

Protocols
 APPN
  TN3270E Server
   General
	TN3270E (checked to enable)
	 IP address : 172.128.1.2
	 Automatic logoff (checked to enable)
		APPN Config>**set tn3270**
		  Enable TN3270E Server
		  TN3270E Server IP Address: **172.128.1.2**
		  Automatic logoff: **Yes**
		
		  (Accept other defaults)	

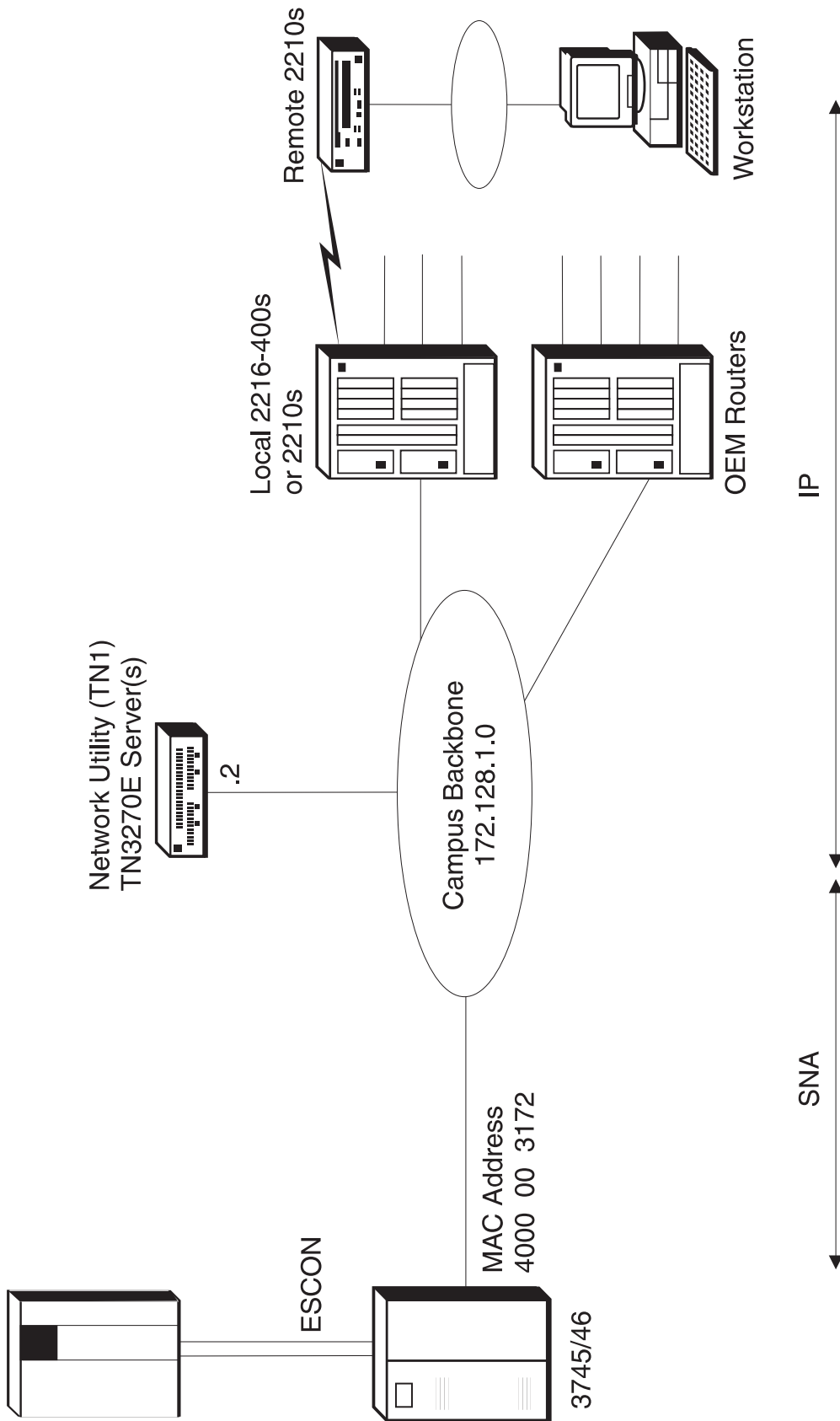*Table 13-2 (Page 4 of 5). TN3270 Subarea. See page 12-5 for a description and 13-2 for a diagram of this configuration.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  APPN
    TN3270E Server
      LUs
	Local PU Name: STAT001
	  (click on Implicit Pool)
	    LU name mask: @LU1A
	    Number of implicit workstation
	     definitions: 10
	Local PU Name: STAT002
	  (click on Implicit Pool)
	    LU name mask: @LU2A
	    Number of implicit workstation
	     definitions: 10
	  (click on LUs to define explicit LUs)
	    LU name: PC03A
	    NAU address: 5
	  (click on "Add")
		APPN Config>**add tn imp**
		  Station Name: **STAT001**
		  LU name mask: **@LU1A**
		  Number of Implicit LUs in Pool: **10**
		
		  add tn imp
		  Station Name: **STAT002**
		  LU name mask: **@LU2A**
		  Number of Implicit LUs in Pool: **10**
		
		  add tn lu
		  Station Name: **STAT002**
		  LU name: **PC03A**
		  NAU address: **5**
			7

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Notes:

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

4. If you have a pure SNA subarea network with no APPN, then the Network ID can be any value. If you have APPN in your network, then the Network ID should conform to your APPN network naming conventions.

5. APPN must be enabled even though we are using SNA subarea for our TN3270E server connection to the host. This is because the TN3270E server code uses the APPN SNA stack both for APPN and subarea communications to the host.

6. When you create the link stations, you are also implicitly creating PUs. These PUs are assigned a "Local Node ID" here. This must match the "IDNUM" in VTAM's SW Major Node definition. The ID Block is always 077 for a Network Utility. If you need to define multiple link stations (PUs), then each link station has to have a different Local SAP address.

   Setting Solicit SSCP session to yes defines the link as a subarea connection.

7. For implicit LUs, you just have to define the pools. The @LU1A is a template that will be used to create the actual LU names in the pool. In this example, with 10 LUs in the pool, the LU names generated are LU1A2, LU1A3, LU1A4,...LU1A11 which correspond to LOCADDRs 2-11 for the PU defined in VTAM. Similarly, @LU2A will generate LU2A2, LU2A3, LU2A4. Note that the LU name LU2A5 is not used because the NAU address of 5 has been reserved for the explicit definition. Therefore, the remaining LUs in the pool are LU2A6 through LU2A12.

   For explicit LUs, the LU name given here must match the name defined in the workstation's 3270 emulation configuration. The NAU address points to the LOCADDR in the appropriate PU definition in the Switched Major node in VTAM.

Network Utility (TN1 or TX1)
(Net Dispatcher)

ND A

ND B

.1

.2

172.128.1.0

Network Utility (TN1)
TN3270E Servers

Remote 2210s

Local 2216-400s
or 2210s

OEM Routers

Workstation

TN B
172.128.2.4
172.128.1.100

.4

Interface 0

172.128.2.0

TN A
172.128.2.3
172.128.1.100

.3

3745/46

IP

SNA

Figure 13-2. TN3270E Server Config -Highly Available, Fault Tolerant TN3270

Table 13-3 (Page 1 of 5). TN3270E Server Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.				
This table provides the configuration for the TN A server. *See Table 13-4 on page 13-14 and Table 13-5 on page 13-18 for the configuration of the Network Dispatchers for this scenario.*				
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Devices
  Adapters
    Slots
	Slot1: 2 port TR	See "add dev" on next row	1

Devices
  Adapters
    Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 1/Port 2: Interface 1: TR
		Config>**add dev tok**
		
		   (once for each interface)
			2

Devices
  Interfaces
	Interface 0
	  Mac Address 400022AA0053
	Interface 1
	  Mac Address 400022AA0003
		Config>**net 0**
		  set phy 40:00:22:AA:00:53
		
		  set phy 40:00:22:AA:00:03	

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		Config>
		  set host
		  set location
		  set contact	

System
  SNMP Config
    General
	SNMP (checked)	Config>**p snmp**
		SNMP Config>**enable snmp**	

System
  SNMP Config
    Communities
      General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		Config>**p snmp**
		SNMP Config>
		  add community
		  set comm access write
			3

Protocols
  IP
    General
	Internal address: 172.128.252.3
	Router ID: 172.128.1.3
	Same Subnet (checked)
		Config>**p ip**
		IP Config>
		  set internal 172.128.252.3
		  set router-id 172.128.1.3
		  enable same-subnet
			4

Table 13-3 (Page 2 of 5). TN3270E Server Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 13-4 on page 13-14 and Table 13-5 on page 13-18 for the configuration of the Network Dispatchers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
IP
  Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.2.3
	  Subnet mask: 255.255.255.0
	
	  IP address: 172.128.1.100
	  Subnet mask: 255.255.255.0
	
	Interface 1 (TR slot 1 port 2)
	  IP address: 172.128.1.3
	  Subnet mask: 255.255.255.0
		Config>**p ip**
		IP Config>
		  add address 0 172.128.2.3 255.255.255.0
		
		  add address 0 172.128.1.100 255.255.255.0
		
		  add address 1 172.128.1.3 255.255.255.0
			5,6

Protocols
IP
  OSPF
    General
	OSPF (checked)	Config>**p ospf**
		OSPF Config>**enable ospf**	

Protocols
IP
  OSPF
    Area Configuration
      General
	Area number: 0.0.0.0
	Stub area (not checked)
		Config>**p ospf**
		OSPF Config>**set area**	

Protocols
IP
  OSPF
    Interfaces
	Interface 1
	  OSPF (checked)
		Config>**p ospf**
		OSPF Config>**set interface**
		    Interface IP address: **172.128.1.3**
		    Attaches to area: **0.0.0.0**
		
		  (Accept other defaults)
			7

Protocols
APPN
  General
	APPN network node (checked to enable)
	  Network ID: NUBNODE
	  Control point name: CPNU
		Config>**p appn**
		APPN Config> **set node**
		  Enable APPN
		  Network ID: **NUBNODE**
		  Control point name: **CPNU**
		
		  (Accept other defaults)
			8

Table 13-3 (Page 3 of 5). TN3270E Server Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 13-4 on page 13-14 and Table 13-5 on page 13-18 for the configuration of the Network Dispatchers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  APPN
    Interfaces
	(highlight Interface 0 Token Ring)
	  (click on the configure tab)
	  Define APPN port (checked to enable)
	  Port name: TR3270
	  High performance routing (HPR) supported
	    (unchecked to disable)
	  Support multiple PUs (checked to enable)
		`Config>`**`protocol APPN`**
		`APPN Config>`**`add port`**
		`  APPN Port Link Type:` **`TOKEN RING`**
		`  Port name:` **`TR3270`**
		`  Enable APPN`
		`  Support multiple PUs`
		`  High performance routing:` **`No`**
		
		`  (Accept other defaults)`
			9

Protocols
  APPN
    Interfaces
	(highlight Interface 0 Token Ring)
	  (click on the Link stations tab)
	  STAT001 (new definition)
	    General-1 Tab:
	      Link station name: STAT001
	      Solicit SSCP session (checked)
	      Link support APPN functions (unchecked)
	    General-2 Tab:
	      MAC address of adjacent node:
	        400000003172
	      Node ID: 12244
	      Local SAP address: 04
	  (click on "Add" to create the Link station)
	
	  STAT002 (new definition)
	    General-1 Tab:
	      Link station name: STAT002
	      Solicit SSCP session (checked)
	      Link support APPN functions (unchecked)
	    General-2 Tab:
	      MAC address of adjacent node:
	        400000003172
	      Node ID: 12245
	      Local SAP address: 08
	  (click on "Add" to create the Link station)
		`APPN Config>`**`add lin`**
		`Port name for the link station:` **`TR3270`**
		`Station name:` **`STAT001`**
		`MAC address of adjacent node:` **`400000003172`**
		`Solicit SSCP Session:` **`Yes`**
		`Local Node ID:` **`12244`**
		`Local SAP address:` **`4`**
		`Does link support APPN function?:` **`No`**
		
		`  (Accept other defaults)`
		
		`APPN Config>`**`add lin`**
		`Port name for the link station:` **`TR3270`**
		`Station name:` **`STAT002`**
		`MAC address of adjacent node:`**`400000003172`**
		`Solicit SSCP Session:` **`Yes`**
		`Local Node ID:` **`12245`**
		`Local SAP address:` **`8`**
		`Does link support APPN function?:` **`No`**
		
		`  (Accept other defaults)`
			10

Table 13-3 (Page 4 of 5). TN3270E Server Config -Highly Available, Fault Tolerant TN3270.   See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 13-4 on page 13-14 and Table 13-5 on page 13-18 for the configuration of the Network Dispatchers for this scenario.*

Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Protocols
 APPN
  TN3270E Server
   General
	TN3270E (checked to enable)
	 IP address : 172.128.1.100
	 Automatic logoff (checked to enable)
		APPN Config>**set tn3270**
		  Enable TN3270E Server
		  TN3270E Server IP Address: **172.128.1.100**
		  Automatic logoff: **Yes**
		
		  (Accept other defaults)	

Protocols
 APPN
  TN3270E Server
   LUs
	Local PU Name: STAT001
	 (click on Implicit Pool)
	  LU name mask: @LU1A
	  Number of implicit workstation
	  definitions: 10
	
	Local PU Name: STAT002
	 (click on Implicit Pool)
	  LU name mask: @LU2A
	  Number of implicit workstation
	  definitions: 10
		APPN Config>**add tn imp**
		  Station Name: **STAT001**
		  LU name mask: **@LU1A**
		  Number of Implicit LUs in Pool: **10**
		
		  add tn imp
		  Station Name: **STAT002**
		  LU name mask: **@LU2A**
		  Number of Implicit LUs in Pool: **10**
			11

This table provides the configuration for the TN A server. *See Table 13-4 on page 13-14 and Table 13-5 on page 13-18 for the configuration of the Network Dispatchers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Notes:

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

4. You must enable the "same-subnet function" because you are using two interfaces with an IP address within the same subnet. (172.128.1.3 is assigned to TR 1 and 172.128.1.100 (cluster address) is assigned as a 2nd address to TR 0.)

5. Note that Interface 0 has been assigned 2 IP addresses, one of which is the cluster address used by the Network dispatcher. The TN3270E Server will be configured for the same address in a subsequent step. All TN3270 traffic will be sent to this address through the Network Dispatcher. In order for this traffic to reach the Network Utility's internal IP queue, this address needs to be assigned to either an interface address or the internal address. In this example, it has been assigned to an interface as the second address of that interface.

6. Note that Interface 0 is on the LAN segment which is connected to the SNA gateway. This segment carries the LLC traffic from the TN3270 server to the gateway. Depending on the rest of the configuration of your Network Utility, this segment may not have any IP traffic on it. However, since all the TN3270E servers will have the same IP address assigned to the interface on this segment, it has been assigned a subnet address (172.128.2) and all the TN3270E servers will have an address on this subnet also (in this case 172.128.2.3) in order to an IP addressing conflict.

7. It is very important **Not** to enable OSPF on the Network Dispatcher cluster address. If you do, the cluster address will be broadcast to the network as being on the TN3270e server (in addition to the Network Dispatcher machine).

8. If you have a pure SNA subarea network with no APPN, then the Network ID can be any value. If you have APPN in your network, then the Network ID should conform to your APPN network naming conventions.

9. APPN must be enabled even though we are using SNA subarea for our TN3270e server connection to the host. This is because the TN3270e server code uses the APPN SNA stack both for APPN and subarea communications to the host.

10. When you create the link stations, you are also implicitly creating PUs. These PUs are assigned a "Local Node ID" here. This must match the "IDNUM" in VTAM's SW Major Node definition. The ID Block is always 077 for a Network Utility. If you need to define multiple link stations (PUs), then each link station has to have a different Local SAP address.

11. For implicit LUs, you just have to define the pools. The `@LU1A` is a template that will be used to create the actual LU names in the pool. In this example, with 10 LUs in the pool, the LU names generated are `LU1A2`, `LU1A3`,...`LU1A11` which correspond to LOCADDRs 2-11 for the PU defined in VTAM. Similarly, `@LU2A` will generate `LU2A2`, `LU2A3`, ... `LU2A11`.

Table 13-4 (Page 1 of 4). Network Dispatcher Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 13-5 on page 13-18 for the configuration of the backup Network Dispatcher. See Table 13-2 on page 13-3 for the configuration of the TN3270e servers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Devices
  Adapters
    Slots
	Slot 1: 2 Port TR	See "add device" on next row	1

Devices
  Adapters
    Ports
	Slot 1/Port 1: Interface 0: TR	Config>**add dev tok**	2

Devices
  Interfaces
	Interface 0
	  MAC address: 400022AA0001
		Config>**net 0**
		TKR Config>**set phy 40:00:22:AA:00:01**	

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		Config>
		  set host
		  set location
		  set contact	

System
  SNMP Config
    General
	SNMP (checked)	Config>**p snmp**
		SNMP Config>**enable snmp**	

System
  SNMP Config
    Communities
      General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		Config>**p snmp**
		SNMP Config>
		  add community
		  set comm access write
			3

Protocols
  IP
    General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		Config>**p ip**
		IP Config>
		  set internal 172.128.252.1
		  set router-id 172.128.1.1
			4

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.1
	  Subnet mask: 255.255.255.0
		Config>**p ip**
		IP Config>**add address**	

Protocols
  IP
    OSPF
      General
	OSPF (checked)	Config>**p ospf**
		OSPF Config>**enable ospf**	

Table 13-4 (Page 2 of 4). Network Dispatcher Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 13-5 on page 13-18 for the configuration of the backup Network Dispatcher. See Table 13-2 on page 13-3 for the configuration of the TN3270e servers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
 IP
  OSPF
   Area Configuration
    General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Protocols
 IP
  OSPF
   Interfaces
	Interface 0
	 OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		`    Interface IP address` **`172.128.1.1`**
		`    Attaches to area` **`0.0.0.0`**
		
		`    (Accept other defaults)`	

Features
 Network Dispatcher
  Router
   Executor
	Executor (checked)	`Config>`**`feat ndr`**
		`NDR Config>`**`enable executor`**	

Features
 Network Dispatcher
  Router
   Clusters
    Detail
	Cluster address: 172.128.1.100	`NDR Config>`**`add cluster`**
		`   Cluster Address:` **`172.128.1.100`**
		
		`   (Accept other defaults)`	

Features
 Network Dispatcher
  Router
   Clusters
    Ports
	Port Number 23	`NDR Config>`**`add port`**
		`   Cluster Address` **`172.128.1.100`**
		`   Port number` **`23`**
		
		`   (Accept other defaults)`	

Features
 Network Dispatcher
  Router
   Clusters
    Servers
	Server address: 172.128.1.3
	              172.128.1.4
		`NDR Config>`**`add server`**
		`   Cluster Address:` **`172.128.1.100`**
		`   Port number:` **`23`**
		`   Server Address:` **`172.128.1.3`**
		
		`   (Accept other defaults)`
		`       (Repeat for 172.128.1.4)`	

Table 13-4 (Page 3 of 4). Network Dispatcher Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 13-5 on page 13-18 for the configuration of the backup Network Dispatcher. See Table 13-2 on page 13-3 for the configuration of the TN3270e servers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Features
  Network Dispatcher
    Router
      Manager
	Manager (checked)
	Proportion
	  Active: 10
	  New: 10
	  Advisor: 80
	  System: 0
		NDR Config>**enable manager**
		   set manager propor
		     Active: **10**
		     New: **10**
		     Advisor: **80**
		     System: **0**
		
		  (Accept other defaults)
			5

Features
  Network Dispatcher
    Router
      Advisors
	Advisor (checked)
	  Advisor name: TN3270
	  Advisor port: 23
	  Timeout: 10
		NDR Config>**add advisor**
		  Advisor name: **3**
		
		  (for TN3270)
		
		Timeout: **10**
		
		   (Accept other defaults)
		NDR Config>**enable advisor**
		  Advisor name: **3**
		
		  (for TN3270)
		
		Port number: **23**
			6

Features
  Network Dispatcher
    Router
      Backup
	Backup (checked to enable)
	Backup role: PRIMARY
	Switch back Strategy: MANUAL
		NDR Config>**add backup**
		  Role: **0**=PRIMARY
		  Switch back strategy: **1**=MANUAL
			7

Features
  Network Dispatcher
    Router
      Reaches
	Reach address:
	  (Enter each address and click on add)
	  172.128.1.3
	  172.128.1.4
		NDR Config>**add reach**
		  Address to reach: **172.128.1.3**
		
		  (Repeat for 172.128.1.4)
			8

Features
  Network Dispatcher
    Router
      Heart Beats
	Source address: 172.128.1.1
	Target address: 172.128.1.2
	  (Enter addresses and click on "Add")
		NDR Config>**add heartbeat**
		  Source Heartbeat address: **172.128.1.1**
		  Target Heartbeat Address: **172.128.1.2**
			8

Table 13-4 (Page 4 of 4). Network Dispatcher Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 13-5 on page 13-18 for the configuration of the backup Network Dispatcher. See Table 13-2 on page 13-3 for the configuration of the TN3270e servers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Notes:

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

4. The internal address must be set in order for the advisor and the manager functions to communicate with the executor component of network dispatcher.

5. The values for Active, New, Advisor, and System must add up to 100. The Advisor proportion defaults to 0. You need to change this so that the Advisor input can be used to load balance the TN3270 traffic. In this case, it has been set to 80 to give it the a much greater weight than those for active and new connections.

6. The communication port number (defaults to 10008) must match the server's "Network Dispatcher advisor port".

7. Switch back Strategy must be the same for both primary and backup network dispatchers. IBM recommends a manual setting so that you can schedule the switch back at a time when you have the least probability of disrupting your SNA sessions.

8. The reach addresses are the addresses that the Network Dispatcher must be able to reach in order for it to determine that it is functioning correctly. The primary sends this information at regular intervals to the backup. If the backup determines that it has better reachability than the primary, then it will perform a switch over and assume the primary role. Choose at least one host on each subnet that the Network Dispatcher uses. Also, add the addresses for each server in the cluster. In this example, the Network Dispatcher uses only one interface and both servers are on the same subnet as this interface.

9. Here, you are configuring the connection that the primary Network dispatcher will use to send the heart beats to the backup Network Dispatcher. You can define several paths if you have multiple connections between the primary and backup. The heartbeats will be sent over the first path that is available. The most robust solution is to configure a second path between the primary network dispatcher and the backup network dispatcher using the second slot that is available in each Network Utility.

Table 13-5. Network Dispatcher Config -Highly Available, Fault Tolerant TN3270. See page 12-9 for a description and 13-8 for a diagram of this configuration.

This table provides the configuration differences for the backup Network Dispatcher ND B based on Table 13-4 on page 13-14, which gives the configuration for the primary Network Dispatcher. *The definition for the backup Network Dispatcher is the same as for the Primary except for the differences that are shown in this table. These differences correspond to the interface addresses and the Network Dispatcher backup functions. The parameters related to the Network Dispatcher that are not shown here must be identical to the values configured on the primary. It is also recommended that the hardware configuration be the same for both the primary and backup Network Dispatchers. See Table 13-3 on page 13-9 for the configuration of the TN3270e servers for this scenario.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Devices
  Interfaces
	Interface 0
	  Mac Address 400022AA0002
		`Config>`**`net 0`**
		  `set phy 40:00:22:AA:00:02`	

System General	System name: NU_ND2	`Config>`**`set host`**	

Protocols
  IP
    General
	Internal address: 172.128.252.2
	Router ID: 172.128.1.2
		`Config>`**`p ip`**
		`IP Config>`
		  `set internal 172.128.252.2`
		  `set router-id 172.128.1.2`
			1

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.2
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP Config>`**`add address`**	

Protocols
  IP
    OSPF
      Interfaces
	Interface 0
	  OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		    `Interface IP address:` **`172.128.1.2`**
		    `Attaches to area:` **`0.0.0.0`**
		
		  (accept other defaults)	

Features
  Network Dispatcher
    Router
      Backup
	Backup (checked to enable)
	Backup role: BACKUP
	Switch back Strategy: MANUAL
		`Config>`**`feat NDR`**
		  `NDR Config>`**`add backup`**
		    `Role:` **`1`**`=BACKUP`
		    `Switch back strategy:` **`1`**`=MANUAL`	

Features
  Network Dispatcher
    Router
      Heart Beats
	Source address: 172.128.1.2
	Target address: 172.128.1.1
	  (Enter addresses and click on "Add")
		`Config>`**`feat NDR`**
		  `NDR Config>`**`add heartbeat`**
		    `Source Heartbeat address:` **`172.128.1.2`**
		    `Target Heartbeat Address:` **`172.128.1.1`**
			2

Notes:

1. The internal address must be set in order for the advisor and the manager functions to communicate with the executor component of network dispatcher.

2. The backup must be configured with all the same information as the primary network dispatcher so that if the primary fails, the backup can assume the full role of primary including the sending of the heartbeats and the reachability information to the primary when it comes back on line.
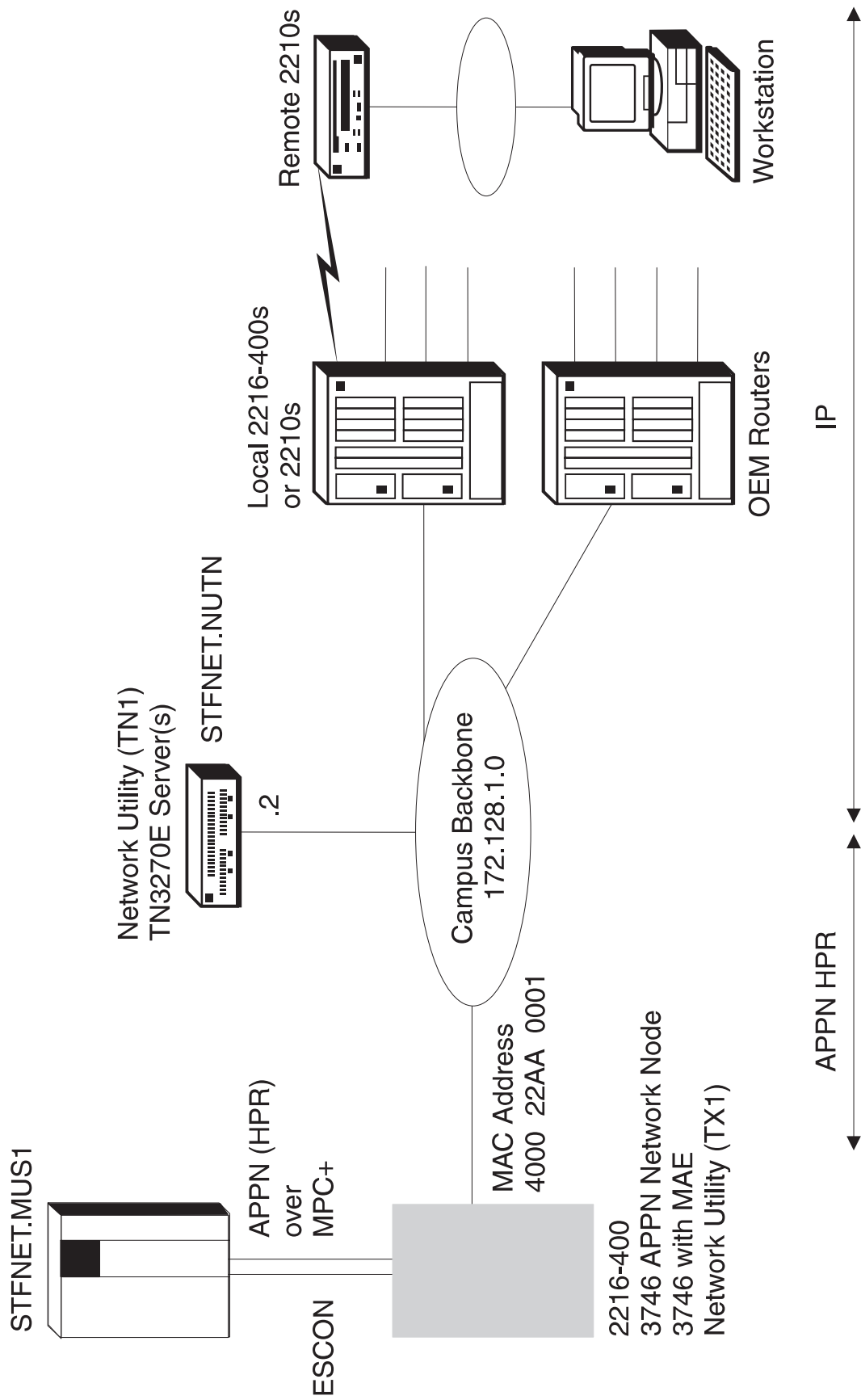
STFNET.MUS1

APPN (HPR) over MPC+

ESCON

MAC Address 4000 22AA 0001

2216-400
3746 APPN Network Node
3746 with MAE
Network Utility (TX1)

Network Utility (TN1)
TN3270E Server(s)

STFNET.NUTN

.2

Campus Backbone 172.128.1.0

Remote 2210s

Workstation

Local 2216-400s or 2210s

OEM Routers

IP

APPN HPR

Figure 13-3. TN3270 via DLUR over APPN

Table 13-6 (Page 1 of 4). TN3270 via DLUR over APPN. See page 12-12 for a description and 13-19 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Devices
  Adapters
    Slots
	Slot1: 2 port TR	See "add dev" on next row	1

Devices
  Adapters
    Ports
	Slot 1/Port 1: Interface 0: TR	`Config>`**`add dev tok`**	2

Devices
  Interfaces
	Interface 0
	  Mac Address 400022AA0011
		`Config>`**`net 0`**
		  `set phy 40:00:22:AA:00:11`	

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System
  SNMP Config
    General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

System
  SNMP Config
    Communities
      General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`Config>`**`p snmp`**
		`SNMP Config>`
		  `add community`
		  `set comm access write`
			3

Protocols
  IP
    General
	Internal address: 172.128.252.2
	Router ID: 172.128.1.2
		`Config>`**`p ip`**
		`IP Config>`
		  `set internal 172.128.252.2`
		  `set router-id 172.128.1.2`	

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.2
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP Config>`**`add address`**	

Protocols
  IP
    OSPF
      General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**	

Protocols
  IP
    OSPF
      Area Configuration
        General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Table 13-6 (Page 2 of 4). TN3270 via DLUR over APPN. See page 12-12 for a description and 13-19 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Protocols
  IP
    OSPF
      Interfaces
	Interface 0
	  OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		   `Interface IP address:` **`172.128.1.2`**
		   `Attaches to area:` **`0.0.0.0`**
		
		  (Accept other defaults)	

Protocols
  APPN
    General
	APPN network node (checked to enable)
	  Network ID: STFNET
	  Control point name: NUTN
		`Config>`**`p appn`**
		`APPN Config>` **`set node`**
		  `Enable APPN`
		  `Network ID:` **`STFNET`**
		  `Control point name:` **`NUTN`**
		
		  (Accept other defaults)	

Protocols
  APPN
    Interfaces
	(highlight Interface 0 Token Ring)
	  (click on the configure tab)
	  Define APPN port (checked to enable)
	  Port name: TR001
		`Config>`**`protocol APPN`**
		`APPN Config>`**`add port`**
		  `APPN Port Link Type:` **`TOKEN RING`**
		  `Port name:` **`TR001`**
		  `Enable APPN`
		
		  (Accept other defaults)
			4

Protocols
  APPN
    Interfaces
	(highlight Interface 0 Token Ring)
	  (click on the Link stations tab)
	  TRTG001 (new definition)
	    General-1 Tab:
	      Link station name: TRTG001
	    General-2 Tab:
	      MAC address of adjacent node:
	        400022AA0001
	      Adjacent Node Type:
	        APPN Network Node
	  (click on "Add" to create the Link station)
		`APPN Config>`**`add lin`**
		`Port name for the link station:` **`TR001`**
		`Station name:` **`TRTG001`**
		`MAC address of adjacent node:` **`400022AA0001`**
		
		  (Accept other defaults)
			5

Protocols
  APPN
    DLUR
	DLUR (checked to enable)
	  Fully-qualified CP name of
	    primary DLUS: STFNET.MVS1
		`APPN Config>`**`set dlur`**
		  `Enable DLUR`
		  `Fully-qualified CP name of`
		    `primary DLUS:` **`STFNET.MVS1`**
		
		  (Accept other defaults)
			6

Table 13-6 (Page 3 of 4). TN3270 via DLUR over APPN. See page 12-12 for a description and 13-19 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
APPN
  TN3270e Server
   General
	TN3270e (checked to enable)
	  IP address : 172.128.1.2
	  Automatic logoff (checked to enable)
		`APPN Config>`**`set tn3270`**
		  `Enable TN3270e Server`
		  `TN3270e Server IP Address:` **`172.128.1.2`**
		  `Automatic logoff:` **`Yes`**
		
		  (Accept other defaults)	

Protocols
APPN
  TN3270e Server
   Local PUs
	Link Station Name: PUPS08T
	Node ID: 12244
	
	Link Station Name: PUPS18T
	Node ID: 12245
		`APPN Config>`**`add loc`**
		  `Station Name:` **`PUPS08T`**
		  `Local Node ID:` **`12244`**
		
		  (Accept other defaults)
		
		`APPN Config>`**`add loc`**
		  `Station Name:` **`PUPS18T`**
		  `Local Node ID:` **`12245`**
		
		  (Accept other defaults)
			7

Protocols
APPN
  TN3270e Server
   LUs
	Local PU Name: PUPS08T
	  (click on Implicit Pool)
	   LU name mask: @LU1A
	   Number of implicit workstation
	    definitions: 5
	Local PU Name: PUPS18T
	  (click on Implicit Pool)
	   LU name mask: @LU2A
	   Number of implicit workstation
	    definitions: 5
		`APPN Config>`**`add tn imp`**
		  `Station Name:` **`PUPS08T`**
		  `LU name mask:` **`@LU1A`**
		  `Number of Implicit LUs in Pool:` **`5`**
		
		  `add tn imp`
		  `Station Name:` **`PUPS18T`**
		  `LU name mask:` **`@LU2A`**
		  `Number of Implicit LUs in Pool:` **`5`**
			8

Table 13-6 (Page 4 of 4). TN3270 via DLUR over APPN. See page 12-12 for a description and 13-19 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Notes:

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

4. When using APPN, you can either use High Performance Routing (HPR) or Intermediate Session Routing (ISR). HPR is the default and is what is used in this scenario.

5. The MAC address specified is the MAC address of the APPN host gateway.

6. The CP name of the DLUS is the host VTAM.

7. The Local Node IDs entered for these PUs need to match the IDNUM fields in the PU definitions in the host VTAM.

8. For implicit LUs, you just have to define the pools. The @LU1A is a template that will be used to create the actual LU names in the pool. In this example, with 5 LUs in the pool, the LU names generated are LU1A2, LU1A3, LU1A4, LU1A5 and LU1A6 which correspond to LOCADDRs 2-6 for the PU defined in VTAM. Similarly, @LU2A will generate LU2A2 through LU2A6.

Chapter 14. Channel Gateway

Overview

The Network Utility provides host connectivity through an ESCON channel or Parallel Channel. It enables the Network Utility to function as a gateway from the host to other networks.

Configurations Supported

There are three interfaces from host software to a Network Utility.

The first interface is the 8232-compatible support, called LAN Channel Station (LCS). This interface defines a number of commands for direct LAN connection and a blocking and deblocking structure. LAN-ready frames are transmitted from the host to the virtual LAN adapters and vice versa. This interface is used by TCP/IP for VM and MVS and AIX/370.

The second interface is the Link Services Architecture (LSA) support, which is accessed in the host through VTAM.

The LSA support is a control interface to allow VTAM to use the Logical Link Control (LLC) portion of the Data Link Control (DLC) layer of the SNA stack. Included is access to LLC Type 1 (connectionless) and LLC Type 2 (connection oriented) data transport. This interface is used by VTAM for both SNA subarea and APPN ISR and HPR data transport.

The third interface is the Multi-path Channel (MPC+) support, which is accessed in the host through VTAM. The MPC+ support is a protocol layer which allows multiple read and write subchannels to be treated as a single transmission group between the host and channel attached devices. This interface is used by OS/390 for APPN HPR, TCP/IP, and HPDT UDP data transport. Note that the Channel does not support MPC+ subchannel groups which are shared over more than one physical channel interface.

The Network Utility can support 32 ESCON subchannels, in any combination of LCS subchannel pairs, LSA subchannels, and MPC+ groups. This allows a maximum of 16 LCS virtual LAN adapters, or 16 LSA virtual LAN adapters, or 16 MPC+ groups (an MPC+ group must include at least one read subchannel and one write subchannel).

LSA and LCS virtual LAN adapters emulate either a Token-Ring, FDDI, or Ethernet interface for communications with the host. This does not restrict the format of the remote network interface. It is intended only to maintain the existing host interfaces of the 3172 Interconnect Controller, to eliminate host support changes.

Each virtual LAN adapter or MPC+ group can support only one host connection type (LCS/LSA/MPC+). LSA and LCS subchannels can support multiple virtual LAN adapters, for example, one Token-Ring interface and one Ethernet interface. There is no perceived value to supporting multiple virtual LAN adapters of the same type on a single subchannel or pair, but configuration will not preclude it.

Host LAN Gateway Function

The host LAN gateway function allows host applications to communicate with LAN-based workstations. The two main host applications supported by the host LAN gateway function are TCP/IP and VTAM. These applications encapsulate LAN frames into Channel Control Words (CCWs) for transport across the channel. This is also referred to as "blocking", because a CCW consists of a block of LAN frames sent as a single logical unit. The CCW is then "deblocked" by the receiver into individual frames.

Much of the Network Utility LAN gateway function is based on the 3172 Interconnect Controller Program (ICP). Even though there are differences in the 3172 ICP gateway function and the Network Utility Channel function, the hardware and software interfaces between the host and the Network Utility Channel are the same as the interfaces between the host and the 3172 ICP (except for the IP routing support provided within the Network Utility). To preserve the software interface, it is necessary for the Network Utility to create the appearance of a LAN adapter so that the host application still believes it is communicating with a real LAN.

ESCON Channel Concepts

Subchannels

The ESCON channel interface is divided into 256 logical addresses (inaccurately but consistently referred to as "subchannels" for historical reasons). Each host application interface uses one or more subchannels to connect the host application to the Network Utility. Through configuration, each subchannel is assigned a unique relative index, which may or may not match its actual logical address. The ESCON channel may be shared by multiple applications on multiple hosts, but each host application will have dedicated use of its subchannels. (This is not strictly true for MPC+, as we will see later, but the statement applies at the MPC+ level; MPC+ subchannels cannot be shared with non-MPC+ applications.) The Network Utility supports up to 32 subchannels at a time.

Channel Protocols

Network Utility supports three channel protocols, corresponding to the three host software interfaces discussed above. Each protocol uses its subchannels differently, and a subchannel can only support one protocol at a time. The channel protocols supported are LAN Channel Station (LCS), Link Services Architecture (LSA) and Multi-Path Channel (MPC+). Each is discussed briefly below.

LAN Channel Station (LCS): LCS is a channel protocol supported by TCP/IP applications in the host. Each application defines a consecutive pair of subchannels, one for TCP/IP to read from the channel, and one for TCP/IP to write to the channel. The LCS interface allows LAN MAC frames to be transported over the channel, and provides a command interface to activate, deactivate, and query the LAN interfaces. Each MAC frame has a header which identifies the virtual LAN adapter destination of the frame.

Link Services Architecture (LSA): LSA is an interface to support SNA traffic over the channel. Each LSA path is a single bidirectional subchannel between the host application and the Network Utility. The host software (VTAM) issues a read command immediately following each write command to retrieve data from the channel. The Network Utility also issues an Attention command when it has

something for the host application to read. LSA has a command interface which allows VTAM to open Service Access Points (SAPs) to communicate with downstream workstations using the IEEE 802.2 Logical Link Control (LLC) interface. The channel blocking/deblocking mechanism for LSA subchannels is the same as for LCS subchannel pairs.

Multi-Path Channel (MPC+): MPC+ is a data link control (DLC) interface for the channel. Each MPC+ path consists of one or more read subchannels and one or more write subchannels, bound together to form a transmission group. MPC+ transmission groups which span more than one physical ESCON channel are not supported in this release. VTAM and the Network Utility exchange XIDs to identify the number and direction of subchannels at initialization, and then each frame has a header to indicate the sending and receiving applications.

Blocks: The host channel interface packages control and data frames in blocks of up to 32K bytes (36K bytes for MPC+). The format of data blocks is different for MPC+ and non-MPC+ host applications. LSA and LCS blocks consist of one or more contiguous frames, each with a header which identifies the destination device by its "LAN type" and "LAN number". MPC+ blocks contain one or more "discontiguous" frames, with the first 4K bytes of the block containing MPC+ PDU headers and offsets of application data, which is stored in the last 32K of the block. MPC+ groups are identified by a "LAN type" and "LAN number" as well for implementation consistency.

A block of data is transmitted either when it is filled, or when the block delay timer (which determines how long the adapter waits for the block to fill before transmitting) expires. The process of receiving a block of data and forwarding the individual frames to the device driver is called "deblocking".

Virtual LAN Adapters: First, a little history: the 3172 Interconnect Control Program (on which the Network Utility is partially based) transferred frames from a host channel to one or more LANs. In its configuration, each subchannel was connected to one or more LAN device drivers. Data from the host was received by a deblocker, which would distribute the frames to one of the LAN adapters based on the "LAN Type" and "LAN Number" contained in the frame header. If a host application needed access to multiple LAN adapters, the configuration file would contain one entry for each LAN adapter.

In the Network Utility, things are a little different. Instead of each subchannel being connected to one or more real LAN adapters, all of the subchannels are connected to the Base Net Handler, which is in turn connected to one or more virtual net handlers. Each virtual net handler supports one of the three channel protocols (LSA/LCS/MPC+) and sends and receives frames with one of the protocol applications (LLC/IP/APPN), which sends the data to another net handler representing a network connection. There may or may not be any real LAN adapters connected to the Network Utility.

In order to preserve the existing host interfaces, the Network Utility takes on the appearance of multiple LAN adapters for LSA and LCS connections. Based on configuration parameters, the Virtual net handlers register with the appropriate protocols as either Token-Ring, Ethernet, or FDDI adapters. The Base Net Handler allows the host to activate and deactivate this "virtual LAN adapter" in the same way it controls the 3172's real LAN adapters. Each virtual LAN adapter has its own

MAC address, which allows the Network Utility to appear to the host as one or more LAN adapters on an actual local area network.

A single subchannel (or pair) may be connected to one or more virtual LAN adapters. This is necessary to allow a single host application to communicate with different types of LANs (Token-Ring, Ethernet, FDDI) over the same subchannel. LAN-bound frames are directed to the correct destination by the LAN Type and LAN Number in the frame header.

However, the inverse is only true for LSA connections. A single LCS virtual LAN adapter may be connected to only one subchannel. This restriction improves data throughput by allowing host-bound frames to be directed to the correct subchannel by the virtual net handler, without forcing the net handler to examine the MAC address or IP address of each host-bound frame. Multiple VTAMs can share a single LSA net handler if each opens a SAP with a unique number. This cannot be done for the LCS net handler because all TCP/IP traffic uses the multiprotocol SAP number 'AA'x. See Figure 14-1.
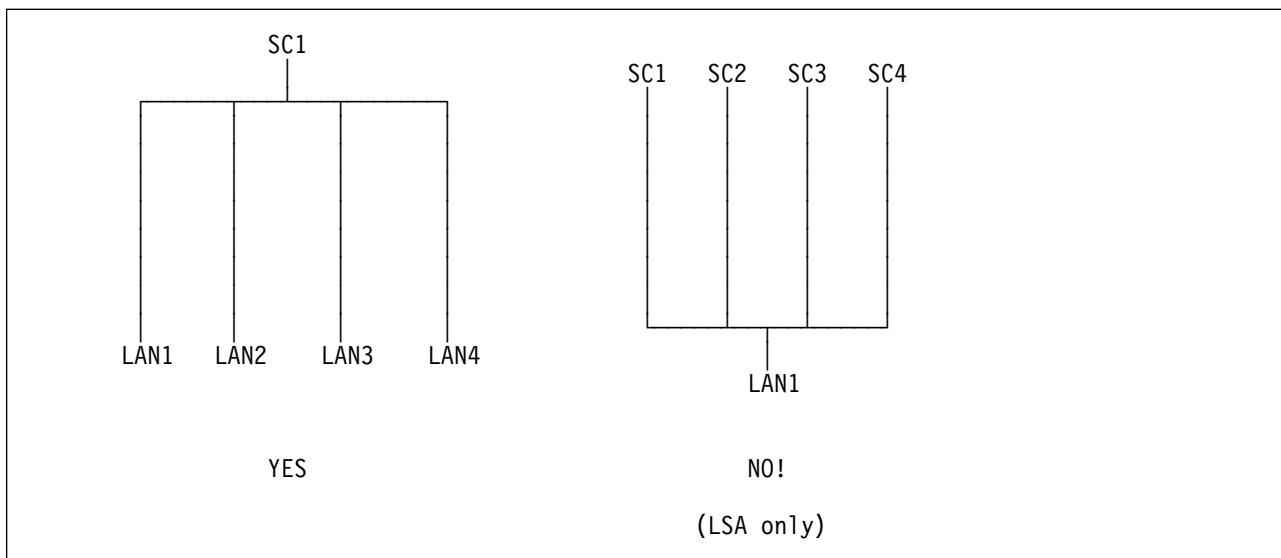
Figure 14-1. LAN to Subchannel Configuration

MPC+ Groups: MPC+ does not use the Virtual LAN Adapter concepts common to both LSA and LCS interfaces, since MPC+ does not support a LAN gateway appearance for the Network Utility. The equivalent interface for MPC+ is the MPC+ group. An MPC+ group is a set of ESCON subchannels configured to act as a single data pipe between the host and Network Utility. An MPC+ group consists of at least one "read" subchannel and at least one "write" subchannel. Any number of subchannels may be designated as read or write, and multiple MPC+ groups may be defined, subject to a maximum of 32 total subchannels per Network Utility.

Data may be sent over any or all of the active subchannels in an MPC+ group. The MPC+ endpoint is responsible for maintaining data order over a group. The number of subchannels is fixed when the MPC+ group is defined.

MPC+ groups are identified in the microcode using the same "LAN type" and "LAN number" designation as virtual LAN adapters. As frames are deblocked by the microcode, each frame is given a "LAN type" of MPC+ and a "LAN number" which corresponds to the MPC+ group associated with the subchannel it was received on.

This allows the microcode and net handler to process MPC+ frames in a manner consistent with LSA and LCS frames.

LLC Loopback: LLC Loopback is an extension of the virtual LAN adapter concept to allow VTAM connections with the APPN and DLSw functions in the Network Utility. To establish an SNA connection, the LSA interface creates an LLC connection between itself and the remote device across the LAN using IEEE 802.2 frames. See Figure 14-2.

```
┌──────────┐
│  VTAM    │
└────┬─────┘
     │  LSA primitive
     │
     ▼
┌──────────┐            ┌──────────┐
│  LSA     │            │ Remote   │
└────┬─────┘            │ Appl     │
     │                  └────▲─────┘
     │  LLC command          │  LLC notification
     ▼                       │
┌──────────┐            ┌────┴─────┐
│  LLC     │───────────▶│  LLC     │
└──────────┘  802.2 frame└──────────┘
```
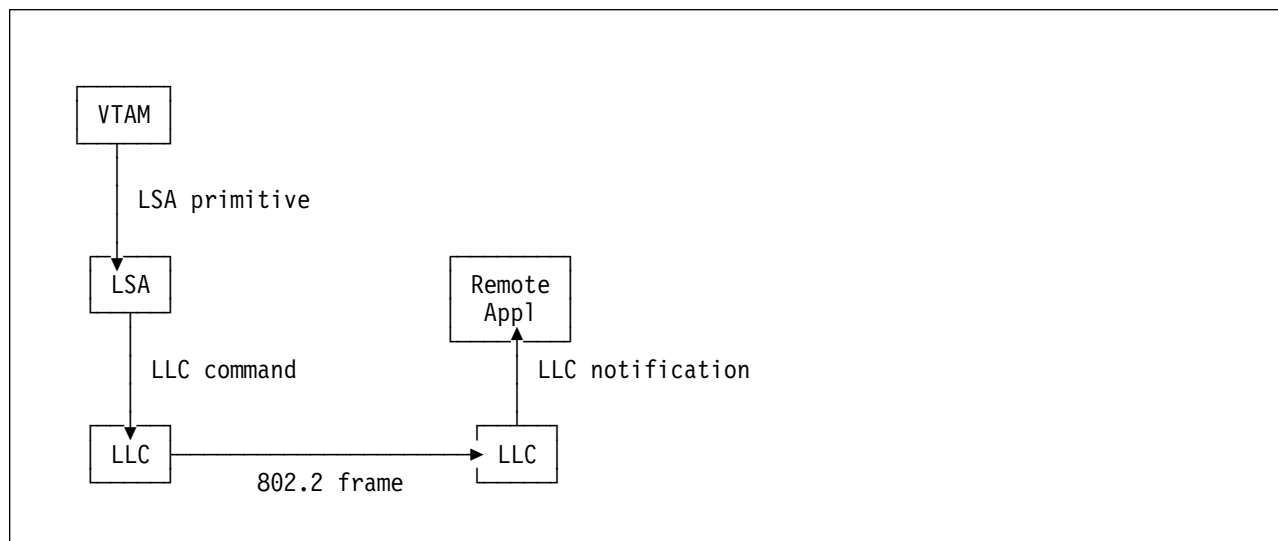
Figure 14-2. Normal LLC Connection

LLC Loopback allows the Network Utility to communicate directly with other LLC users (APPN and DLSw) in the Network Utility. Instead of turning LLC commands from LSA into 802.2 frames, they are converted into LLC notifications and sent to the appropriate LLC user. See Figure 14-3.

```
┌──────────┐
│  VTAM    │
└────┬─────┘
     │  LSA primitive
     │
     ▼
┌──────────┐            ┌──────────┐
│  LSA     │            │APPN/DLSW │
└────┬─────┘            └────▲─────┘
     │                       │
     │  LLC command          │  LLC notification
     ▼                       │
┌───────────────────────────┴──┐
│        LLC Loopback           │
└──────────────────────────────┘
```
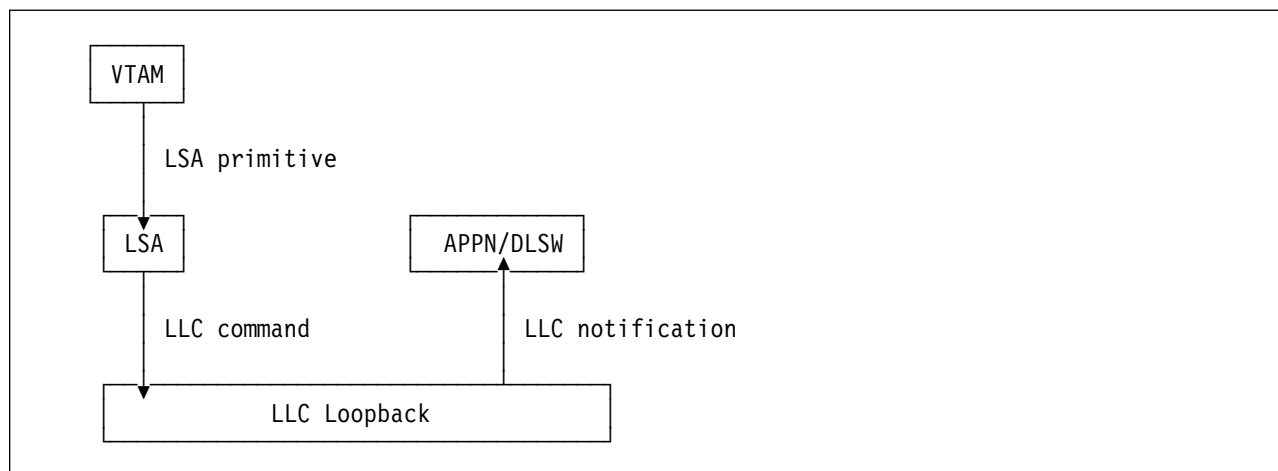
Figure 14-3. LLC Loopback Connection

LLC Loopback allows the APPN Network Node in Network Utility to act as the adjacent node to VTAM. It also permits VTAM to connect to remote devices and applications using Data Link Switching without requiring changes to VTAM's LSA support, since the loopback connection appears the same as a normal LLC connection to VTAM.

Example Configurations

This section describes four sample configurations where the Network Utility is used as a channel gateway to a mainframe system. Three of the scenarios show ESCON channel configurations and one shows a parallel channel. These configurations are:

- ESCON Channel Gateway (SNA and IP)
- Parallel Channel Gateway (SNA and IP)
- ESCON Channel Gateway (APPN and IP)
- ESCON Channel Gateway - High Availability

All of these configurations can be built using either the Network Utility model TN1 or TX1. You do not need the extra function provided by the model TN1 unless you are planning to configure the TN3270e server function in the same machine.

ESCON Channel Gateway

This scenario is shown in Figure 14-4. The Network Utility is configured to support both SNA and IP traffic into the host from both remote sites and LAN segments at the main site. The ESCON channel adapter is configured with an LSA direct interface to transport the SNA traffic and an LCS interface to perform IP forwarding.
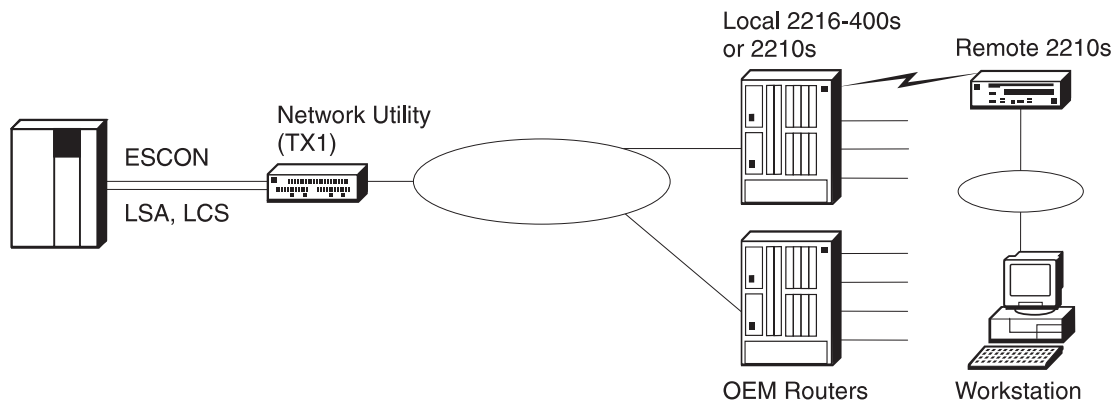
Figure 14-4. ESCON Channel Gateway

Keys to Configuration

The subchannel definitions for both the LCS and the LSA interfaces must match parameters used in the host to define the Network Utility to the host channel subsystem. The key subchannel parameters to configure at the Network Utility are shown in Table 14-1 on page 14-7.

Table 14-1. Network Utility Subchannel Configuration Parameters

Command	Description
device	The unit address transmitted on the channel path to select the Network Utility. It is also referred to as subchannel number in S/370 I/O architecture. It is a two-digit hexadecimal value that may range from 00-FF. This value is defined in the host Input/Output Configuration Program (IOCP) by the UNITADD statement on the CNTLUNIT macro instruction for the real device.
	Valid Values: X'00' to X'FF'
	Default:　None
cu	The Control Unit address defined in the host for the Network Utility. This value is defined in the host IOCP by the CUADD statement on the CNTLUNIT macro instruction. The Control Unit Address must be unique for each logical partition defined on the same host.
	Valid Values: X'0' - X'F'
	Default:　X'0'
link	This parameter is significant when an IBM 9032 ESCON Director (ESCD) is used between the Network Utility and the host. When an ESCD is used, the link address is the port number of the ESCON Director (ESCD) to which the *host* is attached. If two ESCDs are in the path, it is the host-side port number of the ESCD defined with the dynamic connection. When no ESCD is in the communication path, this value must be set to X'01'.
	Valid Values: X'01' - X'FE'
	Default:　X'01'
lpar	Logical partition number. This allows multiple logical host partitions to share one ESCON fiber. This value is defined in the host IOCP by the RESOURCE macro instruction. If the host is not using ESCON Multiple Image Facility (EMIF), use the default of 0 for the LPAR number.
	Valid Values: X'0' - X'F'
	Default:　X'0'

The LSA Direct Interface: Figure 14-5 on page 14-8 shows how the configuration parameters for the Network Utility correlate to the host parameters for an LSA interface definition.
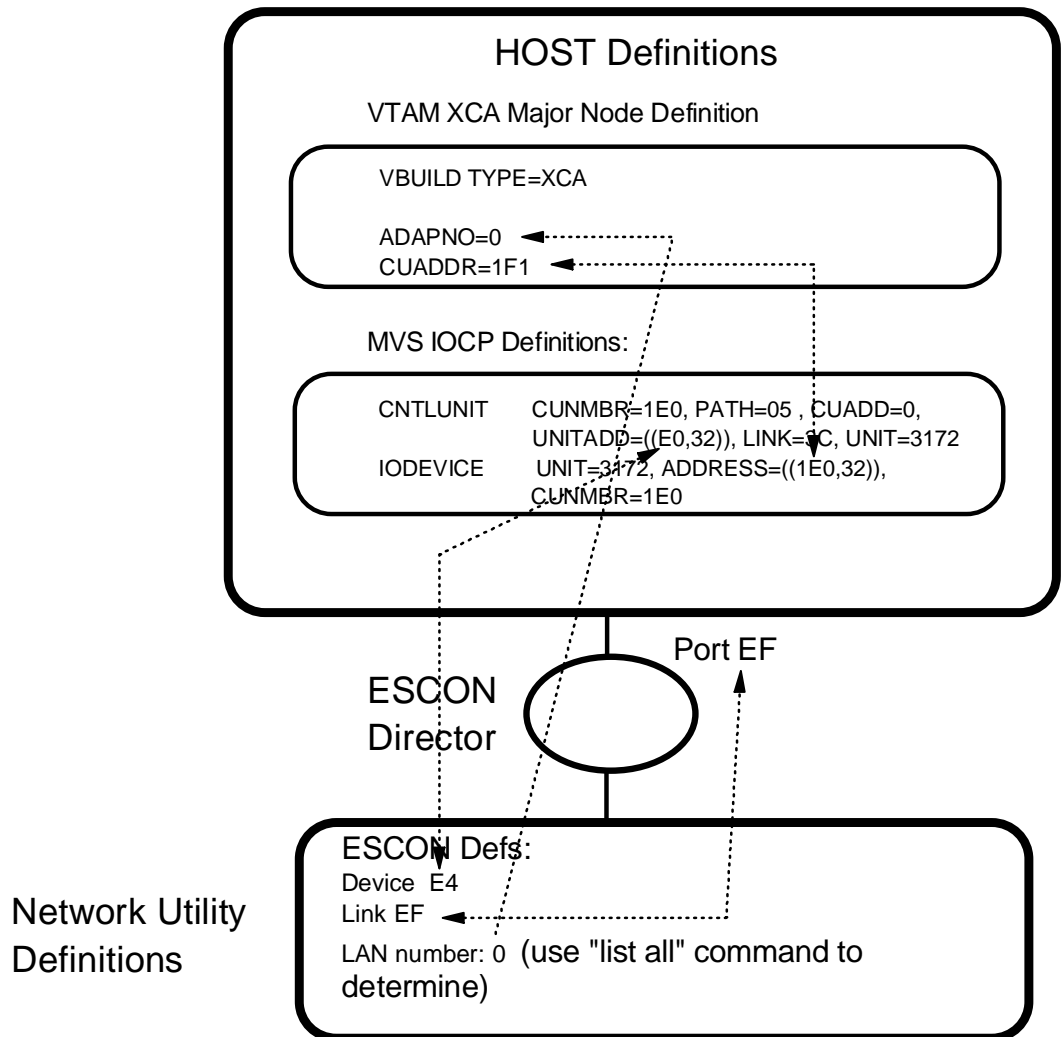
Figure 14-5. Host/Network Utility Parameter Relationships - LSA

Notes:

1. LSA uses a single bi-directional subchannel between the host and the Network Utility. VTAM issues a read command immediately following each write command to retrieve data from the channel.

2. The device address specified in the Network Utility LSA interface definition must be within the range specified in the UNITADD parameter in the CNTLUNIT macro from the IOCP. For example, the UNITADD parameter in Figure 14-5 shows that 32 (decimal) device addresses starting at E0 (hex) are being reserved for the Network Utility definition. A device address of E4 has been specified for the Network Utility LSA interface. Since E4 is in the range between E0 and FF hex, this is OK as long as no other device (or interface on this Network Utility) tries to use that subchannel.

3. The value specified in the CUADDR parameter in the VTAM XCA major node definition must be within the range specified in the ADDRESS parameter in the IODEVICE macro from the IOCP. For example, the CUADDR parameter in the XCA major node definition in Figure 14-5 is 1F1 hex, which is in the range between 1E0 and 1FF that the ADDRESS parameter in the IODEVICE statement specifies.

4. The values specified for the ADDRESS parameter in the IODEVICE macro and the UNITADD parameter in the CNTLUNIT macro are related *by convention only*. In this example, the value for the ADDRESS parameter has been determined from the value for the UNITADD parameter by prepending a *logical channel identifier* (1 in this case) to the UNITADD value. This will often be the case. However, when defining the device address on the Network Utility LSA definition, use the UNITADD parameter and not the ADDRESS parameter to determine the valid range of values.

5. When you define an LSA direct interface on the Network Utility, you associate the interface with one of the LAN interfaces on the Network Utility. In effect, this puts the LSA direct interface on this same LAN segment. Every frame with a destination address of the MAC address of the Network Utility adapter on this LAN segment automatically gets forwarded over the channel to the host.

See Chapter 18, "Sample Host Definitions" for more explanation and samples of host definitions for this interface type.

For a complete look at the configuration parameters needed for this scenario, see Table 15-2 on page 15-3.

The LCS interface: Defining an LCS interface creates a virtual LAN inside the Network Utility. There are two IP stations on this LAN: the Network Utility and the host. This LAN must be a unique IP subnet in the network. A MAC address is also needed for the LCS interface. After you create the LCS interface, do not forget to assign the IP address to this interface.

Important Note

The initial release of the Network Utility does not provide the TCP pass-through support equivalent to the IBM 3172 implementation. This means that if you are replacing a 3172 with a release 1 Network Utility, you will need to configure an additional IP subnet for the virtual LAN segment inside the Network Utility. IBM intends to implement the TCP pass-through function in the next release of the Network Utility, which will remove this requirement.

Figure 14-6 on page 14-10 shows how the parameters correlate between the host and the Network Utility for an LCS interface definition.
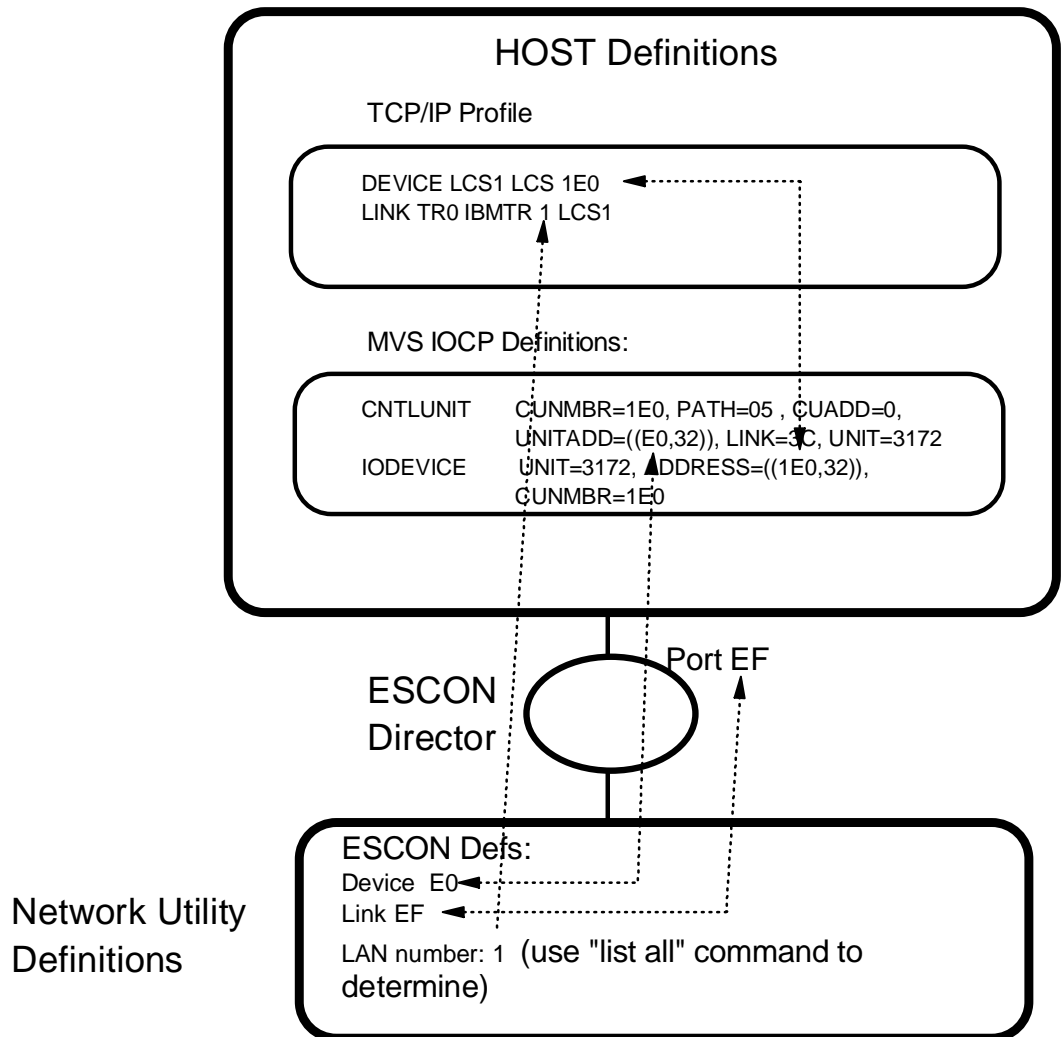
Figure 14-6. Host/Network Utility Parameter Relationships - LCS

Notes:

1. LCS uses a pair of subchannels, one for reading and one for writing. When configuring the subchannels used by the LCS interface, you actually only need to specify one subchannel address. LCS automatically assigns two adjacent subchannels for the LCS connection, one for the read (device address is odd) and one for the write (device address is even).

2. The device address specified in the Network Utility LCS interface definition must be within the range specified in the UNITADD parameter in the CNTLUNIT macro from the IOCP. For example, the UNITADD parameter in Figure 14-6 shows that 32 (decimal) device addresses starting at E0 (hex) are being reserved for the Network Utility definition. A device address of E0 has been specified for the Network Utility LCS interface. The Network Utility will automatically allocate E1 also. Since E0 and E1 are in the range between E0 and FF hex, this is OK as long as no other device (or interface on this Network Utility) tries to use these same subchannels.

3. The value specified in the DEVICE statement in the host TCP/IP profile must be within the range specified in the ADDRESS parameter in the IODEVICE macro from the IOCP. For example, the DEVICE statement in the host TCP/IP

profile in Figure 14-6 is 1E0 hex, which is in the range between 1E0 and 1FF that the ADDRESS parameter in the IODEVICE statement specifies.

4. The values specified for the ADDRESS parameter in the IODEVICE macro and the UNITADD parameter in the CNTLUNIT macro are related *by convention only*. In this example, the value for the ADDRESS parameter has been determined from the value for the UNITADD parameter by prepending a *logical channel identifier* (1 in this case) to the UNITADD value. This will often be the case. However, when defining the device address on the Network Utility LCS definition, use the UNITADD parameter and not the ADDRESS parameter to determine the valid range of values.

See Chapter 18, "Sample Host Definitions" for more explanation and samples of host definitions for this interface type.

For a complete look at the configuration parameters needed for this scenario, see Table 15-2 on page 15-3.

Parallel Channel Gateway

This scenario is shown in Figure 14-7. It is identical to the ESCON channel gateway except that the connection to the host is via a S/370 Bus and Tag (Parallel Channel) Adapter instead of an ESCON channel. Like the ESCON gateway, this configuration uses an LSA direct connection for the SNA traffic and an LCS interface for the IP traffic.
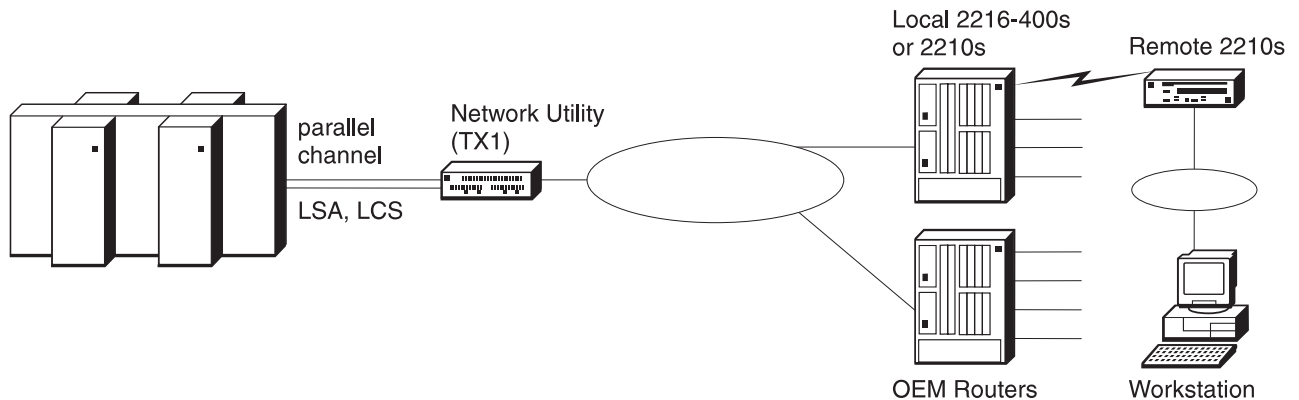
Figure 14-7. Parallel Channel Gateway

Keys to Configuration
The configuration for this scenario is very similar to that for the ESCON gateway (see "ESCON Channel Gateway" on page 14-6). The configuration of the LSA and LCS interfaces require fewer parameters since no LPAR, Link Address, or Control Unit values are required for a Bus and Tag connection. The device address is still required to identify the Network Utility on the channel.

For a complete look at the configuration parameters needed for this scenario, see Table 15-3 on page 15-7. Also, Chapter 18, "Sample Host Definitions" contains a sample of the host IOCP definition for a Network Utility with a Parallel Channel Adapter.

Channel Gateway (APPN and IP over MPC+)

This scenario is shown in Figure 14-8. Here, a Multi-Path Channel (MPC+) Group is used to transport both IP and APPN traffic between the Network Utility and the host. MPC+ uses a group of ESCON subchannels to maximize data transfer performance.
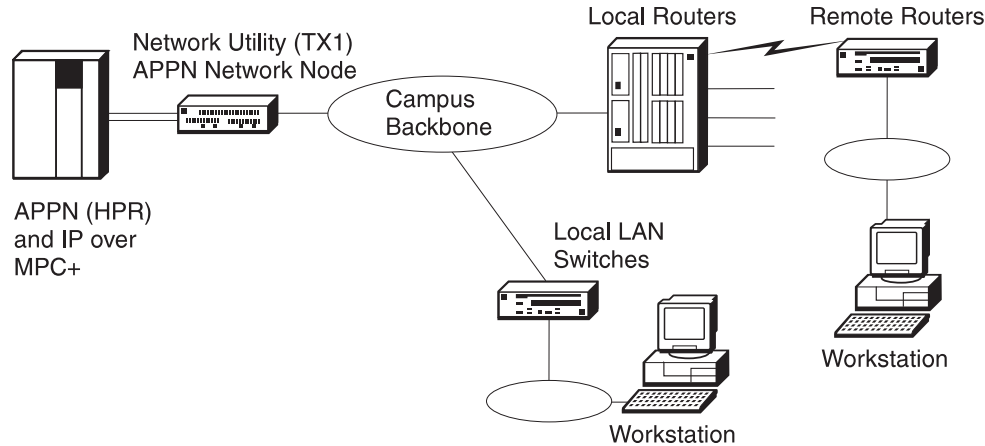
Figure 14-8. Channel Gateway (APPN and IP)

The APPN traffic coming through the Network Utility is comprised of several different types from the routers in the remote branches:

- TN3270 traffic from TN3270e servers in the branches that are configured with an APPN connection to the host. (See "Distributed TN3270e Server" on page 12-14 for an example of this type of configuration.)

- DLUR traffic from the routers in the branches that are providing support for PU 2.0 (dependent) devices.

- APPN host-to-host traffic from distributed processors (such as AS/400s) communicating with the mainframe at the central site.

In each of the above cases, the Network Utility is providing ANR forwarding only of the APPN traffic.[1] However, in addition to providing the ANR function, the Network Utility in this scenario could also be configured for TN3270e server support and DLUR support. The DLUR support could provide PU 2.0 devices on the local campus with access to the host and the TN3270e server could provide TN3270 support for workstations and printers on the local campus or for branches that do not have a distributed TN3270e server.

Keys to Configuration

Note the following when configuring the Network Utility for this scenario:

- You can either define a separate MPC+ group for your APPN and TCP/IP traffic or you can define a single group that is shared between APPN and TCP/IP.

- An MPC+ group can have as many as 32 subchannels in it. It must have at least one read and one write subchannel defined. From the talk 6 command line (from the `ESCON Add Virtual` prompt), the `sub addr` command is used to

---

[1] The RTP sessions are between the APPN nodes at each end of the conversations.

add a read subchannel while the `sub addw` command is used to add a write subchannel.

- TCP/IP is configured on an MPC+ interface the same way it is for other interfaces. Specifically, configuring an IP address for the MPC+ virtual net handler enables TCP/IP over the MPC+ interface.

- APPN is configured over the MPC+ connection the same way that it is configured for other interfaces. When you do the `add port` command, specify a port type of `M` for MPC+.

- To run APPN / HPR traffic over a MPC+ Channel, two VTAM definitions need to be created:

  - A Transport Resource List (TRL) element that defines the line control, the subchannels, the number of buffers, and the channel programs to be used.

  - A Local SNA major node with a local PU definition

- Like the LSA and LCS definitions, the subchannel parameters must match parameters used in the host definitions when defining the Network Utility to the host channel subsystem. See Table 14-1 on page 14-7 for a description of the subchannel parameters and Figure 14-9 on page 14-14 for a diagram of how these parameters correlate to the host parameters for an MPC+ definition.

Host Definitions

VTAM Start List

VTAM Local PU

VTAM Transport Resource List

NETID=STFNET

VBUILD TYPE=LOCAL
PU   TRLE=LNETU

VBUILD TYPE=TRL
LNETU TRLE

WRITE=1F0,
READ=1F1,

MVS IOCP Definitions:

CNTLUNIT   CUNMBR=1E0, PATH=05 , CUADD=0,
              UNITADD=((E0,32)), LINK=3C, UNIT=3172
IODEVICE   UNIT=3172, ADDRESS=((1E0,32)),
              CUNMBR=1E0

Port EF

ESCON
Director

Network Utility Definitions

ESCON Definitions:
Device F0 (READ)
Device F1 (Write)
Link EF
APPN Definitions:
NETID=STFNET
Control Point name=NUGW

Figure 14-9. Host/Network Utility Parameter Relationships - MPC+

Notes:

1. The device addresses specified in the Network Utility MPC+ interface definition must be within the range specified in the UNITADD parameter in the CNTLUNIT macro from the IOCP. For example, the UNITADD parameter in Figure 14-9 shows that 32 (decimal) device addresses starting at E0 (hex) are being reserved for the Network Utility definition. Device addresses of F0 and F1 have been specified for the Network Utility MPC+ interface. Since F0 and F1 are in the range between E0 and FF hex, this is OK as long as no other device (or interface on this Network Utility) tries to use these same subchannels.

2. The values specified in the VTAM TRL major node definition must be within the range specified in the ADDRESS parameter in the IODEVICE macro from the IOCP. For example, the TRL major node definition in Figure 14-9 specifies 1F0 and 1F1, which are in the range between 1E0 and 1FF that the ADDRESS parameter in the IODEVICE statement specifies.

3. The values specified for the ADDRESS parameter in the IODEVICE macro and the UNITADD parameter in the CNTLUNIT macro are related *by convention*

only. In this example, the value for the ADDRESS parameter has been determined from the value for the UNITADD parameter by prepending a *logical channel identifier* (1 in this case) to the UNITADD value. This will often be the case. However, when defining device addresses on the Network Utility MPC+ definition, use the UNITADD parameter and not the ADDRESS parameter to determine the valid range of values.

Please see Chapter 18, "Sample Host Definitions" for examples of these host definitions.

Dynamic Routing Protocols on the ESCON Interface

In a single host environment it is not necessary to run a routing protocol (RIP, for example) on the ESCON subnet. In this case, it is sufficient to add the Network Utility as the default gateway in the host TCP/IP profile.

However, if there are multiple hosts or multiple Network Utility gateways, you should consider running RIP on the ESCON interface. Running a dynamic routing protocol in this environment allows you to route around network failures if an alternate path exists.

Network Utility supports both RIP V1 and V2. RIP V2 offers variable length subnets and other advanced features that RIP V1 does not, and is the recommended choice.

Importing the ESCON Subnet into OSPF

If you are running OSPF on your network, then you should import the ESCON subnet into OSPF (unless your host TCP/IP supports OSPF). If this is not done, only workstations connected directly to an interface on the Network Utility will be able to access the TCP/IP host on the ESCON interface.

For a complete look at the configuration parameters needed for this scenario, see Table 15-4 on page 15-11.

ESCON Channel Gateway - High Availability

This scenario is shown in Figure 14-10 on page 14-16. It utilizes redundant Network Utilities, each with an ESCON channel connection to the Host. Also, the campus backbones have been duplexed and each Network Utility attaches to a different backbone.

With this configuration, you can still access the host even if you have a failure in one of the campus backbones or a Network Utility. The traffic coming in from the 2216s will still have a valid path to the host through one campus backbone and Network Utility. This is true for both IP and SNA traffic.

The ESCON Director (ESCD) is important in this configuration, especially in Parallel Sysplex environments, because it allows you to fully mesh the connections between the gateways and the LPARs in the sysplex. This provides the highest level of fault tolerance for host access.
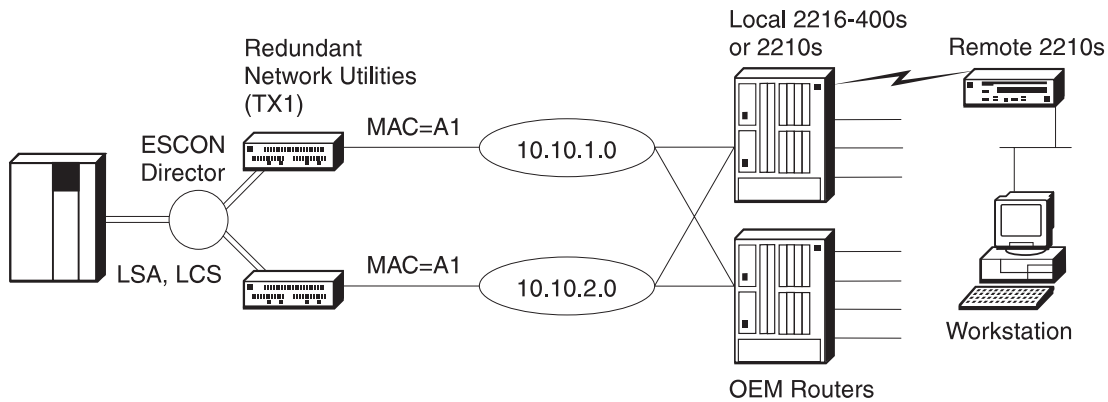
Figure 14-10. ESCON Channel Gateway - High Availability

Keys to Configuration

The configuration for this scenario is very much like the one in "ESCON Channel Gateway" on page 14-6. Each Network Utility is configured as a LAN Channel Gateway with a separate LSA and LCS interface defined on each. Please see Table 15-2 on page 15-3 to see the parameters needed for configuring a Network Utility as a LAN Channel gateway.

Since each Network Utility is on a different token ring, the same MAC address can be used for the token ring interface in each. The IP address used for each interface, however, must be different since each interface is on a different subnet.

Note: While this example shows the use of LSA and LCS connections on the ESCON channel, the use of MPC+ is equally effective in the high availability environment.

Managing the Gateway Function

The example configurations in this chapter and in "DLSw LAN Channel Gateway" on page 16-5 show several different usages of channel DLCs:

* A direct LSA interface maps to a LAN interface with no involvement from DLSw or APPN in forwarding frames

* An LCS or MPC+ virtual interface appears to the IP routing code as another interface, and IP performs its normal routing function to forward frames to other interfaces

* The loopback LSA virtual interface appears as a link to either DLSw or APPN

* An MPC+ virtual interface can appear as a link to APPN

To manage the complete range of Network Utility gateway function, you need to manage IP, DLSw, and APPN as appropriate. This section does not cover these upper-layer functions, but focuses instead on the ways you can monitor and manage channel physical and virtual interfaces.

Command-Line Monitoring

You access the talk 5 commands that show the status of channel resources hierarchically as follows:

1. From the * prompt, type **talk 5** and press **Enter** to reach the + prompt.

2. From the + prompt, type **int** and note the logical interface number for the physical ESCON or PCA interface you are interested in.

   The physical interface is commonly called the *base net*, and may have a number of LSA, LCS, or MPC+ virtual interfaces defined on top of it. The base net and all virtual interfaces each have a different logical interface number.

3. From the + prompt, type **net** *base net number* to reach the ESCON or PCA Console subprocess. The command prompt changes to ESCON> or PCA> as appropriate.

   At these prompts, you can use the **li nets** command to see the current state of every (LSA, LCS, MPC+) virtual interface using this base net. You can also type **li sub** to view the currently running subchannel configuration for this base net.

4. From the base net ESCON> or PCA> prompt, type **net** *virtual net number* to see more detail on a particular virtual interface that uses this base net. The command prompt changes to LSA>, LCS>, or MPC+>, depending on the type of the virtual interface you select.

   Each of these prompts supports a **list** command, to show configuration and current status information relevant to the virtual interface type.

5. To back out from any of these nested levels, type **exit**, and **Ctrl-p** to go back to the * prompt.

For examples and a detailed explanation of the output of these commands, see the chapter "Configuring and Monitoring the ESCON and Parallel Channel Adapters" in the *MAS Software Users Guide*.

Event Logging Support

Events occurring within the channel functions are covered by the following ELS subsystems:

ESC   Low-layer ESCON events
PCA   Low-layer Parallel Channel events
LSA   Events related to LSA virtual interfaces
LCS   Events related to LCS virtual interfaces
MPC+   Events related to MPC+ virtual interfaces

To enable event logging, type **event** from talk 5 or talk 6 to reach the ELS Console or Config subprocess. If you want the logging output to go to talk 2, type **disp sub** *subsystem name* to enable normal error reporting, or **disp sub** *subsystem name* **all** to enable all messages. To get the greatest visibility to a problem, you might enable both one of the ESCON or PCA subsystems, and one of the virtual interface subsystems. If you use these commands from talk 5, you can immediately move to talk 2 and monitor events as they occur.

You can get a feel for the events reported by each of these subsystems using the command **li sub** *subsystem name* from either the talk 5 or talk 6 ELS subprocess.

SNA Management Support

From a VTAM or NetView/390 operator console, you can control SNA resources associated with LSA direct gateway function, DLSw, or APPN as described in "NetView/390" on page 8-10.

The channel function itself does not send SNA alerts. It does not send traps that can be converted to alerts, but you can enable traps for channel ELS messages and use the products mentioned in "IBM Nways Manager for AIX" on page 8-7 to convert those traps to alerts.

SNMP MIB and Trap Support

Network Utility supports an IBM enterprise-specific MIB for ESCON. This MIB provides access to the following information:

- A list of physical interfaces and the fiber signal status of each
- A list of channel links and the host connection status of each
- A list of channel stations with both configuration and normal/error traffic statistics for each.

The ESCON MIB does not define any traps. Parallel channel functions have no MIB support.

Both ESCON and parallel channel interfaces are represented in the Interfaces MIB (RFC 1573), so a management station can access their status and basic per-interface traffic statistics. Network Utility allows a management station to control interface state, and can send traps to report when the interfaces go up or down.

Network Management Application Support

The Network Utility Java-based application discussed in "IBM Nways Manager Products" on page 8-7 provides integrated support for both the ESCON MIB and the Interfaces MIB. You can see color-coded interface status as well as specific panels that present key information from these MIBs. You can also use integrated browser support to view the information in either of these MIBs.

You can disable or enable the emission of interface up/down traps from the Nways Manager products.

Chapter 15. Channel Gateway Example Configuration Details

This chapter contains diagrams and configuration parameter tables for several of the example channel gateway network configurations in Chapter 14, "Channel Gateway." The parameter values shown are from real working test configurations.

For an explanation of the columns and conventions in the configuration parameter tables, see "Example Configuration Table Conventions" on page 11-3.

The Network Utility World Wide Web pages contain binary configuration files that match these configuration parameter tables. To access these files, follow the `Download` link from:

> `http://www.networking.ibm.com/networkutility`

The configurations documented in this chapter are:

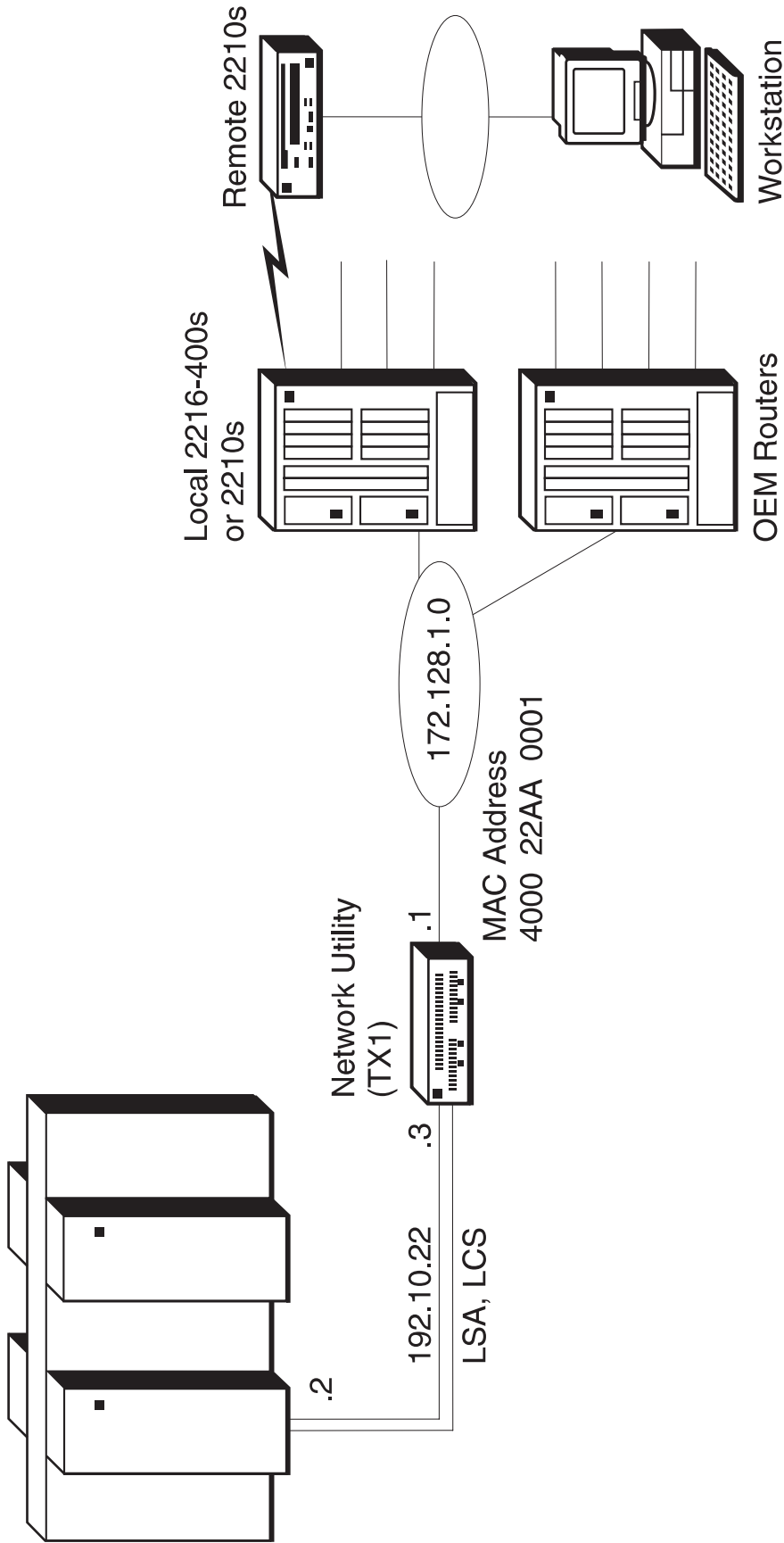| Table 15-1. Cross-Reference of Example Configuration Information	
Configuration Description	**Parameter Table**
"ESCON Channel Gateway" on page 14-6	Table 15-2 on page 15-3
"Parallel Channel Gateway" on page 14-11	Table 15-3 on page 15-7
"Channel Gateway (APPN and IP over MPC+)" on page 14-12	Table 15-4 on page 15-11
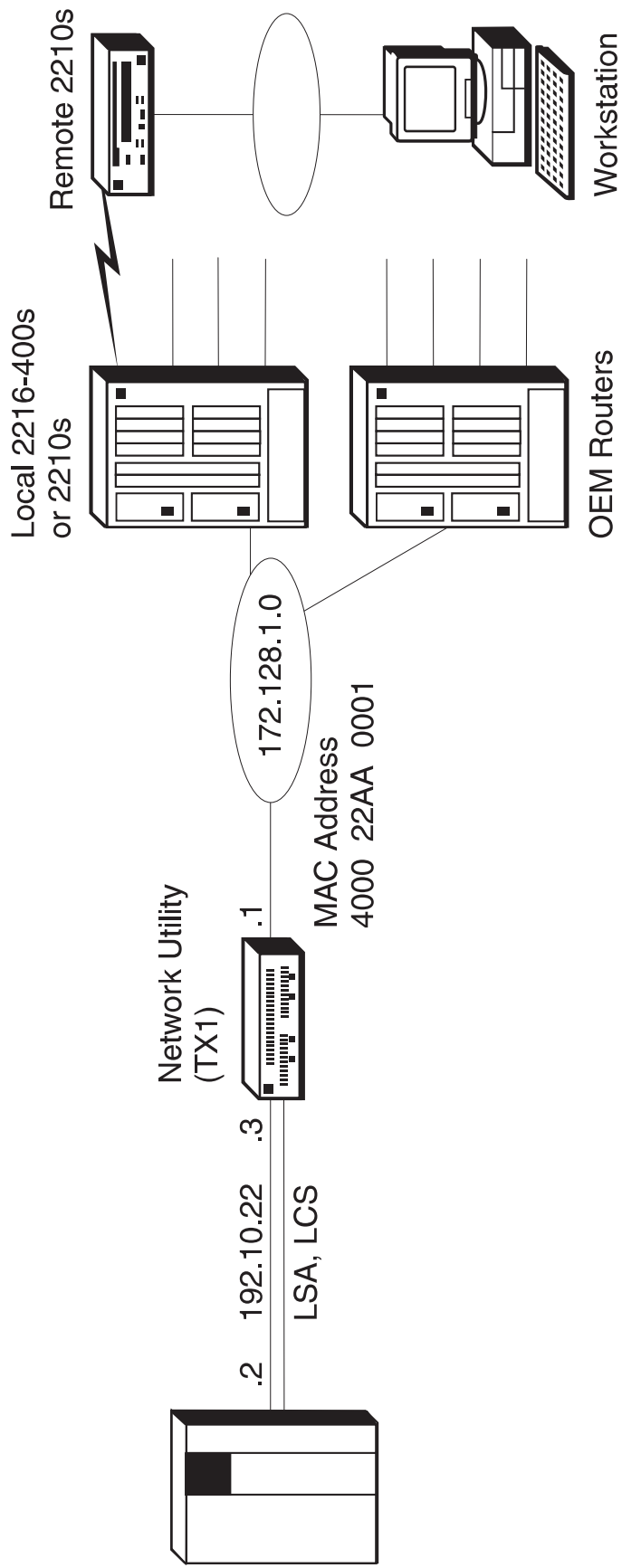
Figure 15-1. ESCON Channel Gateway

Table 15-2 (Page 1 of 3). ESCON Channel Gateway. See page 14-6 for a description and 15-2 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Devices
  Adapters
   Slots
	Slot 1: 2 Port TR
	Slot 2: ESCON
		See "add device" on next row	1

Devices
  Adapters
   Ports
	Slot 1
	  Port 1: Interface 0: TR
	Slot 2
	  Port 1: Interface 1: ESCON
		Config>
		  add dev tok
		  add dev esc
			2

Devices
  Interfaces
	Interface 0
	  MAC address: 400022AA0001
		Config>**net 0**
		TKR config>**set phy 40:00:22:AA:00:01**	

Devices
  Channel Adapters
   ESCON Interfaces
    ESCON Interfaces
	Interface 2 (new definition)
	  Base Network Number: 1
	  Protocol Type: LSA
	  Maximum Data Frame: 2052
	  LAN Net Number: 0
	  (click on "Add" to create interface 2)
		Config>**net 1**
		ESCON Config>**add lsa**
		
		(added as interface 2)
		
		ESCON Add Virtual>
		  maxdata 2052
		  net 0
		
		  (continue in same session with next row)
			3,4,5

Devices
  Channel Adapters
   ESCON Interfaces
    ESCON Subchannels
	Interface 2 (highlight LSA interface)
	  Device Address: E4
	  Link Address: EF
	  (click on "Add")
		ESCON Add Virtual>**subchannel add**
		ESCON Add LSA Subchannel>
		  device E4
		  link EF
		
		  (type two "exits" and "list all" to verify results)
			6

Devices
  Channel Adapters
   ESCON Interfaces
    ESCON Interfaces
	Interface 3 (new definition)
	  Base Network Number: 1
	  Protocol Type: LCS
	  LAN Type: Token Ring
	  Maximum Data Frame: 2052
	  MAC Address: 400022AA0009
	  (click on "Add" to create interface 3)
		Config>**net 1**
		ESCON Config>**add lcs**
		
		 (added as interface 3)
		
		ESCON Add Virtual>
		  lantype token
		  Maxdata 2052
		  mac 40:00:22:AA:00:09
		
		  (continue in same session with next row)	

Table 15-2 (Page 2 of 3). ESCON Channel Gateway. See page 14-6 for a description and 15-2 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Devices
  Channel Adapters
    ESCON Interfaces
      ESCON Subchannels
	Interface 3 (highlight LCS interface)
	  Device Address: E0
	  Link Address: EF
	  (click on "Add")
		`ESCON Add Virtual>`**`subchannel add`**
		`ESCON Config LCS Subchannel>`
		  `device E0`
		  `link EF`
		
		(type two "exits" and "list all" to verify results)
			7

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System
  SNMP Config
    General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

System
  SNMP Config
    Communities
      General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`Config>`**`p snmp`**
		`SNMP Config>`
		  `add community`
		  `set comm access write`	

Protocols
  IP
    General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		`Config>`**`p ip`**
		`IP config>`
		  `set internal 172.128.252.1`
		  `set router-id 172.128.1.1`	

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.1
	  Subnet mask: 255.255.255.0
	Interface 3 (LCS interface)
	  IP address: 192.10.22.3
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP config>`**`add address`**
		
		(once per i/f)
			8

Protocols
  IP
    OSPF
      General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**
			8

Protocols
  IP
    OSPF
      Area Configuration
        General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  IP
    OSPF
      AS Boundary Routing
	AS Boundary Routing
	  (checked to enable)
	Import direct routes
	  (checked to enable)
		`Config>`**`p ospf`**
		`OSPF config>`**`enable as`**
		  `Import direct routes`
		
		  (Accept other defaults)
			9

Protocols
  IP
    OSPF
      Interfaces
	Interface 0
	  OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		  `Interface IP address:` **`172.128.1.1`**
		  `Attaches to area:` **`0.0.0.0`**
		
		  (Accept other defaults)	

.Notes: .

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. When you select an interface of type LSA, the "LAN type" field is disabled (gets grayed out) and the "LAN net number" and "loopback" checkbox appears.

4. The "LAN number" field is disabled because a value is assigned by the router automatically. This value must be configured in the host definition for "ADAPTNO."

5. When you "Add" the interface, a new interface will be generated and it will be assigned the next available interface number.

6. The values that you enter when configuring the subchannels must match values configured at the host. See Chapter 18, "Sample Host Definitions" on page 18-1 for examples of how to match these values.

7. When you add subchannels for an LCS virtual interface, it is only necessary to define one subchannel although LCS requires two. LCS automatically uses the next subchannel in addition to the one defined here. LCS uses the even device address (E0 in this case) as the write subchannel and the odd address (E1) as the read subchannel.

8. You can also use RIP in place of OSPF.

9. You need to import direct routes into OSPF from the ESCON interface because OSPF is not enabled on the ESCON interface. Instead, the subnet on the ESCON interface is imported into OSPF in the Network Utility and then propagated to the network. This is necessary to prevent error messages from occurring at the host if the Network Utility sends the OSPF updates over the LCS connection. TCP/IP at the host does not (yet) support Link State Advertisements from an OSPF router.
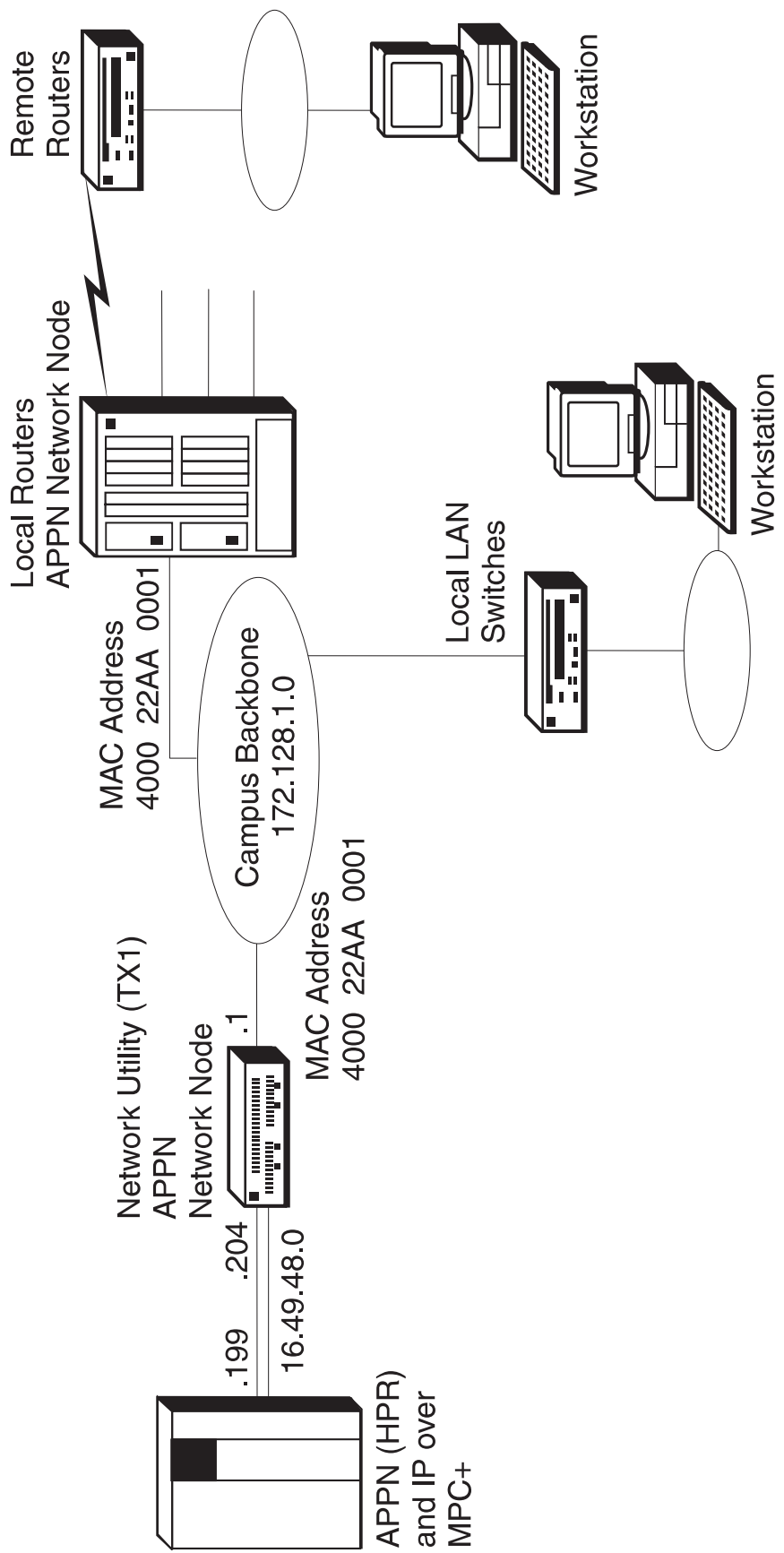
Local 2216-400s
or 2210s

Remote 2210s

Workstation

OEM Routers

172.128.1.0

MAC Address
4000 22AA 0001

Network Utility
(TX1)

.1

.3

192.10.22

LSA, LCS

.2

Figure 15-2. Parallel Channel Gateway

Table 15-3 (Page 1 of 3). Parallel Channel Gateway. See page 14-11 for a description and 15-6 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Devices
  Adapters
   Slots
	Slot 1: 2 Port TR
	Slot 2: Parallel Channel Adapter (PCA)
		See "add device" on next row	1

Devices
  Adapters
   Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 2/Port 1: Interface 1: PCA
		`Config>`
		  `add dev tok`
		  `add dev PCA`
			2

Devices
  Interfaces
	Interface 0
	  MAC address: 400022AA0001
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**	

Devices
  Channel Adapters
   PCA Interfaces
    PCA Interfaces
	Interface 2 (new definition)
	  Base Network Number: 1
	  Protocol Type: LSA
	  LAN Net Number: 0
	  (click on "Add" to create interface 2)
		`Config>`**`net 1`**
		`PCA Config>`**`add lsa`**
		
		 (added as interface 2)
		
		`PCA Add Virtual>`**`net 0`**
		
		  (continue in same session with next row)
			3,4,5

Devices
  Channel Adapters
   PCA Interfaces
    PCA Subchannels
	Interface 2 (highlight LSA interface)
	  Device Address: 00
	  Subchannel type: read/write
	  (click on "Add")
		`PCA Add Virtual>`**`subchannel add`**
		`PCA Add LSA Subchannel>`**`device 00`**
		
		  (type two "exits" and "list all" to verify results)
			6

Devices
  Channel Adapters
   PCA Interfaces
    PCA Interfaces
	Interface 3 (new definition)
	  Base Network Number: 1
	  Protocol Type: LCS
	  MAC Address: 400022AA0009
	  (click on "Add" to create interface 3)
		`Config>`**`net 1`**
		`PCA Config>`**`add lcs`**
		
		(added as interface 3):
		
		`PCA Add Virtual>`**`mac 40:00:22:AA:00:09`**
		
		  (continue in same session with next row)	

Devices
  Channel Adapters
   PCA Interfaces
    PCA Subchannels
	Interface 3 (highlight LCS interface)
	  Device Address: 02
	  Subchannel type: write
	  (click on "Add")
		`PCA Add Virtual>`**`subchannel add`**
		`PCA Add LCS Subchannel>`**`device 02`**
		
		  (type two "exits" and "list all" to verify results)
			7

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System
  SNMP Config
   General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

Table 15-3 (Page 2 of 3). Parallel Channel Gateway. See page 14-11 for a description and 15-6 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

System
 SNMP Config
  Communities
   General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`Config>`**`p snmp`**
		`SNMP Config>`
		  `add community`
		  `set comm access write`	

Protocols
 IP
  General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		`Config>`**`p ip`**
		`IP config>`
		  `set internal 172.128.252.1`
		  `set router-id 172.128.1.1`	

Protocols
 IP
  Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.1
	  Subnet mask: 255.255.255.0
	Interface 3 (LCS interface)
	  IP address: 192.10.22.3
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP config>`**`add address`**
		
		  (once per i/f)
			8

Protocols
 IP
  OSPF
   General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**
			8

Protocols
 IP
  OSPF
   Area Configuration
    General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Protocols
 IP
  OSPF
   AS Boundary Routing
	AS Boundary Routing
	 (checked to enable)
	Import direct routes
	 (checked to enable)
		`Config>`**`p ospf`**
		`OSPF config>`**`enable as`**
		  Import direct routes
		
		  (Accept other defaults)
			9

Protocols
 IP
  OSPF
   Interfaces
	Interface 0
	 OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		   Interface IP address: **172.128.1.1**
		   Attaches to area: **0.0.0.0**
		
		  (Accept other defaults)	

Table 15-3 (Page 3 of 3). Parallel Channel Gateway. See page 14-11 for a description and 15-6 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**
.Notes: .			
1. "add dev" defines a single port, not an adapter.			
2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.			
3. When you select an interface of type LSA, the "LAN type" field is disabled (gets grayed out) and the "LAN net number" and "loopback" checkbox appears.			
4. The "LAN number" field is disabled because a value is assigned by the router automatically. This value must be configured in the host definition for "ADAPTNO."			
5. When you "Add" the interface, a new interface will be generated and it will be assigned the next available interface number.			
6. The values that you enter when configuring the subchannels must match values configured at the host. See Chapter 18, "Sample Host Definitions" on page 18-1 for examples of how to match these values.			
7. When you add subchannels for an LCS virtual interface, it is only necessary to define one subchannel although LCS requires two. LCS automatically uses the next subchannel in addition to the one defined here. LCS uses the even device address (02 in this case) as the write subchannel and the odd address (03) as the read subchannel.			
8. You can also use RIP in place of OSPF.			
9. You need to import direct routes into OSPF from the PCA interface because OSPF is not enabled on the PCA interface. Instead, the subnet on the PCA interface is imported into OSPF in the Network Utility and then propagated to the network. This is necessary to prevent error messages from occurring at the host if the Network Utility sends the OSPF updates over the LCS connection. TCP/IP at the host does not (yet) support Link State Advertisements from an OSPF router.			

Remote
Routers

Workstation

Local Routers
APPN Network Node

MAC Address
4000 22AA 0001

Campus Backbone
172.128.1.0

Local LAN
Switches

Workstation

Network Utility (TX1)
APPN
Network Node

.1

MAC Address
4000 22AA 0001

.199 .204

16.49.48.0

APPN (HPR)
and IP over
MPC+

Figure 15-3. Channel Gateway (APPN & IP over MPC+)

Table 15-4 (Page 1 of 4). Channel Gateway (APPN & IP over MPC+). See page 14-12 for a description and 15-10 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**
Devices Adapters Slots	Slot 1: 2 Port TR Slot 2: ESCON	`See "add device" on next row`	1

Devices
 Adapters
  Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 2/Port 1: Interface 1: ESCON
		`Config>`
		  `add dev tok`
		  `add dev esc`
			2

Devices Interfaces	Interface 0 MAC address: 400022AA0001	`Config>`**`net 0`** `TKR config>`**`set phy 40:00:22:AA:00:01`**	

Devices
 Channel Adapters
  ESCON Interfaces
   ESCON Interfaces
	Interface 2 (new definition)
	  Base Network Number: 1
	  Protocol Type: MPC+
	  (click on "Add" to create interface 2)
		`Config>`**`net 1`**
		`ESCON Config>`**`add mpc`**
		
		 `(added as interface 2)`
		
		`ESCON Add Virtual>`
		
		  `(continue in same session with next row)`
			3

Devices
 Channel Adapters
  ESCON Interfaces
   ESCON Subchannels
	(highlight interface 2)
	  Device Address: F0
	  Link Address: EF
	  Subchannel type: Read
	   (click on "Add" to define subchannel)
	
	  Device Address: F1
	  Link Address: EF
	  Subchannel type: Write
	   (click on "Add" to define subchannel)
		`ESCON Add Virtual>`**`sub addr`**
		`ESCON Add MPC+ Read Subchannel>`
		  `dev f0`
		  `link ef`
		  `exit`
		`ESCON Add Virtual>`**`sub addw`**
		`ESCON Add MPC+ Write Subchannel>`
		  `dev f1`
		  `link ef`
		
		  `(type two "exits" and "list all" to verify results)`
			4

System
 General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System SNMP Config General	SNMP (checked)	`Config>`**`p snmp`** `SNMP Config>`**`enable snmp`**	

Table 15-4 (Page 2 of 4). Channel Gateway (APPN & IP over MPC+). See page 14-12 for a description and 15-10 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

System
 SNMP Config
  Communities
   General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		Config>**p snmp**
		SNMP Config>
		  add community
		  set comm access write
			5

Protocols
 IP
  General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		Config>**p ip**
		IP config>
		  set internal 172.128.252.1
		  set router-id 172.128.1.1	

Protocols
 IP
  Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.1
	  Subnet mask: 255.255.255.0
	Interface 2 (MPC+ interface)
	  IP address: 16.49.48.204
	  Subnet mask: 255.255.255.0
		Config>**p ip**
		IP config>**add address**
		
		  (once per i/f)	

Protocols
 IP
  OSPF
   General
	OSPF (checked)	Config>**p ospf**
		OSPF Config>**enable ospf**
			6

Protocols
 IP
  OSPF
   Area Configuration
    General
	Area number: 0.0.0.0
	Stub area (not checked)
		Config>**p ospf**
		OSPF Config>**set area**	

Protocols
 IP
  OSPF
   AS Boundary Routing
	AS Boundary Routing
	  (checked to enable)
	Import direct routes
	  (checked to enable)
		Config>**p ospf**
		OSPF config>**enable as**
		  Import direct routes
		
		  (Accept other defaults)
			7

Protocols
 IP
  OSPF
   Interfaces
	Interface 0
	  OSPF (checked)
		Config>**p ospf**
		OSPF Config>**set interface**
		    Interface IP address: **172.128.1.1**
		    Attaches to area: **0.0.0.0**
		
		  (Accept other defaults)	

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  APPN
    General
	APPN network node (checked to enable)
	  Network ID: STFNET
	  Control point name: NUGW
		`Config>`**`p appn`**
		`APPN config>` **`set node`**
		  `Enable APPN`
		  `Network ID:` **`STFNET`**
		  `Control point name:` **`NUGW`**
		
		  (Accept other defaults)	

Protocols
  APPN
    Interfaces
	(highlight Interface 0 Token Ring)
	  (click on the configure tab)
	  Define APPN port (checked to enable)
	  Port name: TR001
		`Config>`**`protocol APPN`**
		`APPN config>`**`add port`**
		  `APPN Port Link Type:` **`TOKEN RING`**
		  `Port name:` **`TR001`**
		  `Enable APPN`
		
		  (Accept other defaults)	

Protocols
  APPN
    Interfaces
	(highlight Interface 0 Token Ring)
	  (click on the Link stations tab)
	  TRTG001 (new definition)
	    General-1 Tab:
	      Link station name: TRTG001
	    General-2 Tab:
	      MAC address of adjacent node:
	        400022AA0011
	      Adjacent Node Type:
	        APPN Network Node
	  (click on "Add" to create the Link station)
		`APPN config>`**`add lin`**
		`Port name for the link station:` **`TR001`**
		`Station name:` **`TRTG001`**
		`MAC address of adjacent node:` **`400022AA0011`**
		
		  (Accept other defaults)
			8

Protocols
  APPN
    Interfaces
	(highlight Interface 2 ESCON-MPC+)
	  (click on the configure tab)
	  Define APPN port (checked to enable)
	  Port name: MPC001
		`config>`**`protocol APPN`**
		`APPN config>`**`add port`**
		  `APPN Port Link Type:` **`MPC`**
		  `Interface Number:` **`2`**
		  `Port name:` **`MPC001`**
		  `Enable APPN`
		
		  (Accept other defaults)	

Table 15-4 (Page 4 of 4). Channel Gateway (APPN & IP over MPC+). See page 14-12 for a description and 15-10 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  APPN
    Interfaces
	(highlight Interface 2 ESCON-MPC+)
	  (click on the Link stations tab)
	  MPCTG001 (new definition)
	    General-1 Tab:
	      Link station name: MPCTG001
	    General-2 Tab:
	      Adjacent Node Type:
	        APPN Network Node
	  (click on "Add" to create the Link station)
		`APPN config>`**`add lin`**
		`Port name for the link station:` **`MPC001`**
		`Station name:` **`MPCTG001`**
		`Adjacent Node Type:` **`0`** `= APPN Network Node`
		
		  (Accept other defaults)	

.Notes: .

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. When you "Add" the interface, a new interface will be generated and it will be assigned the next available interface number.

4. The values that you enter when configuring the subchannels must match values configured at the host. See Chapter 18, "Sample Host Definitions" on page 18-1 for examples of how to match these values.

5. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

6. You can also use RIP in place of OSPF.

7. You need to import direct routes into OSPF from the ESCON interface because OSPF is not enabled on the ESCON interface. Instead, the subnet on the ESCON interface is imported into OSPF in the Network Utility and then propagated to the network. This is necessary to prevent error messages from occurring at the host if the Network Utility sends the OSPF updates over the MPC+ connection. TCP/IP at the host does not (yet) support Link State Advertisements from an OSPF router.

8. The destination MAC address in this example is the local router on the right-hand side of the campus backbone in Figure 15-3 on page 15-10. This router is also configured to be an APPN network node.

Chapter 16. Data Link Switching

Overview

This section introduces Data Link Switching (DLSw) and summarizes the DLSw function implemented in Network Utility.

What is DLSw?

DLSw is an IBM-invented standard technology for transporting connection-oriented protocols, mainly SNA and NetBIOS, across IP backbone networks. DLSw routers on the edges of an IP network field link establishment requests from native SNA and NetBIOS end stations, search among peer DLSw routers for one serving the target end station, then set up a path and relay application data between the end stations through the peer router.

The protocol that flows between DLSw routers is documented in RFC 1795, "Data Link Switching: Switch to Switch Protocol". Clarifications about this protocol and multicast IP-based scalability enhancements are documented in RFC 2166, "DLSw v2.0 Enhancements".

Many DLSw implementations provide a *local DLSw* function that connects two links within a single router, as opposed to connecting them across an IP network to another DLSw router. Depending on the DLC types involved, this function may be equivalent to that of a FRAD or X.25 PAD.

Network Utility DLSw Function

The Network Utility DLSw implementation is nearly identical in function to that of the IBM 2210 and 2216 routers. It can handle the following end-station protocols:

- SNA

    - PU 4/5 to PU 2.0 (and IBM 5394 on SDLC)
    - T2.1 to T2.1
    - PU 4/5 to PU 4/5

- NetBIOS

    - Point-to-point sessions
    - Broadcast datagram traffic

- LAN Network Manager

    - LNM to bridge servers (e.g., LBS, CRS, REM)
    - LNM to 8235 intelligent hub
    - LNM to LAN Station Manager

Network Utility DLSw can communicate with end stations across the following data link control (DLC) types:

- 802.2 LLC

    LLC can be carried over any of these interface types:

    - Token-ring
    - Ethernet (10Mbps or 10/100Mbps adapters)

- FDDI
- PPP links enabled for remote bridging
- Frame relay VCs enabled for remote bridging (RFC 1490 bridged frame formats)
- ATM LAN emulation
- ATM native bridging (RFC 1483 bridged frame formats)

- SDLC

  DLSw can represent the primary station on a multipoint line, multiple secondary stations, or a single fully negotiable station on a point-to-point line.

- QLLC

  DLSw supports any combination of QLLC PVCs and SVCs on a single X.25 interface. It can handle parallel virtual circuits to the same remote DTE address, as well as incoming calls from non-configured SVCs.

- APPN

  You can configure APPN to attach to the DLSw function residing in the same Network Utility. This allows APPN to have links with any PU2.0 or T2.1 SNA end station in the DLSw network, without requiring APPN to be present in remote (especially branch office) routers.

- Channel/LSA

  DLSw supports an internal interface to the ESCON and parallel channel LSA function residing in the same Network Utility. This allow the host to have links with any SNA end station in the DLSw network, without requiring separate channel gateway and central-site DLSw router products.

With remote DLSw (across IP to another router), Network Utility DLSw supports conversion from TCP DLSw frames to any of the supported DLC types. Local DLSw is supported only for specific combinations of DLC types, as shown here:

```
          LLC     SDLC    QLLC    APPN    CHANNEL
LLC       (1)     x       x       (2)     x
SDLC      x       x       x               x
QLLC      x       x       x               x
APPN
CHANNEL   x       x       x

Notes:
1 - You should use bridging for local LLC-to-LLC connectivity.  The
      only exception supported by local DLSW is LLC to a Frame Relay
      bridge port that is configured as a Boundary Access Node (BAN) port.
2 - APPN has native support for LLC, SDLC, and QLLC, so DLSw does not
      allow APPN to reach local DLCs of these types.
```

The following list summarizes some of the other capabilities and features of IBM Network Utility DLSw.

- Dynamic compatibility to all DLSw protocol standards

  IBM DLSw supports RFC 1434+, RFC 1795 (DLSw Version 1), and RFC 2166 (DLSw Version 2). It dynamically detects the protocol level of each partner router with no pre-configuration, and can simultaneously handle partners at different protocol levels.

- Dynamic and on-demand partners

IBM DLSw supports bringing up TCP connections to configured partners only when required, as well as discovering end stations served by non-configured partners, and bringing up those TCP connections on demand.

- Multicast IP discovery

  With the simple configuration of multicast IP addresses or groups, IBM DLSw can perform multicast searches for both end stations and partners. IBM DLSw provides a number of dynamic extensions to the DLSw Version 2 standard, including resource registration and simplified group configuration.

- Traffic prioritization

  There are configuration options allowing you to control not only SNA versus NetBIOS prioritization, but also individual circuit priorities. This is in addition to the Bandwidth Reservation System's (BRS) extensive support for interface-level traffic prioritization.

- Advanced filtering and static cache entries

  IBM DLSw includes extensive support for MAC address and NetBIOS name lists and static caching, allowing you to control what links are used for searching for resources as well as which remote partners are preferred.

- Load balancing and fault tolerance

  IBM DLSw can cache multiple remote partners and select among them on the basis of neighbor priority, largest frame size support, or first to reply. You can also use the neighbor priority feature to ensure that one central-site router serves only as a backup for another.

Example Configurations

This section describes three sample configurations that use the Data Link Switching feature of the Network Utility. These configurations are:

- DLSw LAN Catcher
- DLSw LAN Channel Gateway
- DLSw X.25 Channel Gateway

DLSw LAN Catcher

This scenario is shown in Figure 16-1 on page 16-4. In this scenario, the SNA traffic in the remote sites uses DLSw to get back to the data center.

The Network Utility is in the data center on the backbone LAN segment. It is a DLSw partner with each remote router and as such requires a TCP session with each. The advantage to this approach is that all of the CPU cycles needed to manage these TCP sessions and to terminate the DLSw connections are concentrated in the Network Utility. Without the Network Utility, the local routers or the host gateway (if DLSw-capable) could be consumed by this workload.

From the host perspective, the SNA LLC2 traffic is bridged into the Network Utility from the host gateway. The host gateway is either an IBM 3745/46, an IBM 3746 with the Multi-Access Enclosure (MAE), or an IBM 2216.

You can take advantage of the 2-Port Token Ring Adapter in the Network Utility by bringing in the IP-encapsulated SNA traffic on one port and delivering LLC2 SNA traffic onto the other Token Ring port. Thus, you have twice the bandwidth

available with an additional benefit of separating the IP and SNA traffic onto separate rings. Because the Network Utility provides LLC local acknowledgements (spoofing) to the host for each LLC connection, this removes a considerable amount of traffic from the campus backbone in large network environments.
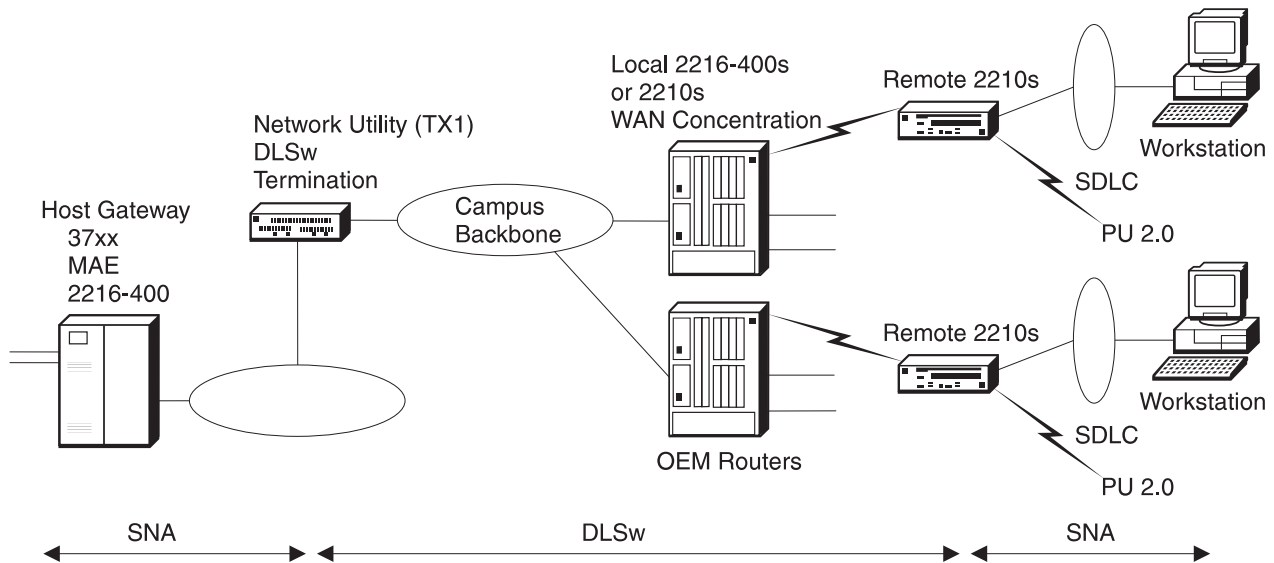
Figure 16-1. DLSw LAN Catcher

Keys to Configuration

For the most part, this is a standard DLSw configuration. However, you should be aware of the following points when configuring the Network Utility as a DLSw LAN Catcher:

- For this scenario, you should configure the Network Utility to allow TCP sessions from any of the remote routers. This is called DLSw dynamic neighbors. This keeps you from having to define the IP address of each DLSw partner on the Network Utility. The default value for dynamic neighbors is "Enabled".

- The Network Utility introduces a new parameter for IBM DLSw implementations that allows you to specify how explorer frames are forwarded. This is especially important in the outbound direction from the central site. The parameter is called *enable/disable forwarding explorers* and it gives you the flexibility to specify any of the following options:

  - Disable forwarding of explorer frames

    This option completely disables forwarding of explorer frames.

  - Forward explorer frames to the local TCP connection only

    If you want to block explorer frames from going out on WAN links, then you can specify this option. This is the default value for the Network Utility.

  - Forward explorer frames to all DLSw partners

    With this option, explorer frames are sent out to all DLSw partners.

For a complete look at the configuration parameters needed for the DLSw LAN Catcher scenario, see Table 17-2 on page 17-3.

DLSw LAN Channel Gateway

This scenario is shown in Figure 16-2. As in the DLSw LAN catcher scenario, the Network Utility terminates the DLSw sessions from the remote routers. However, in this case, there is an ESCON Channel Adapter in the Network Utility. Instead of bridging the traffic from the DLSw function onto the LAN segment, this configuration passes it directly to the channel via an LSA loopback interface configured in the Network Utility.

This configuration also demonstrates the use of the Network Utility to support SNA traffic from the local campus to the host. This traffic is bridged off the campus backbone through the LSA loopback interface. All SNA devices in the network are configured with the same host destination MAC address which is the MAC address of the LSA loopback interface. This includes the devices at the main site as well as the devices in the remote sites.
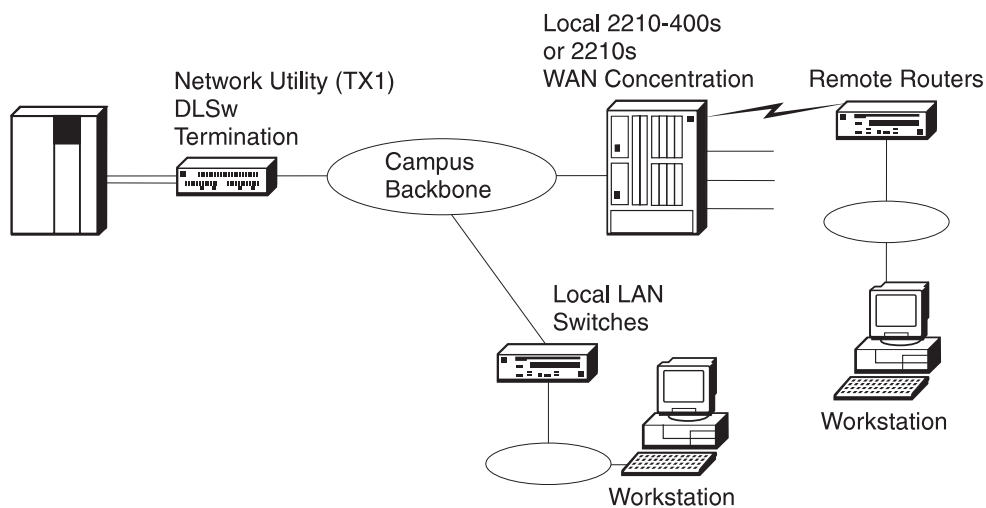
Figure 16-2. DLSw LAN Channel Gateway

Note: This example illustrates the use of the Network Utility as a channel gateway for DLSw traffic only. However, many of the functions illustrated in the Channel Gateway example configurations on page 14-6 could be combined with DLSw termination in a valid channel gateway configuration.

Keys to Configuration

Note the following points when configuring the Network Utility as a DLSw LAN Channel Gateway:

- An LSA interface must be configured and loopback must be enabled on this interface. Enabling loopback creates a virtual LAN inside the Network Utility. The only two devices on this LAN are the host and the DLSw termination point. A MAC address is defined on the LSA interface that represents the host on the channel. This is the destination MAC address that is configured in the downstream devices.

  Note: You can also define an LSA direct connection for the traffic to be bridged in from the local LAN segments. If you choose to do this, then the devices on these segments will have a different destination MAC address than the remote devices since the LSA direct interface will have a different MAC address than the LSA loopback interface.

- When configuring DLSw, you need to open SNA SAPs for the LSA interface as well as the token ring interface.

- The subchannel configuration for the LSA interface must match parameters configured in the host. See Table 14-1 on page 14-7 for a description of the subchannel parameters and Chapter 18, "Sample Host Definitions" on page 18-1 for example host definitions. This information will help you see how these parameters correlate.

- You need to configure a *local TCP connection*. This is done by defining a DLSw partner whose IP address is the internal address of the Network Utility. This is used for the traffic that is bridged from the local LAN segments into the host. This traffic gets bridged into the Network Utility into DLSw where the local TCP connection passes the traffic to the LSA loopback interface.

- The Network Utility currently supports a maximum of 2048 link stations per MAC address/SAP pair (for example, a destination MAC address of 400022AA0099 with SAP 04). If you need more then 2048 workstations, you have to define another LSA interface with a different SAP or a different MAC address. Remember that each LSA interface requires one subchannel of the 32 available on one ESCON channel adapter. You must also define the corresponding XCA major node to support each LSA interface.

X.25 Channel Gateway

This scenario is shown in Figure 16-3 on page 16-7. This scenario uses Local DLSw in the Network Utility to map between X.25 addresses and MAC address/SAP pairs. The transport across the WAN is native Qualified Logical Link Control (QLLC), a protocol that allows SNA devices to communicate over X.25 networks. In the Network Utility, local DLSw performs protocol conversion between QLLC and LLC2 frames.

From the remote device perspective, there are two cases to consider:

1. A device on a LAN segment attached to the branch router

   On the workstation, the SNA application generates an LLC frame that it wants to send to the host. If the branch router is an IBM 2210, this LLC frame gets bridged into the 2210 DLSw function, which does three things:

   a. Protocol conversion from the LLC frame to a QLLC frame
   b. Maps the destination MAC address/SAP pair into the appropriate X.25 LCN (PVC) or DTE address (SVC)
   c. Passes the QLLC frame to X.25

   The X.25 PAD function in the branch router creates the LAPB link layer packets and sends them over the PVC (or SVC).

   If some product other than the IBM 2210 plays the role of branch router, it needs to perform these same functions but may do so without using local DLSw.

2. A device directly on the X.25 network (for example, an IBM 3174 Control Unit or an eNetwork Communications Server gateway machine attached via a Wide Area Connector Adapter)

   On these devices, SNA uses QLLC as a native DLC type. It generates a QLLC frame and sends it out over the configured PVC (or SVC).

In each of these cases, at the Network Utility, the LAPB packets are received over the X.25 circuit and passed to QLLC and then on to DLSw. DLSw does two things:

1. Protocol conversion from QLLC into an LLC2 frame
2. Maps the X.25 LCN (PVC) or DTE address (SVC) into the MAC address/SAP for the LSA local loopback interface

The traffic is then passed to the LSA loopback interface for transport across the ESCON channel.
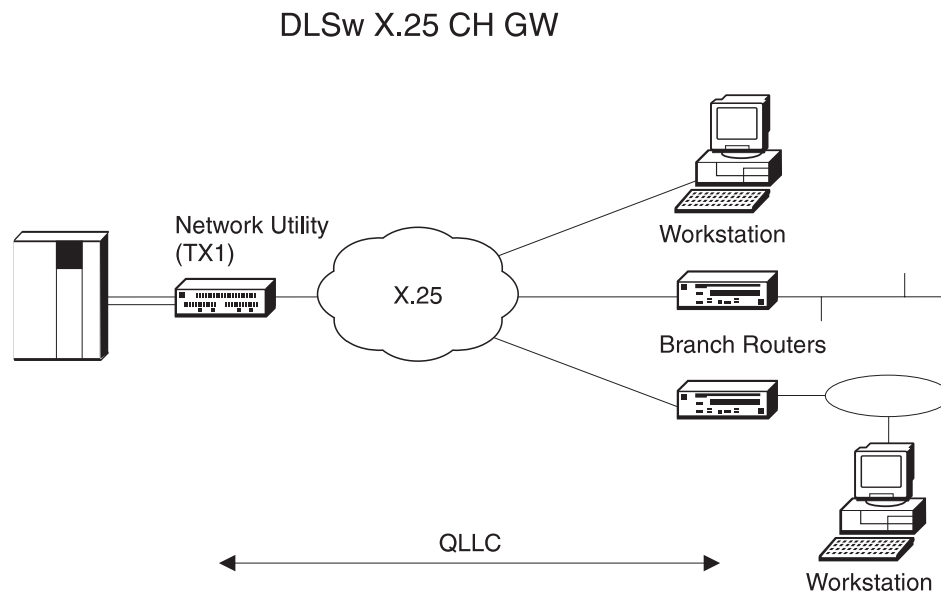
DLSw X.25 CH GW

Figure 16-3. DLSw X.25 Channel Gateway

Keys to Configuration

The following list summarizes general configuration tasks you need to perform for this scenario. Please refer to other DLSw and LSA loopback configurations for details. The LSA loopback interface is configured the same as in "DLSw LAN Channel Gateway" on page 16-5.

- Add and configure the ESCON and LSA interfaces
- Add and configure the X.25 interface. From the command line, use the **net** command in talk 6 to enter the X.25 Config subprocess, then use the following commands:
  - **set address** (to set the local DTE address)
  - **add protocol dlsw** (to add DLSw as an X.25 protocol)
  - **add pvc** or **add svc** (to add the individual PVCs or range of SVCs)
- Configure the IP internal address as in other examples
- Configure DLSw
  - Configure general DLSw (enable, SRB segment, forward explorers locally)
  - Configure the DLSw local TCP connection
  - Configure DLSw for LSA loopback (open SAPs on the LSA interface)

In addition to these general tasks, you need to configure Network Utility DLSw to map X.25 addresses to the LSA loopback MAC address. There are three ways to do this:

- Configure the X.25 stations individually at DLSw, each with its own destination MAC address. This option applies to both PVCs and SVCs.
- Configure a list of connection IDs, each of which has its own destination MAC address. Some X.25 stations can send a connection ID when they place a call, and Network Utility matches this value to the configured list. This option applies to SVCs only.
- Configure a default destination MAC address for incoming calls that do not contain a connection ID. This option applies to SVCs only.

The remainder of this section describes how to configure each of these three address mapping methods.

If the number of remote X.25 stations is relatively small, then you can configure each remote X.25 device in DLSw to be mapped to the LSA loopback MAC address. To do this using the command line, enter **talk 6** at the * prompt and type the following:

- **protocol dls**
    - **add qllc station** (once for each remote X.25 station; the system prompts you for ...)
        - interface number (X.25 interface)
        - PVC or SVC
        - logical channel number (for PVCs) or DTE address (for SVCs)
        - source MAC and SAP (can be generated by DLSw)
        - destination MAC and SAP (enter the LSA loopback MAC address)
        - PU type
        - XID block/num (if the PU type is 2)

To do this using the Configuration Program, do the following:

- Protocols/DLSw/Interfaces/Serial-X25/QLLC Stations
    - add a QLLC station (enter the same information as above)

If your remote X.25 stations can be configured to send a connection ID when they place a call,[1] you can configure DLSw to map connection ID values to destination MAC addresses. To do this using the command line, enter **talk 6** at the * prompt and type the following:

- **protocol dls**
    - **add qllc destination** (once for each valid connection ID; the system prompts you for ...)
        - connection ID
        - destination MAC and SAP (enter the LSA loopback MAC address)

To do this using the Configuration Program, do the following:

- Protocols/DLSw/QLLC Destinations
    - add a QLLC destination (enter the same information as above)

Finally, if it is not feasible to configure each remote X.25 station or to use a connection ID, you can use the DLSw ANYCALL feature to accept any incoming X.25 call and map it to the LSA loopback MAC address. To do this using the command line, enter **talk 6** at the * prompt and type the following:

- **protocol dls**

---

[1] QLLC products frequently present this parameter as a connection password.

- **add qllc destination** (once, plus you can add specific connection IDs if you wish; the system prompts you for ...)
    - connection ID (use the word 'ANYCALL')
    - destination MAC and SAP (enter the LSA loopback MAC address)

To do this using the configuration tool, do the following:

- Protocols/DLSw/QLLC Destinations
    - add a QLLC destination (enter the same information as above)

Managing DLSw

This section introduces some of the ways you can monitor and manage the DLSw function.

Command-Line Monitoring

DLSw supports an extensive set of commands to display status, dynamically modify configuration parameters, and actively control the state of connections. These commands are described in detail in *MAS Protocol Configuration and Monitoring Reference Volume 1*, in the chapter "Configuring and Monitoring DLSw". To access them, enter **talk 5** at the * prompt and **protocol dls** at the + prompt.

Some particularly useful commands for monitoring status are:

list tcp sess
  Shows the status of all known TCP connections to partner routers. You can see the state of the TCP connections as they come up and go down, as well as the level of the DLSw protocol in use, and summary statistics on the number of DLSw circuits using each connection. If you configure DLSw to accept TCP connections only from dynamic (not configured) partners, this command displays the status of connections as initiated by remote routers. There will be no status if the remote routers are not actively bringing up TCP connections.

  If you configure a "local TCP connection" to enable local DLSw function, this connection is flagged as such on the command output so that it can be distinguished from remote partner connections.

list dls sess all
  Shows the status of all active DLSw sessions. A session, also called a circuit, is defined by a MAC and SAP address 4-tuple and corresponds to an SNA link, not an SNA LU-LU session. Sessions are normally driven up and down by SNA end stations, so the output of this command is dynamic. For every session, you see its identifying MAC and SAP addresses, state, which partner the session is connected through, and an identifier that you can use with the **list dls sess detail** command to get more information. Local DLSw sessions (those that involve only this router) show as two lines of output from this command.

  Because a Network Utility may easily have hundreds or thousands of active sessions, you can use different variations of the **list dls session** command to display only a subset of them. Instead of the keyword "all", you use different keywords to show only those circuits through a given partner, or only those in a given state, and so on. There are roughly 10 keywords defined to select sessions. The output of all these commands pauses when the screen fills,

waiting for a keystroke from you to continue or quit. Press the space bar to view the next screen of output.

list dls mem
Shows the status of various pools of DLSw memory, as well as the memory congestion status for all active sessions.

list llc sess all
Shows 802.2 LLC-specific status information for all DLSw sessions that use LLC as the protocol between the router and the end station. These include sessions running over LAN, channel, ATM, and remotely bridged WAN interfaces. The command output includes more state information as well as the source route to the end station, if applicable.

list sdlc sess all
Shows SDLC-specific status information for all DLSw sessions that use SDLC as the protocol between the router and the end station. The command output includes SDLC addressing information as well as state information. If you are working with SDLC devices, this command is more useful than the generic **list dls sess**.

list qllc sess
Shows QLLC-specific status information for all DLSw sessions that use QLLC over X.25 as the protocol between the router and the end station. The command output includes QLLC addressing information as well as detailed state information. Because the router supports incoming dynamic SVCs, this command is essential to see the status of both configured and dynamic QLLC PVCs and SVCs.

DLSw supports dynamic modification under talk 5 of the vast majority of parameters you can configure under talk 6. DLSw follows the standard model where changes made under talk 5 have an immediate effect but do not survive a box reboot, while changes made under talk 6 take effect only after a box reboot. The talk 5 list commands show the values that are currently active in the running product.

The talk 5 commands **delete** and **disable** give you the power to tear down an existing DLSw connection. For example, you can use **delete dls** *session number* to clean up a hung session and allow the end stations to redrive it. **Delete/add** and **disable/enable** sequences are powerful methods to recycle configured TCP, SDLC, and QLLC connections.

Event Logging Support

DLSw has several hundred ELS messages defined, ranging from informational messages about normal events, to warnings of serious error conditions. Here are some of the types of DLSw events that can generate ELS messages:

- Initialization and configuration errors
- Partner TCP connection and capabilities frames sent or received
- Explorer frames sent or received for a particular MAC address or NetBIOS name
- Circuit setup/takedown frames sent or received
- DLC link setup/takedown frames sent or received
- Data frames sent or received on active circuits
- Pacing window changes on active circuits
- Memory allocation errors
- Unexpected protocol flows, frames discarded

- Frame flows don't match configuration

Although these messages are used primarily by software engineers to resolve problems, a user with a basic knowledge of the DLSw protocol and DLC link activation flows should be able to make sense of them and debug simple configuration mistakes. By activating these ELS messages and watching the output via talk 2, you should be able at least to answer the question "is anything happening?".

"DLS" is one of the named *subsystems* within ELS. To activate the standard set of error messages, type **disp sub dls** from the event menu under either talk 6 or talk 5. To activate all DLSw messages, enter **disp sub dls all**. The corresponding commands to deactivate messages begin with **nodisp**. For general information on controlling and viewing ELS messages, see "Monitoring Event Messages" on page 8-2.

If you are trying to trace a link activation attempt, DLSw messages alone may not show the complete picture. You can activate the ELS messages for the underlying DLC type as follows:

LLC	disp sub llc all
SDLC	disp sub sdlc all
QLLC	disp sub qllc all
	disp sub x253 all (X.25 layer 3, the packet layer)
ESCON/LSA	disp sub lsa all

Refer to *Event Logging System Messages Guide* (on CD-ROM and the 2216 web page) for a full list of individual messages and their meaning.

SNA Management Support

From a VTAM or NetView/390 operator console, you can control the links, PU, and LUs involved with DLSw as described in "NetView/390" on page 8-10.

Unlike APPN, Network Utility DLSw does not send SNA alerts. It does send traps (described below) and trigger ELS messages that can generate traps. You can use the products mentioned in "IBM Nways Manager for AIX" on page 8-7 to convert those traps to alerts.

SNMP MIB and Trap Support

Network Utility DLSw provides full read-only and partial read-write support for the IETF standard DLSw MIB documented in RFC 2024. This large MIB gives visibility to most of the important configuration, status, and accounting information that products implementing RFC 1795 and 2166 should have. This information includes:

- Configuration

  - Node characteristics, for example, dynamic partners are enabled
  - Configured partner information
  - Configured directory/cache entries

- Status

  - Node up or down, for how long
  - Active TCP connections, for how long, dynamic partner information
  - Dynamic directory/cache information

       – Active circuits, for how long, DLC information

- Statistics and Accounting

      – Counts of TCP connections up and down (normal and error)
      – Data and control frames counts per partner
      – Counts of circuits up and down
      – Indices to underlying DLC MIBs for per-circuit frame counts
      – Pacing counts for active circuits

Network Utility DLSw supports all the traps defined in RFC 2024, reporting the following events:

- A TCP connection is terminated due to capabilities exchange failure or a DLSw protocol violation
- A TCP connection comes up or goes down
- A circuit comes up or goes down

DLSw all supports trap control data items so a management station can set the conditions under which a trap is generated.

Network Management Application Support

The Network Utility Java-based application implemented in the Nways Manager products discussed in "IBM Nways Manager Products" on page 8-7 provides integrated support for the standard DLSw MIB.

To view DLSw resources and their status using these products, you bring up specific panels that present key information from the DLSw MIB and from its underlying DLC-layer MIBs (LLC, SDLC, or X.25). You can also use integrated browser support to view the information in any of these MIBs.

You can control the emission of DLSw traps from the Nways Manager products, so a given trap is generated always, never, or only under certain conditions.

Nways Manager for AIX can show you a DLSw topology view of your network, including DLSw connectivity, resources, and color-coded status. The topology is refreshed as new nodes are discovered. This application does not present the topology of DLSw IP multicast groups.

Chapter 17. DLSw Example Configuration Details

This chapter contains diagrams and configuration parameter tables for several of the example DLSw network configurations in Chapter 16, "Data Link Switching." The parameter values shown are from real working test configurations.

For an explanation of the columns and conventions in the configuration parameter tables, see "Example Configuration Table Conventions" on page 11-3.

The Network Utility World Wide Web pages contain binary configuration files that match these configuration parameter tables. To access these files, follow the `Download` link from:

   `http://www.networking.ibm.com/networkutility`

The configurations documented in this chapter are:

Table 17-1. Cross-Reference of Example Configuration Information	
Configuration Description	**Parameter Table**
"DLSw LAN Catcher" on page 16-3	Table 17-2 on page 17-3
"DLSw LAN Channel Gateway" on page 16-5	Table 17-3 on page 17-8
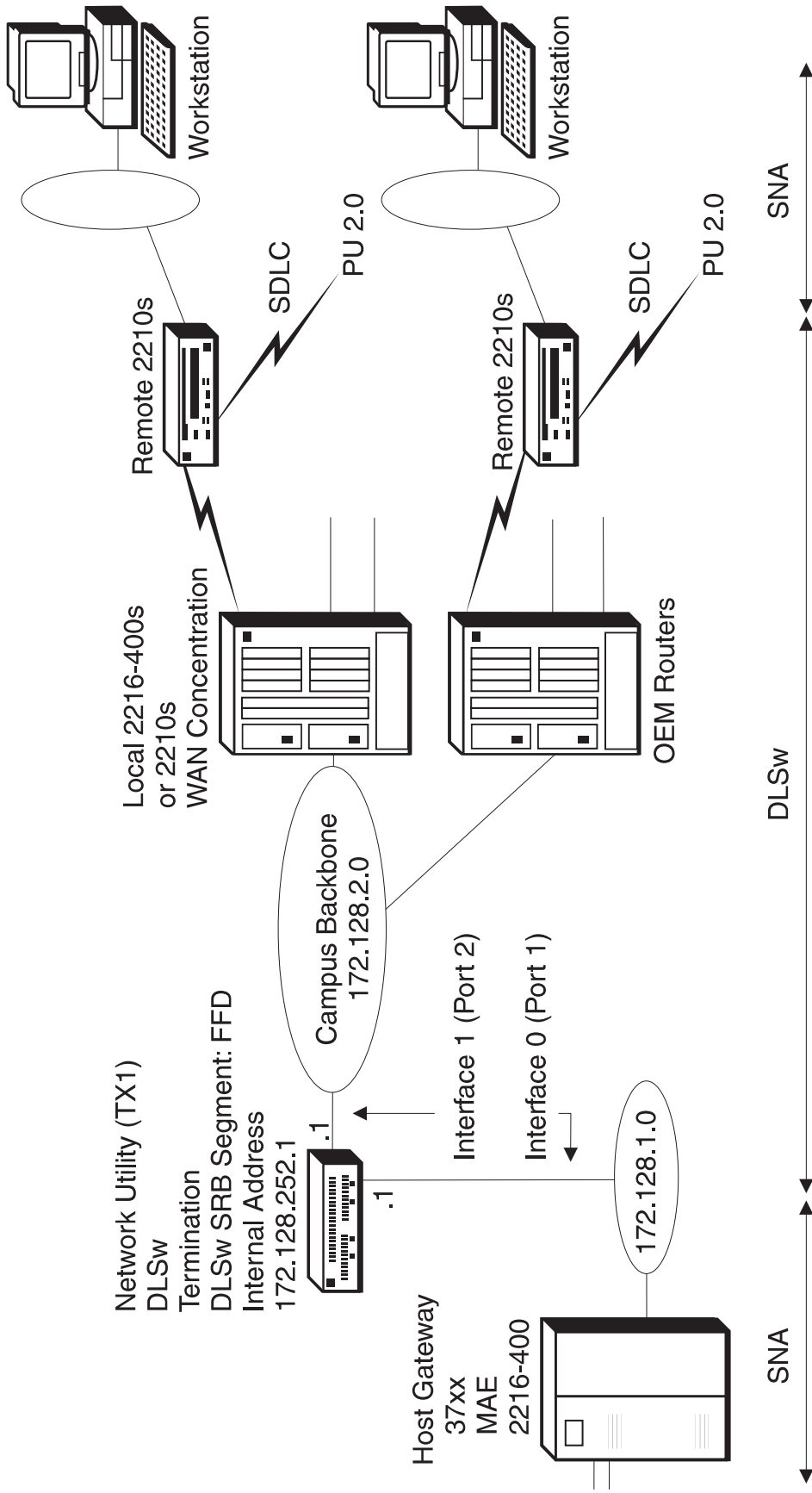
Figure 17-1. DLSw LAN Catcher

Table 17-2 (Page 1 of 4). DLSw LAN Catcher. See page 16-3 for a description and 17-2 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Devices
  Adapters
    Slots
	Slot 1: 2 Port TR	See "add device" on next row	1

Devices
  Adapters
    Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 1/Port 2: Interface 1: TR
		`Config>`**`add dev tok`**
		
		  (once per port)
			2

Devices
  Interfaces
	Interface 0
	  MAC address: 400022AA0001
	  Packet size: 4399
	Interface 1
	  MAC address: 400022AA0002
	  Packet size: 4399
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**
		  `packet 4399`
		`Config>`**`net 1`**
		`TKR config>`**`set phy 40:00:22:AA:00:02`**
		  `packet 4399`	

System
  General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System
  SNMP Config
    General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

System
  SNMP Config
    Communities
      General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`Config>`**`p snmp`**
		`SNMP Config>`
		  `add community`
		  `set comm access write`
			3

Protocols
  IP
    General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		`Config>`**`p ip`**
		`IP config>`
		  `set internal`
		  `set router-id`	

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.1
	  Subnet mask: 255.255.255.0
	Interface 1 (TR slot 1 port 2)
	  IP address: 172.128.2.1
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP config>`**`add address`**
		
		  (once per i/f)
			4

Table 17-2 (Page 2 of 4). DLSw LAN Catcher. See page 16-3 for a description and 17-2 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  IP
    OSPF
      General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**
			5

Protocols
  IP
    OSPF
      Area Configuration
        General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Protocols
  IP
    OSPF
      Interfaces
	Interface 0
	  OSPF (checked)
	Interface 1
	  OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>` **`set interface`**
		    `Interface IP address` **`172.128.1.1`**
		    `Attaches to area` **`0.0.0.0`**
		
		  (Accept other defaults)
		
		`set interface`
		    `Interface IP address` **`172.128.2.1`**
		    `Attaches to area` **`0.0.0.0`**
		
		  (Accept other defaults)	

Protocols
  DLSw
    General
      General
	DLSw (checked)
	SRB segment: FFD
	Forward explorers: disabled
		`Config>`**`p dls`**
		`DLSw Config>`
		  `enable dls`
		  `set srb`
		  `disable forward all`
			6

Protocols
  DLSw
    General
      Dynamic Neighbors
	Dynamic neighbors (checked)	`Config>`**`p dls`**
		`DLSw Config>`**`enable dynamic`**
			7

Protocols
  DLSw
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  SAP type: SNA (SAPs 0,4,8,C)
		`Config>`**`p dls`**
		`DLSw Config>`**`open 0 sna`**
			8

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  Bridging
    General
	Bridging (checked)
	DLSw (checked)
		`Config>`**`p asrt`**
		`ASRT config>`
		  `enable br`
		  `enable dls`
			9

Protocols
  Bridging
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  Bridging port (checked)
	  Interface supports: SRB
	  Segment number: 001
	  MTU size: 4399
		`Config>`**`p asrt`**
		`ASRT config>`
		
		  ('enable br' assumed)
		
		  `disable transp 1`
		  `enable source 1`
		  `delete port 2`
			10

Table 17-2 (Page 3 of 4). DLSw LAN Catcher. See page 16-3 for a description and 17-2 for a diagram of this configuration.

Table 17-2 (Page 4 of 4). DLSw LAN Catcher. See page 16-3 for a description and 17-2 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Notes:

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

4. Only Interface 1 needs to be configured for IP for DLSw to function correctly in this example. Interface 0 is configured here for IP, solely for box management purposes.

5. You can also use RIP in place of OSPF.

6. We disable the forwarding of remote explorers as a general filter, to prevent backbone LAN traffic from generating DLSw search messages on the WAN links to remote sites. This means that all circuits must be initiated by the remote routers. If your network requires the host to be able to initiate connections out to the remote sites, change this parameter to "forward to all DLSw peers".

   If the remote routers are IBM routers, you can configure them individually to control which search messages they want to receive, using MAC address and NetBIOS name lists. You can also configure whether each will bring up its TCP connection to Network Utility all the time or drop it when unused, using the *connectivity setup type* parameter.

7. Having dynamic neighbors enabled is the default value, so you do not have to change this panel or issue this command. We show it here to point out that this is the parameter that allows remote DLSw partners (neighbors) to establish TCP connections to this Network Utility without you having to define their IP addresses here. Each remote router needs to be configured with this Network Utility's internal IP address (172.128.252.1) as its partner address.

8. SAPs do not need to be opened on Interface 1 since that interface is only carrying IP traffic and not LLC traffic.

9. "enable br" automatically creates TB bridge ports for both token-ring interfaces. Bridge port numbers are 1 and 2, and are independent of adapter port numbers.

10. The disable and enable commands change bridge port 1 from TB to SRB. The "delete port" command turns off bridging on interface 1 (bridge port 2). Bridging would be required on this interface if we needed to support local end station traffic bridging from the Campus Backbone to the host.

Remote Routers

Workstation

Local 2210-400s
or 2210s
WAN Concentration

Local LAN
Switches

Workstation

Campus Backbone
172.128.1.0

Network Utility (TX1)
DLSw
Termination

.1

LSA Loopback
MAC Address:
4000 22AA 0099
Internal Address:
172.128.252.1
DLSw SRB Segment: FFD
Internal Virtual Segment: FF0
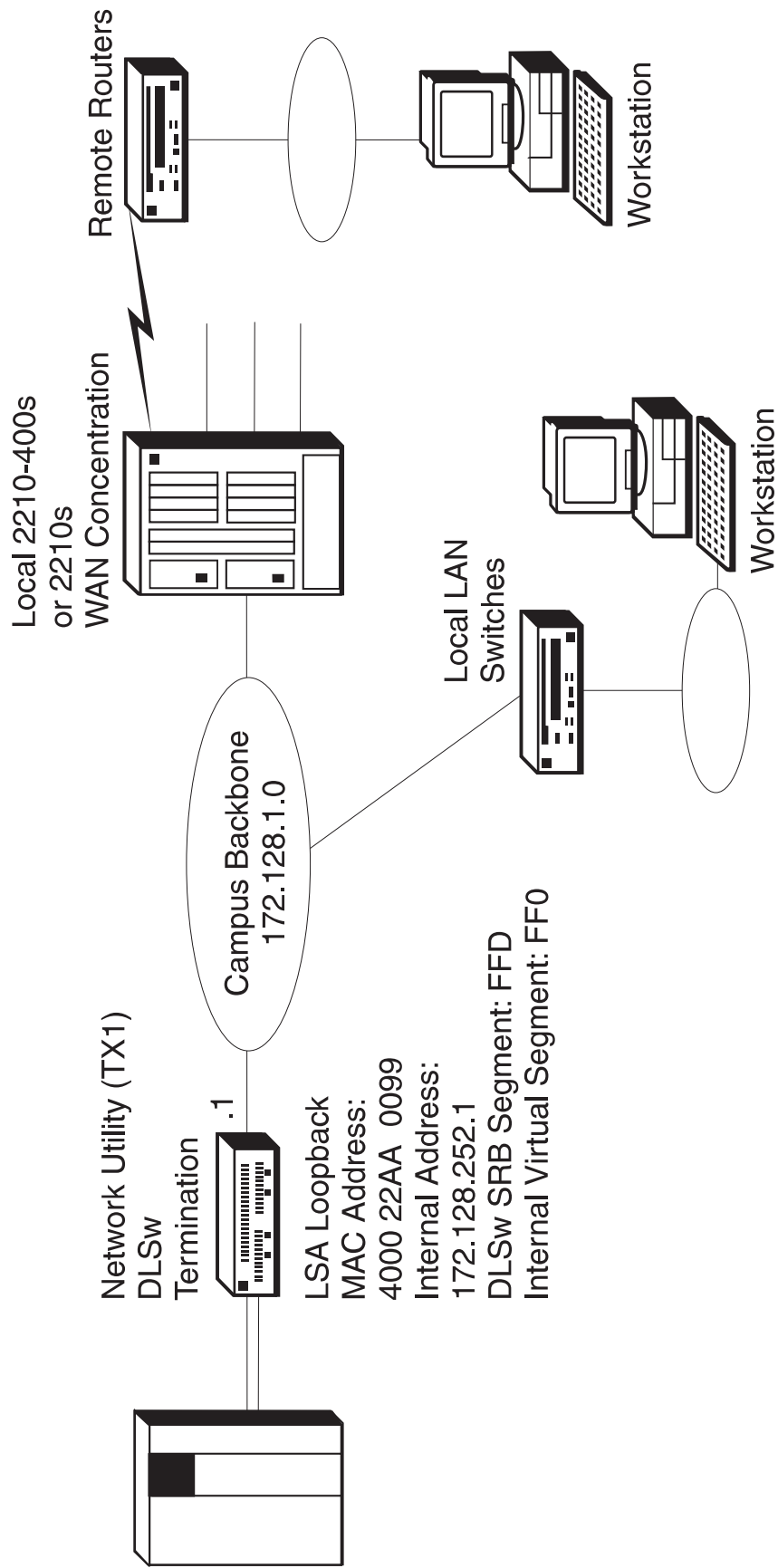
Figure 17-2. DLSw LAN Gateway

Table 17-3 (Page 1 of 4). DLSw LAN Gateway. See page 16-5 for a description and 17-7 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Devices
  Adapters
   Slots
	Slot 1: 2 Port TR
	Slot 2: ESCON
		See "add device" on next row	1

Devices
  Adapters
   Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 2/Port 1: Interface 1: ESCON
		`Config>`
		  `add dev tok`
		  `add dev escon`
			2

Devices
  Interfaces
	Interface 0
	  MAC address: 400022AA0001
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**	

Devices
  Channel Adapters
   ESCON Interfaces
    ESCON Interfaces
	Base network number: 1
	Protocol type: LSA (do this first)
	Loopback (checked - do this second)
	LAN type: Token Ring
	Maximum data frame: 2052
	MAC address: 400022AA0099
		`Config>`**`net 1`**
		`ESCON Config>`**`add lsa`**
		
		(added as interface 2)
		
		`ESCON Add Virtual>`**`enable loopback`**
		  `mac 40:00:22:AA:00:99`
		  `lan tok`
		  `maxdata 2052`
		
		  (continue in same session with next row)
			3

Devices
  Channel Adapters
   ESCON Interfaces
    ESCON Subchannels
	Interface 2, Base net 1, Protocol LSA
	  Device address: E4
	  Subchannel type: read/write
	  Link address: EF
		`ESCON Add Virtual>`**`subchannel add`**
		
		  (cont'd)
		
		`ESCON Add LSA Subchannel>`
		  `device E4`
		  `link EF`
		
		  (type two "exit"s and "list all" to verify results)	

System
  General
	System name: NUA_SC1C
	Location: XYZ
	Contact: Admin
		`Config>`
		  `set host`
		  `set location`
		  `set contact`	

System
  SNMP Config
   General
	SNMP (checked)	`SNMP Config>`**`enable snmp`**	

System
  SNMP Config
   Communities
    General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`Config>`**`p snmp`**
		`SNMP Config>`
		  `add community`
		  `set comm access write`
			4

Table 17-3 (Page 2 of 4). DLSw LAN Gateway. See page 16-5 for a description and 17-7 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**

Protocols
  IP
    General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		`Config>`**`p ip`**
		`IP config>`
		  `set internal`
		  `set router-id`	

Protocols
  IP
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  IP address: 172.128.1.1
	  Subnet mask: 255.255.255.0
		`Config>`**`p ip`**
		`IP config>`**`add address`**	

Protocols
  IP
    OSPF
      General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**
			5

Protocols
  IP
    OSPF
      Area Configuration
        General
	Area number: 0.0.0.0
	Stub area (not checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set area`**	

Protocols
  IP
    OSPF
      Interfaces
	Interface 0
	  OSPF (checked)
		`Config>`**`p ospf`**
		`OSPF Config>`**`set interface`**
		    `Interface IP address` **`172.128.1.1`**
		    `Attaches to area` **`0.0.0.0`**
		
		(Accept other defaults)	

Protocols
  DLSw
    General
      General
	DLSw (checked)
	SRB segment: FFD
	Forward explorers: local TCP connection only
		`Config>`**`p dls`**
		`DLSw Config>`
		  `enable dls`
		  `set srb`
		  `enable forward local`
			6

Protocols
  DLSw
    General
      Dynamic Neighbors
	Dynamic neighbors (checked)	`Config>`**`p dls`**
		`DLSw Config>`**`enable dynamic`**
			7

Protocols
  DLSw
    TCP Connections
	(add)
	Neighbor IP address: 172.128.252.1
	  (this is the router internal IP address)
		`Config>`**`p dls`**
		`DLSw Config>`**`add tcp`**
		    `DLSw neighbor IP address:` **`172.128.252.1`**
		
		(Accept other defaults)
			8

Table 17-3 (Page 3 of 4). DLSw LAN Gateway. See page 16-5 for a description and 17-7 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  DLSw
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  SAP type: SNA (SAPs 0,4,8,C)
	Interface 2 (ESCON-LSA)
	  SAP type: SNA (SAPs 0,4,8,C)
		Config>**p dls**
		DLSw Config>
		  open 0 sna
		  open 2 sna
			9

Protocols
  Bridging
    General
	General Tab:
	  Bridging (checked)
	  DLSw (checked)
	SRB Tab:
	  Internal Virtual Segment: FF0
		Config>**p asrt**
		ASRT config>
		  enable br
		  enable dls
			10

Protocols
  Bridging
    Interfaces
	Interface 0 (TR slot 1 port 1)
	  Bridging port (checked)
	  Interface supports: SRB
	  Segment number: 001
	  MTU size: 2052
		Config>**p asrt**
		ASRT config>
		
		  ('enable br' assumed)
		
		  disable transp 1
		  enable source 1
			11

Table 17-3 (Page 4 of 4). DLSw LAN Gateway. See page 16-5 for a description and 17-7 for a diagram of this configuration.			
Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**
Notes:			

Notes:

1. "add dev" defines a single port, not an adapter.

2. The configuration program assigns an "interface number" to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the "add dev" command for each port you want to use, and the interface number (also known as "net number") is the output of the command.

3. The MAC address representing this LSA loopback interface is the target MAC address that all end stations in the DLSw network use to reach the host through this Network Utility.

4. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.

5. You could choose to use RIP in place of OSPF.

6. We enable local forwarding to allow end stations on the local campus to reach the host. We disable the forwarding of remote explorers as a general filter, to prevent backbone LAN traffic from generating DLSw search messages on the WAN links to remote sites. This means that all remote circuits must be initiated by the remote routers. If your network requires the host to be able to initiate connections out to the remote sites, change this parameter to "forward to all DLSw peers".

   If the remote routers are IBM routers, you can configure them individually to control which search messages they want to receive, using MAC address and NetBIOS name lists. You can also configure whether each will bring up its TCP connection to Network Utility all the time or drop it when unused, using the *connectivity setup type* parameter.

7. Having dynamic neighbors enabled is the default value, so you do not have to change this panel or issue this command. We show it here to point out that this is the parameter that allows remote DLSw partners (neighbors) to establish TCP connections to this Network Utility without you having to define their IP addresses here. Each remote router needs to be configured with this Network Utility's internal IP address (172.128.252.1) as its partner address.

8. Adding the internal IP address as a neighbor is required to enable DLSw to carry traffic from the ESCON/LSA interface to the backbone LAN.

9. SAPs are opened on Interface 0 to enable the LLC flow to the local LAN switches, and are not required for remote DLSw to work.

10. "enable br" automatically creates a TB bridge port for the token-ring interface. The bridge port number is 1, and is independent of adapter port numbers and box interface numbers.

11. The disable and enable commands change bridge port 1 from TB to SRB. Bridging is required on this interface to support local end station traffic looping through DLSw from the Campus Backbone to the host.

Chapter 18. Sample Host Definitions

This appendix contains examples of host definitions for the Network Utility in the configurations used in this manual.

Specifically, definitions for the following environments are presented:

- LSA
- LCS
- MPC+

Additionally, the differences between a Network Utility with an ESCON channel adapter and a parallel channel adapter are highlighted.

For more information on defining the Network Utility to the host, refer to the *IBM 2216 Nways Multiaccess Connector Software User's Guide*, SC30-3886.

Overview

There are three steps to define a channel-attached Network Utility to the host:

1. Define the Network Utility to the host channel subsystem

   This will be done either from the I/O configuration program (IOCP) or the Hardware Configuration Definition (HCD), depending on your MVS version. (HCD requires MVS/ESA SP version 4.2 or later with APAR# OY67361.)

   The definition statements are slightly different for an ESCON channel-attached device than for a parallel channel-attached device. An example of these definitions is given in "Sample Host IOCP Definitions" on page 18-2.

2. Define the Network Utility as a control unit to the host operating system

   For most systems, the definitions are the same for an ESCON adapter as they are for a parallel channel adapter. Obviously, they depend on the operating system being used. An example of these definitions is given in "Defining the Network Utility in the Operating System" on page 18-5.

3. Define the Network Utility to the host TCP/IP or VTAM

   These definitions depend on whether you are defining an LSA (SNA), an LCS (TCP/IP), or an MPC+ (SNA and/or TCP/IP) interface on the Network Utility. Section "VTAM Definitions" on page 18-6 shows examples of the required VTAM definitions. Section "Host IP Definitions" on page 18-15 shows examples of the required TCP/IP definitions.

Definitions at the Channel Subsystem Level

You make definitions at this level via the IOCP or with HCD. If HCD is available, you will probably want to use it. HCD offers an improved method of defining system hardware configuration. With HCD several complex steps required for entering hardware configuration data can be accomplished using an interactive dialog. This chapter only presents the IOCP macros that would be generated from HCD.

Sample Host IOCP Definitions

An example of the definitions required in the host I/O Configuration Program (IOCP) for a Network Utility configured with an ESCON adapter is shown in Figure 18-1.

```
CHPID                   PATH=((05)),TYPE=CNC
CNTLUNIT                CUNUMBR=1E0,PATH=05,CUADD=0,
                        UNITADD=((E0,32)),LINK=3C,UNIT=3172
IODEVICE                UNIT=3172,ADDRESS=((1E0,32)),
                        CUNUMBR=1E0
```

Figure 18-1. Sample Host IOCP Definitions for the Network Utility (ESCON)

The following sections describe the IOCP macros that you need for defining the Network Utility at the host.

RESOURCE Statement

This identifies the host logical partitions (LPARs) by name and number. This statement is not present if the host is not partitioned *as is the case in the example above*.

- PART=((name1,x),(name2,y)...(nameX,z))

  The name identifies the LPAR and is used in the rest of the channel path definition. The number is the corresponding LPAR number. The LPAR number is used in defining the subchannel on the Network Utility. If the host is not partitioned, the LPAR number is always 0.

Channel Path ID (CHPID) Statement

The CHPID identifies the type of channel connection and who uses it.

- PATH=x

  This uniquely identifies the channel path. This value is often called the "CHPID number".

- TYPE=CNC

  This indicates that the channel is an ESCON channel. The channel type is CNC for ESCON and BL for block multiplexor (Parallel Channel Adapter).

- SWITCH=x

  This identifies which ESCON Director is in this path. If no director is being used, this parameter is omitted.

- SHARED

  This indicates that the CHPID can be used by multiple LPARs simultaneously. If not present, only one LPAR can use the CHPID at a time.

- PARTITION=(name1,name2,...,nameX)

  This is one form of the PARTITION parameter and it contains an access list of LPARS that indicates which partitions have access to this channel. The names must be included in the RESOURCE statement.

- PARTITION=((name1,...,nameX),(name2,...,nameY))

  This is the other form of the PARTITION parameter. In this form, the first grouping of names is the access list of LPARs, as above. The second

grouping is the list of candidate LPARs that an operator could configure to have access to the channel. The second grouping will have at least the same LPARs as the first grouping and it may specify additional LPARs also.

Control Unit (CNTLUNIT) Statement

This statement, along with the IODEVICE statement, defines the path from the host to the Network Utility. The CNTLUNIT and IODEVICE statements occur in pairs. If multiple LPARs are being defined to use a single CHPID, there must be a CNTLUNIT and IODEVICE statement for each LPAR.

- CUNUMBR=x

  This is an identifier for the control unit definition.

- PATH=x

  This number identifies the CHPID being used.

- UNIT=3172

  This identifies the type of control unit at the other end of the channel. The value is always 3172 when talking to a Network Utility. The IBM 3172 was the predecessor of the Network Utility ESCON channel function.

- CUADD=x

  This value identifies the control unit address of the Network Utility. The default is 0. For the Network Utility, each LPAR on a given CHPID must have a unique CUADD value. Usually (but not always) the CUADD value will be chosen to match the LPAR number.

- UNITADD=((addr,number))

  This defines the range of addresses reserved for this control unit. The first number is the hex address of the first subchannel assigned to this control unit. The second number is the decimal number of subchannels being assigned to this control unit. The example above defines a maximum of 32 control unit addresses, or subchannels, starting from E0 (HEX) and going upwards. The device address(es) specified on the Network Utility LCS, LSA, or MPC+ interface definition must be from within this range. The Network Utility can use a maximum of 32 subchannels.

- LINK=xx The value for the LINK parameter should be set to the port of the ESCON Director (ESCD) that the *Network Utility* is attached to. Since the ESCD is a switch, you can think of the link parameter as the phone number that the host will use to reach the Network Utility through the switch.

IODEVICE Statement

This statement, along with the CNTLUNIT statement, identifies the Network Utility connection to the host.

- ADDRESS=(addr,number)

  This parameter identifies the range of addresses to the rest of the host. As in the CNTLUNIT statement, the first number is the hex address *being assigned* to the first address reserved, and the second number is the decimal number of subchannels reserved. This address is different from the UNITADD. It is used in the TCP/IP profile (for LCS), the VTAM XCA Major Node Definition (for LSA), and the VTAM TRL (for MPC+) to identify the subchannels being used.

- CUNUMBR=x

This identifies the corresponding CNTLUNIT statement to this IODEVICE statement. While the value for this parameter has to be the same for both the CNTLUNIT and the IODEVICE macros, it does not have to relate to any other parameter. It is a good idea, however, to make it the same value specified in the ADDRESS parameter in the IODEVICE macro. The value for CUNUMBR has no significance outside the Channel Path Definition.

- UNIT=3172

  This identifies the type of device that is downstream. It should always be 3172 if the control unit is a Network Utility. The IOCP software in the host does not look at this field. If you are migrating from an IBM 3172 to the Network Utility, you might have a value of UNIT=SCTC in the existing IOCP statement. This is should be changed to 3172 for the Network Utility.

- PARTITION=(name)

  This is the device candidate list and it contains a list of one or more LPARs that have access to the device. This list is a subset of the list of LPARs specified in the CHPID statement and it is used to restrict which LPARs in the channel candidate list are allowed to use these devices. If the host is not partitioned, this field will not be present.

Figure 18-2 shows an example of the IOCP statements for defining a Network Utility with a Parallel Channel Adapter (PCA).

```
CHPID              PATH=((05)),TYPE=BL
CNTLUNIT           CUNUMBR=640,PATH=05
                   PROTOCL=S4,UNIT=3172
                   SHARED=N,UNITADD=((40,32))
IODEVICE           UNIT=3172,ADDRESS=((640,32))
                   STADET=N,CUNUMBR=640,TIMEOUT=Y
```

Figure 18-2. Sample Host IOCP Definitions for the Network Utility (PCA)

Please note the following points concerning the IOCP statements for a Network Utility with a PCA.

- The TYPE is BL for Block Multiplexer

- PROTOCL parameter can be set to the following values, depending on the device capability:

  D Direct-Coupled Interlock (DCI) mode

  S Maximum 3.0 MB data streaming speed

  S4 Maximum 4.5 MB data streaming speed

  For the Network Utility, set the value to S4. The transfer mode and channel parameter must conform with the PCA setting for transfer mode and channel transfer speed.

- The UNIT parameter on the CNTLUNIT and IODEVICE statements must be set to 3172.

- When an ESCON Converter is the channel path, the CHPID TYPE parameter must be set to CVC, otherwise it is set to BL.

Defining the Network Utility in the Operating System

The following sections describe the definitions needed for various host operating systems.

Network Utility Definition for VM/SP

The Network Utility must be defined to a VM/SP operating system. This definition is accomplished by updating the real I/O configuration file (DMKRIO) with entries for the Network Utility in the RDEVICE and the RCTLUNIT macros. In the following example, 640 is the base unit address and the size of the address range is 32.

```
RDEVICE ADDRESS=(640,32),DEVTYPE=3088
RCTLUNIT ADDRESS=640,CUTYPE=3088,FEATURE=32-DEVICE
```

Network Utility Definition for VM/XA and VM/ESA

The Network Utility must be defined to a VM/Extended Architecture (VM/XA or VM/ESA operating system. This definition is accomplished by updating the real I/O configuration file (HCPRIO) with an entry for the Network Utility in the RDEVICE macro. In the following examples, 640 and 2A0 are base control unit addresses. The address range size, as defined in the UCW or IOCP, is 8 in both examples.

The following example is a VM/XA HCPRIO definition:

```
RDEVICE ADDRESS=(640,8),DEVTYPE=CTCA
```

The following example is a VM/ESA HCPRIO definition:

```
RDEVICE ADDRESS=(2A0,8),DEVTYPE=CTCA
```

Network Utility Definition for MVS/XA and MVS/ESA without HCD

The Network Utility must be defined to an IBM Multiple Virtual Storage/Extended Architecture (MVS/XA) or MVS/ESA operating system. This definition is accomplished by updating the MVS Control Program with an entry for the Network Utility in the IODEVICE macro.

For ESCON channels, an example IODEVICE macro is:

```
IODEVICE UNIT=3172,ADDRESS(540,8)
```

For parallel channels, an example IODEVICE macro is:

```
IODEVICE UNIT=CTC,ADDRESS(640,8)
```

The base control unit addresses are 640 and 540. The address range size, as defined in the UCW or IOCP, is 8 in both examples.

Network Utility Definition for MVS/ESA with HCD

The hardware configuration definition (HCD) component of MVS/ESA SP Version 4.2 and 4.3 with APAR #OY67361 offers an improved method of defining system hardware configuration for Network Utility. Several complex steps required for entering hardware configuration data can be accomplished using an interactive dialog with HCD.

The required configuration data for the Network Utility is:

- When using HCD, with APAR #OY67361, the Network Utility is defined as (UNIT=3172). For example,

  ```
  IODEVICE UNIT=3172,ADDRESS(740,8)
  ```

- Without HCD, the Network Utility is defined for:

  - Parallel channels as a 3088 device (UNIT = 3088 or CTC)

    ```
    IODEVICE        UNIT=CTC,ADDRESS(840,8)
    ```

  - ESCON channels as a serial CTC device (UNIT = SCTC)

    ```
    IODEVICE        UNIT=SCTC,ADDRESS(A40,8)
    ```

Notes:

1. If you are using HCD for MVS Version 4 to define your ESCON host connection, you will need APAR # OY67361 to obtain the UIM support for the device definition (UNIT=3172).

2. When migrating your IOCP definition and operating system definitions to the HCD environment, it is important that all Network Utility device statements be changed to device type (UNIT=3172).

Network Utility Definition for VSE/ESA

The Network Utility must be defined to a VSE/ESA operating system. This definition is accomplished by supplying an ADD statement for each channel unit address at initial program load (IPL) time. Code the device type on the ADD statement as CTCA,EML as shown in the following example:

```
ADD 640,CTCA,EML
```

The base control unit address is 640 in the example. For the number of channel unit addresses added, increment the IOTAB storage macro by this count.

VTAM Definitions

This section gives sample VTAM definitions for an XCA major node, an MPC+ local PU and Transport Resource List (TRL) major node, and an example of defining VTAM for APPN and DLUR support. It also shows an example of a switched major node for a PU in a TN3270 server. This section is not meant to be a complete reference on the subject. For more information on configuring VTAM, refer to the *CS OS/390 Resource Definition Reference*, SC31-8565.

VTAM XCA Major Node Definition

When defining a channel gateway using LSA to VTAM, a definition for an External Communications Adapter (XCA) is required. This definition is the same as that used for an IBM 3172. An example is shown in Figure 18-3 on page 18-7.

```
********************************************************************
RAINETU VBUILD TYPE=XCA      1
**
**
RANETUP  PORT  ADAPNO=0,     2                                    * X
              CUADDR=285,    3                                    * X
              MEDIUM=RING,   4                                    * X
              SAPADDR=4,     5                                    * X
              TIMER=60
**
********************************************************************
RANETUG1 GROUP DIAL=YES,CALL=INOUT,DYNPU=YES
*
RANETUL1 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP1 PU   ISTATUS=ACTIVE
RANETUL2 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP2 PU   ISTATUS=ACTIVE
RANETUL3 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP3 PU   ISTATUS=ACTIVE
RANETUL4 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP4 PU   ISTATUS=ACTIVE
RANETUL5 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP5 PU   ISTATUS=ACTIVE
RANETUL6 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP6 PU   ISTATUS=ACTIVE
RANETUL7 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP7 PU   ISTATUS=ACTIVE
RANETUL8 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP8 PU   ISTATUS=ACTIVE
RANETUL9 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP9 PU   ISTATUS=ACTIVE
```

Figure 18-3. XCA Major Node Definition Sample for LSA Direct Connection

Notes:

1 TYPE must be XCA

2 ADAPNO is the LAN number for the Network Utility interface. This value is assigned to the Network Utility's LSA interface when it is created. The value can be obtained from the Network Utility by listing the configuration of the interface from the talk 6 menus or it can be retrieved by entering the list nets command from the ESCON console in Talk 5. Note that a wrong value for this parameter is the single most common error in LSA configuration.

3 CUADDR specifies the subchannel to be used to communicate with the Network Utility. This value must be within the range of values specified in the IODEVICE statement in the IOCP definition.

4 This specifies the physical LAN topology to which the LSA interface is attached. This corresponds to the value specified for LANtype for the Network Utility interface. Valid values are MEDIUM=RING for token-ring, MEDIUM=CSMACD for Ethernet, and MEDIUM=FDDI for a Fiber Distributed Data Interface (FDDI) network.

5 SAPADDR is the Service Access Point number VTAM wishes to open on the Network Utility. Note that it is the SOURCE SAP, not the DESTINATION SAP. When more than one active XCA major node refers to the same LAN, all the XCA major nodes have to use different SAPs.

LINE Statement: The CALL field can be one of the following:

- IN means only remote devices may establish connections.
- OUT means only VTAM can initiate connections.
- INOUT connections may be initiated at either end.

If VTAM is going to dial out, the Switched Major Node definition must specify a destination with a PATH statement.

An asterisk in the first column indicates a statement has been commented out, and should be ignored. A character in the last column indicates the next line is a continuation of this line.

VTAM Definitions for an MPC+ Connection

An MPC+ connection requires entries in two VTAM control blocks:

- The Local Major Node

- The Transport Resource List (TRL) Major Node

Figure 18-4 shows a sample definition for a local SNA major node for a Network Utility MPC+ connection. This is the local PU that resides in VTAM that supports the channel connection defined in the TRL. The connection type must be APPN and you also need to enable HPR.

```
LOCNETU  VBUILD TYPE=LOCAL
MPCNETUP PU    TRLE=MPCNETU,
               XID=YES,
               CONNTYPE=APPN,
               CPCP=YES,
               HPR=YES
```

Figure 18-4. VTAM Local Major Node Definition

Notes:

1. TYPE must equal LOCAL on the VBUILD statement.

2. TRLE identifies the TRL being used. The name must match the name of an existing TRL.

3. XID indicates whether XIDs will be exchanged. It must be XID=YES.

4. CONNTYPE must be set to CONNTYPE=APPN since APPN is the only protocol that VTAM uses with an MPC+ connection.

5. CPCP specifies that CP-CP connections with APPN can be established over this MPC+ connection. This could be either set to YES or NO, depending upon your APPN topology.

6. HPR specifies that APPN HPR traffic can flow over this MPC+ connection. HPR is normally used by default, but setting this value to YES ensures it. This is important because an MPC+ connection requires RTP (and HPR).

Next, you need a transport resource list for the MPC+ connection from the Network Utility. An example definition is shown in Figure 18-5 on page 18-9.

```
        VBUILD TYPE=TRL
MPCNETU TRLE   LNCTL=MPC,
               MAXBFRU=9,
               READ=280,
               WRITE=281,
               MPCLEVEL=HPDT,
               REPLYTO=3.0
```

Figure 18-5. VTAM Transport Resource List (TRL) Definition

Notes:

1. TYPE must be TRL.

2. MPCNETU is the name that identifies the TRL. It must match what is specified in the TRLE= field in the local major node definition. (See Figure 18-4 on page 18-8.)

3. LNCTL identifies the connection type. It must be LCNTL=MPC.

4. MAXBFRU is the number of 4K pages per read subchannel.

5. READ/WRITE specifies the subchannels in the MPC+ group, and indicates their direction. The subchannel numbers must be in the range of addresses specified in the IODEVICE statement in the IOCP definition. There can be multiple READ and WRITE parameters in the TRLE statement but there must be at least one of each.

   Note: The designations READ and WRITE here are from the HOST perspective. In the Network Utility MPC+ definition, the designations are from the Network Utility perspective. Therefore, subchannels designated as READ on the host MUST be designated as WRITE on the Network Utility, and vice versa.

6. REPLYTO is the reply timeout value in seconds.

VTAM Definitions for APPN

If VTAM is configured for DLUS, then it must be an APPN network node. Configuring VTAM as an APPN network node requires certain parameters to be specified in the VTAM startup parameters. These are shown in Figure 18-6 on page 18-10. Set the CONNTYPE to APPN and the NODETYPE to a Network Node (NN).

```
                    ASYDE=TERM,IOPURGE=5M,
                    CONFIG=I0,
                    CONNTYPE=APPN,
                    CPCP=YES,
                    CSALIMIT=0,
                    DYNADJCP=YES,
                    ENCRYPTN=NO,
                    GWSSCP=YES,
                    HOSTPU=ISTPUS18,
                    HOSTSA=18,
                    HPR=RTP,
                    NETID=USIBMRA,
                    NODETYPE=NN,
                    NOTRACE,TYPE=VTAM,IOINT=0
                    PPOLOG=YES
                    SORDER=APPN,
                    SSCPDYN=YES,
                    SSCPID=18,
                    SSCPNAME=RAI,
                    SSCPORD=PRIORITY,
                    SUPP=NOSUP,
                    TNSTAT,CNSL,
                    VRTG=YES
                    OSITOPO=LLINES,
                    OSIMGMT=YES
                    XNETALS=YES
```

Figure 18-6. VTAM Start-up Parameters

VTAM Static Definition of TN3270 Resources

VTAM definitions are required for the PUs used by the TN3270e Server. You need
a switched major node definition for each PU in the TN3270e server. For example,
each PU in the TN3270e server can support up to 253 LUs. If you need 500 3270
sessions, then you will need 2 PUs in the router and 2 PU definitions in VTAM.

Figure 18-7 shows an example of a VTAM switched major node definition for a
TN3270e server PU that is connected via DLUR and APPN.

```
LOCNETU  VBUILD TYPE=SWNET
MNETUA   PU     ADDR=01,ISTATUS=ACTIVE,VPACING=0,                          *
                DISCNT=NO,PUTYPE=2,SSCPFM=USSSCS,USSTAB=US327X,            *
                IDBLK=077,IDNUM=02216,IRETRY=YES,MAXDATA=521,              *
                MAXOUT=7,MAXPATH=8,PASSLIM=7,PACING=0,ANS=CONTINUE
********************************************************************
PNETUA   PATH   PID=1,DLCADDR=(1,C,INTPU),DLCADDR=(2,X,07702216),         *
                DLURNAME=MNETUA
********************************************************************
JC7LU2   LU     LOCADDR=2
JC7LU3   LU     LOCADDR=3
JC7LU4   LU     LOCADDR=4
```

Figure 18-7. VTAM Definitions for a TN3270E Server PU (DLUR/APPN)

Figure 18-8 on page 18-11 shows an example of a VTAM switched major node definition for a TN3270e server PU that uses a subarea connection to the host.

```
LSAP08T VBUILD TYPE=SWNET
PUPS08T PU ADDR=01,IDBLK=077,IDNUM=12244,MAXOUT=7,PACING=0,VPACING=0,
               DLOGMOD=B22NNE,PUTYPE=ANY,
               SSCPFM=USSSCS,MAXDATA=2000,MODETAB=LMT3270
PT08LU2 LU LOCADDR=02,LOGAPPL=TSO
PT08LU3 LU LOCADDR=03,LOGAPPL=TSO
PT08LU4 LU LOCADDR=04,LOGAPPL=TSO
PT08LU5 LU LOCADDR=05,LOGAPPL=TSO
PT08LU6 LU LOCADDR=06,LOGAPPL=TSO
```

Figure 18-8. VTAM Definitions for a TN3270E Server PU (Subarea)

The following sections provide an overview of the statements in the Switched Major Node Definition.

VBUILD Statement
The TYPE field must be TYPE=SWNET.

PU Statement
This statement defines the type of data flow and the destination. The pertinent parameters are:

- ADDR is merely an identifier.

- MAXDATA is the maximum packet size VTAM will support over this interface. This value will be negotiated down with the Network Utility during the XID exchange.

- IDBLK/IDNUM identify the remote device when VTAM is communicating with PU 2.0 (dependent) devices.

LU Statement
These statements define the logical units (LUs) that can be contacted through this PU. The name on the left of each statement is the name that the host uses to address each LU. The LOCADDR is used by the Network Utility to identify the correct LU in VTAM.

PATH Statement
If VTAM is going to dial out, the Switched Major Node definition must specify a destination with a PATH statement. The path statement will be different depending on whether the TN3270e server attaches via a Subarea or a DLUR/APPN connection.

For a subarea connection, the format is:

`PATH DIALNO=xxyyzzzzzzzzzzzz`

 where:

- xx is a place holder
- yy is the destination SAP number
- zz is the destination MAC address

The example in Figure 18-8 does not have a PATH statement because in this example, the downstream PU will contact VTAM instead of VTAM dialing out to the device.

The example in Figure 18-7 on page 18-10 shows a PATH statement for a TN3270e server PU that is using DLUR to connect to the host. Here, the PATH statement identifies the CP name of the Network Utility (MNETUA) via the DLURNAME parameter. This is needed in order for the LU6.2 conversation between the DLUR and DLUS to be established. Once this session has been established, the SSCP-PU session between VTAM and the TN3270e server PU will be established using the IDBLK/IDNUM value specified by DLCADDR=(2,X,07702216).

VTAM Dynamic Definition of TN3270 Resources

VTAM contains support for several facilities that reduce the amount of user coding required to define its resources such as PUs and LUs. When implemented in Network Utility, TN3270 PUs and LUs appear as switched resources to the VTAM host and require corresponding definitions in VTAM. When large TN3270E environments are being implemented, the definition of these resources could be a very labor intensive task.

VTAM provides a facility that allows switched resources to be defined dynamically. TN3270E can take advantage of this facility to reduce the amount of VTAM definitions the user has to provide. This facility is called *VTAM Dynamic Dial-In* support. This function should not be confused with a similar VTAM function call *Dynamic Definition of Dependent LUs (DDDLU)* which requires corequisite function to be present in the TN3270 server. Network Utility does not currently have this corequisite function.

The details for Dynamic Dial-In Support can be found in the document *VTAM Customization Manual* for the release level of VTAM that is installed on your VTAM host. A brief description of this function and its potential use in a TN3270E environment follows.

General Overview

Dynamic Dial-In Support makes use of a VTAM exit called the Configuration Services Exit (ISTEXCCS), and a set of model PU/LU definitions that the user must define. Each time VTAM receives a connection request from a PU that is not defined to VTAM, the Configuration Services exit is driven and a set of PU and LU definitions are dynamically generated based on the model definitions. These definitions are associated with the PU requesting connection. This set of definitions may be tailored down to the specific PU level, with matching based on information contained in the dial-in PU's XID. This process is repeated each time a connection request is received from a PU which is not defined to VTAM.

Figure 18-9 on page 18-13 contains a VTAM definition for a set of Model definitions that could be used to implement Dynamic Dial-In support. Notice that the definition is created in a VTAM member with a VBUILD TYPE= MODEL. This example contains two PU models and 2 models for LUs. They are the prototypes from which VTAM will generate its dynamic resource definitions. From a practical point of view, if all of the PUs and LUs configured for TN3270E have similar characteristics such as logmode and pacing values, then a single PU and LU definition in the model definition would be sufficient. The VTAM Configuration

Services exit mentioned above could be used to select the appropriate model definition based on XID parameters such as CPNAME and IDNUM/IDBLK. The corresponding values stored in the VTAM datasets CPNDEF and NIDDEF indicate which model should be used. If these datasets are not defined, the exit has an internal algorithm to select model the model and generate LU resource names. This exit routine can be used as is or modified to fit user needs. See "Resource Name Generation" on page 18-14 for a description of the default name generation algorithm and how the user can control what resource names are generated.

VTAM Host

Model Major Node   ISTDSWMN

```
           VBUILD          PUV1     PU IDNUM=11111,,,,
TYPE=MODEL                 LU1_1    LOCADDR=2,,,
PUMOD1  PU                 LU1_2    LOCADDR=3,,,
ADDR=C1,ANS=,...           .
LUMOD1  LU  LOCADDR=2,..   .
PUMOD2  PU  ADDR=C2        LU120  LOCADDR=20
LUMOD2  LU  LOCADDR=3,...
                           PUV2   PU IDNUM=22222,,,
                           LU2_1    LOCADDR=2,,,
                           LU2_2    LOCADDR=3,,,
                           .
                           .
                           LU2_10 LOCADDR=10
```

Network Utility

```
PU1 IDNUM=11111,POOL=20
PU2 IDNUM=22222,POOL=10
```
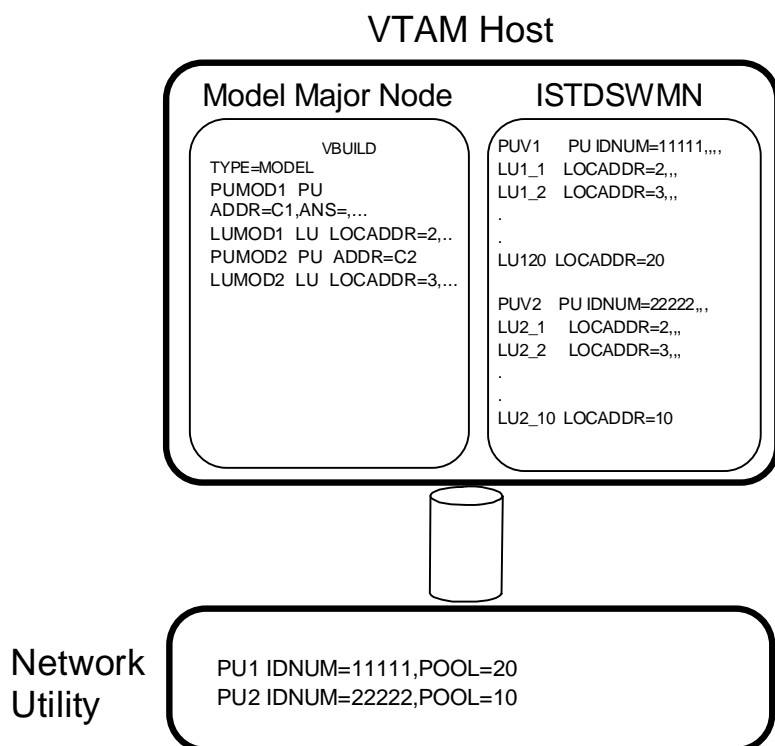
Figure 18-9. VTAM Dynamic Resource Definition

In Figure 18-9, the Network Utility has two PUs defined. PU1 has 20 pooled LU resources defined and PU2 has 10 pooled LU resources defined. In the VTAM host a model major node which contains 2 PU and 2 LUs is defined. These model definitions will contain all of the normal PU and LU parameters such as polling address, logmode tables, MAXDATA values, and PUTYPE that would normally be defined for specific PUs and LUs. As mentioned earlier, if all of the characteristics of the connecting devices are similar a single PU and LU definition in the model would be sufficient.

In this example, VTAM would dynamically generate the definitions shown in ISTDSWMN, a dynamically created major node. ISTDSWMN contains definitions for a PUV1 and 20 LUs (LU1_1 through LU1_20) and for a PUV2 with 10 LUs (LU2_1 through LU2_10). Notice that the PU names in the Network Utility configuration do not have to match the names that VTAM generates. The PUs in the Network Utility are correlated to the VTAM PUs based on matching IDNUM values. Notice also that the only definitions required to implement the dynamic definitions are done on the VTAM host. The Network Utility does not know that VTAM is performing the dynamic definition process.

Dynamic Dial-In Exit Overview

The Configuration Services Installation EXIT (ISTEXCCS) is shipped with VTAM and can be found in the dataset SYS1.SAMPLIB. This exit is described in an appendix of the *VTAM Customization Manual* for the release of VTAM installed on your host.

If the exit is installed in the VTAMLIB dataset at VTAM initialization time, it will be loaded and started during VTAM initialization. VTAM calls the exit whenever it receives a REQCONT RU from a real switched PU, or a REQACTPU from a DLUR node representing a PU. These RUs are generated by the receipt of an XID from the connecting device. They contain XID information such as NETID, CPNAME and IDNUM/IDBLK. The exit uses this information to construct a *build* vector. The build vector contains the model names for PU and LU definitions contained in an active Model major node, and the names to be used for the PU and LU definitions that VTAM is to create. VTAM then builds the definitions for the connecting PU in the dynamic Switched Major Node ISTDSWMN. From this point on, VTAM treats these resources just as if they had been predefined by the user.

Implementing Dynamic Definitions

Resource Name Generation: There are two ways for the user to influence what resource names it will associate with the resources that VTAM dynamically generates. The first permits complete control by allowing the user to supply specific LU and PU names for the resources definitions that VTAM generates. This support is implemented by coding the appropriate information in the VTAM datasets CPNDEF and/or NIDDEF. These datasets provide the information that the Configuration Services exit uses to construct the Build vector that VTAM uses to generate the dynamic resources. While this approach requires more user VTAM definitions, it is much less definition intensive than manually defining the required VTAM definitions. At the other extreme, the user can allow VTAM to generate the resource names based on an internal algorithm. Implementations not requiring unique naming conventions would be candidates for this approach, which requires minimal VTAM definitions.

The name generated for the PU has the format **CNNNNNSS**, where:

C　　　　can be user specified in a name definition table

NNNNN　is the IDNUM extracted from the received XID

SS　　　is the station address (if specified), otherwise it consists of two blank characters

The name generated for the LU is of the form **CNNNNNLL** where:

C　　　　is a user-specified name prefix

NNNNN　is the IDNUM extracted from the received XID

LL　　　is the local address of the LU

The details for both of these approaches to resource name generation can be found in the *VTAM Customization Manual*.

Operational Considerations: The dynamically created definitions in the dynamic Switched Major Node ISTDSWMN are kept as long as there are active LU sessions on the LU. If **DISCNT=YES** is specified on the Model PU definition, all of the dynamic resources associated with the PU will be deleted from ISTDSWMN when

all of the sessions on the PU have ended. If DISCNT=NO is specified, these definitions will not be deleted as long as the PU remains active to VTAM.

Security Considerations: When the Configuration Services Exit dynamically defines resources, VTAM has no predefined IDNUM/IDBLK or CPNAME to validate the identity of connecting devices. If this is considered a security exposure, the exit an be modified to consult a list of acceptable IDNUM/IDBLKs or CPNAMEs and compare these values against those contained in the XID from the connecting device. Recall that the XID from the connecting device is passed to the exit.

Sources for Additional Information: The details for this facility and additional information can be found in the following VTAM library documents:

> *VTAM Customization Manual*

> *VTAM Resource Definition Reference*

> *VTAM Network Implementation Guide*

You should use the manuals corresponding to the release level of he VTAM installed on your VTAM host.

Host IP Definitions

Defining the Network Utility to the host for a TCP/IP connection requires you to make changes to the host TCP/IP profile. This section gives an overview of the relevant statements that need to be changed.

DEVICE Statement

This statement defines the subchannel pair being used by TCP/IP. The format is:

```
DEVICE  Name  LCS  Subchannel
```

where:

- Name identifies the subchannel path being used. It has local significance only, and can be anything.

- Subchannel identifies the even subchannel being used for this connection. This value comes from the IODEVICE statement in the IOCP definition. When specified, that subchannel and the next one are both being used.

A TCP/IP profile must contain one DEVICE statement for each subchannel pair being used.

LINK Statement

This statement identifies which LCS interfaces on the Network Utility are being used on a given subchannel pair. The format is:

```
LINK Name Lantype Lannumber Devicename
```

where:

- Name identifies the LCS interface. It has local significance only, and can be anything.

- Lantype identifies the type of LAN interface the Network Utility   . LCS interface is emulating. The allowable values are:

- – IBMTR for Token-Ring
- – ETHERNET for Ethernet V2
- – 802.3 for Ethernet (IEEE 802.3)
- – ETHERor802.3 for either Ethernet format accepted
- – FDDI for FDDI

- Lannumber identifies which LCS interface on the Network Utility is being used. The lannumber is generated sequentially for each lantype on the Network Utility when the user adds an LCS interface. The lannumber can be found by entering "list nets" from the ESCON console in Talk 5. Note that the lannumber is NOT the net number. Having the wrong Lannumber is the single most common configuration error for an LCS interface.

- Devicename correlates the LCS interface to a subchannel pair. It must match a previously defined DEVICE statement.

There may be multiple LINK statements associated with a single DEVICE statement. There must be an LCS interface on the Network Utility for each LINK statement.

HOME Statement

This statement specifies the IP address(es) of the host TCP/IP stack. The format is:

```
HOME     ipaddress1     link1
         ipaddress2     link2
```

where:

- IpaddressX specifies an IP address on the host.

- LinkX specifies which link is associated with this IP address.

There must be one and only one HOME address for each LINK statement. The HOME address must be in the same IP subnet as the IP address of the LCS interface in the Network Utility, but *they must be different addresses*

GATEWAY Statement

This statement identifies the IP routing information for the host. It is divided into three sections:

- Direct Routes are routes directly connected to the host. The subnet containing the Network Utility LCS interface is a direct route.

- Indirect Routes are routes that are accessible via routers. The subnets of the LANs on the Network Utility are indirect routes, for example.

- Default Route is the route to be used if the host doesn't have a direct or indirect route to an IP address.

Direct Routes
The format for Direct Routes is:

```
network firsthop linkname pktsize submask subvalue
```

where:

- Network is the non-subnetted part of the IP address.

- Firsthop indicates the IP address of the next hop in the IP network. For Direct Routes, this should be an equal sign (=).

- Linkname identifies which link the host should use to get to the addresses on this route. For routes accessible via the Network Utility, this should be the name from the LINK statement associated with the LCS interface on this subnet.

- Pktsize is the maximum frame size to be used on the interface. It should be less than or equal to the packet size defined in the LCS configuration on the Network Utility. A value of DEFAULTSIZE indicates the default packet size will be used.

- Submask specifies the subnet mask used on this link. The subnet mask should correspond to the subnet mask defined for the LCS interface in the IP configuration on the Network Utility. This field may also be set to HOST to identify a point-to-point connection. In this case, the network field should contain the full IP address of the LCS interface.

- Subvalue specifies the subnetted part of the IP address, and together with the network field, should fully specify the IP subnet associated with this LCS interface.

Indirect Routes

The format for Indirect Routes is:

```
network     firsthop     linkname     pktsize     submask     subvalue
```

where:

- Network is the full address of the IP subnet.

- Firsthop indicates the IP address of the next hop in the IP network. For Indirect Routes accessible via the Network Utility, this should be the IP address of the Network Utility LCS interface.

- Linkname identifies which link the host should use to get to the addresses on this route. For routes accessible via the Network Utility, this should be the name from the LINK statement associated with the LCS interface on this subnet.

- Pktsize is the same value as for Direct Routes.

- Submask should either be 0 or blank if the network field contains the full subnet address.

- Subvalue should be left blank if there is no subnet mask specified.

Default Routes

The format for Default Routes is:

```
network firsthop linkname pktsize submask subvalue
```

where:

- Network should say DEFAULTNET.

- Firsthop indicates the IP address of the next hop in the IP network. For Default Routes to the Network Utility, this should be the IP address of the Network Utility LCS interface.

- Linkname identifies which link the host should use to get to the addresses on this route. For routes accessible via the Network Utility, this should be the name from the LINK statement associated with the LCS interface on this subnet.
- Pktsize is the same value as for Direct Routes.
- Submask should either be 0 or blank.
- Subvalue should be left blank.

START Statement

This statement causes the specified subchannels to be started. The format is:

```
START devicename
```

where devicename is the name on the DEVICE statement above.

There must be a START statement for every DEVICE statement if the customer wishes to activate the devices when TCP/IP is started. If the START statement is not here, the devices can be started using the OBEY file. Note that the name here is the one from the DEVICE statement, not the LINK statement. Note also that the Network Utility LCS interface will remain in the DOWN state until the START has been issued from TCP/IP.

Host TCP/IP Definitions for LCS

This section gives you examples of the above statements required if you are defining an LCS connection.

1. DEVICE statement:

```
DEVICE LCS1 LCS 210
```

where LCS1 is the device name being defined, LCS is the type of device, and 210 is the host read (Network Utility write) subchannel used for this definition.

2. LINK statement

```
LINK ETHLCS1 802.3 0 LCS1
```

where ETHLCS1 is the link name, 802.3 is the LAN type to which the LCS interface attaches on the Network Utility, 0 is the LAN number assigned by the Network Utility, and LCS1 is the name of the device (from the device statement above).

Note: Remember that the LAN number is automatically assigned by the Network Utility when you define the LCS interface. You can obtain it by issuing a `list all` command from the ESCON Config> prompt in the talk 6 process on the Network Utility console.

3. HOME command

```
HOME 9.24.106.72 ETHLCS1
```

where 9.24.106.72 is the IP address of this LCS interface and ETHLCS1 is the name of the link.

4. GATEWAY command

```
GATEWAY 9.24.106   9.24.106.1   ETHLCS1   4096  0
```

where 9.24.106 is the IP address for the network, 9.24.106.1 is the IP address of the default router, ETHLCS1 is the link name defined by the LINK statement

above, 4096 is the MTU size, 0 is the subnet mask, and the subnet value has been left blank.

5. Activate the TCP/IP profile

   To activate the device defined in 1 on page 18-18, issue the following command:

   ```
   start lcs1
   ```

Host TCP/IP Definitions for MPC+

The steps for configuring TCP/IP in the host for an MPC+ connection are the same as for an LCS connection. However, the command syntax for the device and link commands is slightly different. For an MPC+ connection, the syntax for the device command is:

```
DEVICE IPTRL1 MPCPTP
```

where IPTRL1 is the name of the TRL that this connection will use and MPCPTP specifies an MPC point-to-point link.

To define the link, the syntax is:

```
LINK LINK1 MPCPTP IPTRL1
```

where LINK1 is the link name and the other two parameters are the same as those used in the device statement.

Appendix

Appendix A. Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Subject to IBM's valid intellectual property, or other legally protectable rights, any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594 USA

Notice to Users of Online Versions of This Book

For online versions of this book, we authorize you to:

- Copy, modify, and print the documentation contained on the media, for use within your enterprise, provided you reproduce the copyright notice, all warning statements, and other required statements on each copy or partial copy.

- Transfer the original unaltered copy of the documentation when you transfer the related IBM product (which may be either machines you own, or programs, if the program's license terms permit a transfer). You must, at the same time, destroy all other copies of the documentation.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization.

THERE ARE NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Your failure to comply with the terms above terminates this authorization. Upon termination, you must destroy your machine-readable documentation.

Electronic Emission Notices

Federal Communications Commission (FCC) Class A Statement

Note: 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 the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used in order to meet FCC emission limits. IBM is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Industry Canada Class A Emission Compliance Statement

This Class A digital apparatus complies with Canadian ICES-003.

Avis de conformité aux normes d'Industrie Canada

Cet appareil numérique de la classe A est conform à la norme NMB-003 du Canada.

Japanese Voluntary Control Council for Interference (VCCI) Statement

This product is a Class A Information Technology Equipment and conforms to the standards set by the Voluntary Control Council for Interference by Technology Equipment (VCCI). In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

　この装置は、情報処理装置等電波障害自主規制協議会（ＶＣＣＩ）の基準に基づくクラスＡ情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

CISPR22 Compliance Statement

This product has been tested and found to comply with the limits for Class A Information Technology Equipment according to CISPR 22/European Standard EN 55022. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Taiwanese Class A Warning Statement

警告使用者:
這是甲類的資訊產品,在
居住的環境中使用時,可
能會造成射頻干擾,在這
種情況下,使用者會被要
求採取某些適當的對策•

EMC Directive 89/336/EEC Statement

This product is in conformity with the protection requirements of EC Council Directive 89/336/EEC on the approximation of the laws of the Member States relating to electromagnetic compatibility. IBM cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the fitting of non-IBM option cards.

Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Geräten (EMVG) vom 30. August 1995 (bzw. der EMC EG Richlinie 89/336)

Dieses Gerät ist berechtigt in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen.

Verantwortlich für die Konformitätserklärung nach Paragraph 5 des EMVG ist die IBM Deutschland Informationssysteme GmbH, 70548 Stuttgart.

Informationen in Hinsicht EMVG Paragraph 3 Abs. (2) 2:

Das Gerät erfüllt die Schutzanforderungen nach EN 50082-1 und EN 55022 Klasse A.

EN 55022 Klasse A Geräte müssen mit folgendem Warnhinweis versehen werden: "Warnung: dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funkstörungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen und dafür aufzukommen."

EN 50082-1 Hinweis: "Wird dieses Gerät in einer industriellen Umgebung betrieben (wie in EN 50082-2 festgelegt), dann kann es dabei eventuell gestört werden. In solch einem Fall ist der Abstand bzw. die Abschirmung zu der industriellen Störquelle zu vergrößern."

Anmerkung: Um die Einhaltung des EMVG sicherzustellen sind die Geräte, wie in den IBM Handbüchern angegeben, zu installieren und zu betreiben.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ESCON	Nways	OS/2
IBM	NetView	Presentation Manger

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

Appendix B. Safety Information

⚠ **Danger:** Before you begin to install this product, read the safety information in *Caution: Safety Information—Read This First*, SD21-0030. This booklet describes safe procedures for cabling and plugging in electrical equipment.

⚠ **Gevaar:** Voordat u begint met de installatie van dit produkt, moet u eerst de veiligheidsinstructies lezen in de brochure *PAS OP! Veiligheidsinstructies—Lees dit eerst,* SD21-0030. Hierin wordt beschreven hoe u electrische apparatuur op een veilige manier moet bekabelen en aansluiten.

⚠ **Danger:** Avant de procéder à l'installation de ce produit, lisez d'abord les consignes de sécurité dans la brochure *ATTENTION: Consignes de sécurité—A lire au préalable,* SD21-0030. Cette brochure décrit les procédures pour câbler et connecter les appareils électriques en toute sécurité.

⚠ **Perigo:** Antes de começar a instalar este produto, leia as informações de segurança contidas em *Cuidado: Informações Sobre Segurança—Leia Isto Primeiro,* SD21-0030. Esse folheto descreve procedimentos de segurança para a instalação de cabos e conexões em equipamentos elétricos.

⚠

危險：安裝本產品之前，請先閱讀
"Caution: Safety Information--Read
This First" SD21-0030　　手冊中所提
供的安全注意事項。　這本手冊將會說明
使用電器設備的纜線及電源的安全程序。

⚠

Opasnost: Prije nego sto pŏcnete sa instalacijom produkta,
pročitajte naputak o pravilima o sigurnom rukovanju u
Upozorenje: Pravila o sigurnom rukovanju ‑ Prvo pročitaj ovo,
SD21-0030. Ovaj privitak opisuje sigurnosne postupke za
priključrivanje kabela i priključivanje na električno napajanje.

⚠

Upozornění: než zahájíte instalaci tohoto produktu, přečtěte si
nejprve bezpečnostní informace v pokynech „Bezpečnostní
informace" č. 21-0030. Tato brožurka popisuje bezpečnostní
opatření pro kabeláž a zapojení elektrického zařízení.

⚠ **Fare!** Før du installerer dette produkt, skal du læse sikkerhedsforskrifterne i *NB: Sikkerhedsforskrifter—Læs dette først* SD21-0030. Vejledningen beskriver den fremgangsmåde, du skal bruge ved tilslutning af kabler og udstyr.

⚠ **Gevaar** Voordat u begint met het installeren van dit produkt, dient u eerst de veiligheidsrichtlijnen te lezen die zijn vermeld in de publikatie *Caution: Safety Information - Read This First*, SD21-0030. In dit boekje vindt u veilige procedures voor het aansluiten van elektrische appratuur.

⚠ **VAARA:** Ennen kuin aloitat tämän tuotteen asennuksen, lue julkaisussa *Varoitus: Turvaohjeet—Lue tämä ensin*, SD21-0030, olevat turvaohjeet. Tässä kirjasessa on ohjeet siitä, miten sähkölaitteet kaapeloidaan ja kytketään turvallisesti.

⚠ **Danger :** Avant d'installer le présent produit, consultez le livret *Attention : Informations pour la sécurité — Lisez-moi d'abord* SD21-0030, qui décrit les procédures à respecter pour effectuer les opérations de câblage et brancher les équipements électriques en toute sécurité.

⚠ **Vorsicht:** Bevor mit der Installation des Produktes begonnen wird, die Sicherheitshinweise in *Achtung: Sicherheitsinformationen—Bitte zuerst lesen,* IBM Form SD21-0030. Diese Veröffentlichung beschreibt die Sicherheitsvorkehrungen für das Verkabeln und Anschließen elektrischer Geräte.

⚠ **Vigyázat:** Mielôtt megkezdi a berendezés üzembe helyezését, olvassa el a *Caution: Safety Information— Read This First,* SD21-0030 könyvecskében leírt biztonsági információkat. Ez a könyv leírja, milyen biztonsági intézkedéseket kell megtenni az elektromos berendezés huzalozásakor illetve csatlakoztatásakor.

⚠ **Pericolo:** prima di iniziare l'installazione di questo prodotto, leggere le informazioni relative alla sicurezza riportate nell'opuscolo *Attenzione: Informazioni di sicurezza — Prime informazioni da leggere* in cui sono descritte le procedure per il cablaggio ed il collegamento di apparecchiature elettriche.

⚠
危険：　導入作業を開始する前に、安全に関する
小冊子SD21-0030　の「最初にお読みください」
(Read This First)の項をお読みください。
この小冊子は、電気機器の安全な配線と接続の
手順について説明しています。

⚠
위험: 이 제품을 설치하기 전에 반드시
"주의: 안전 정보-시작하기 전에"
(SD21-0030)　에 있는 안전 정보를
읽으십시오.

⚠ **Fare:** Før du begynner å installere dette produktet, må du lese sikkerhetsinformasjonen i *Advarsel: Sikkerhetsinformasjon — Les dette først*, SD21-0030 som beskriver sikkerhetsrutinene for kabling og tilkobling av elektrisk utstyr.

⚠️

Uwaga:
Przed rozpoczęciem instalacji produktu należy zapoznać się z instrukcją:
"Caution: Safety Information - Read This First", SD21-0030.
Zawiera ona warunki bezpieczeństwa przy podłączaniu do sieci elektrycznej
 i eksploatacji.

⚠️ **Perigo:** Antes de iniciar a instalação deste produto, leia as informações de
segurança *Cuidado: Informações de Segurança — Leia Primeiro*, SD21-0030. Este
documento descreve como efectuar, de um modo seguro, as ligações eléctricas
dos equipamentos.

⚠️

ОСТОРОЖНО: Прежде чем инсталлировать этот
продукт, прочтите Инструкцию по технике безо-
пасности в документе "Внимание: Инструкция по
технике безопасности -- Прочесть в первую очередь",
SD21-0030. В этой брошюре описаны безопас-
ные способы каблирования и подключения элект-
рического оборудования.

⚠️

Nebezpečenstvo: Pred inštaláciou výrobku si prečítajte
bezpečnosté predpisy v
Výstraha: Bezpeč osté predpisy - Prečítaj ako prvé,
SD21 0030. V tejto brožúrke sú opísané bezpečnosté
postupy pre pripojenie elektrických zariadení.

⚠️

Pozor: Preden zaènete z instalacijo tega produkta
preberite poglavje: 'Opozorilo: Informacije
o varnem rokovanju-preberi pred uporabo,"
SD21-0030. To poglavje opisuje pravilne
postopke za kabliranje,

⚠️ **Peligro:** Antes de empezar a instalar este producto, lea la información de
seguridad en *Atención: Información de Seguridad — Lea Esto Primero,* SD21-0030.
Este documento describe los procedimientos de seguridad para cablear y enchufar
equipos eléctricos.

⚠️ **Varning — livsfara:** Innan du börjar installera den här produkten bör du läsa
säkerhetsinformationen i dokumentet *Varning: Säkerhetsföreskrifter— Läs detta
först,* SD21-0030. Där beskrivs hur du på ett säkert sätt ansluter elektrisk
utrustning.

⚠️

危險：

開始安裝此產品之前，請先閱讀安全資訊。

注意：

請先閱讀 - 安全資訊 SD21-0030

此冊子說明插接電器設備之電纜線的安全程序。

Index

Special Characters

Numerics

A

B

C

IBM ®

Part Number: 30L6933

30L6933