Using NETBuilder® Family Software
Chapter 21 through Appendix Q

Software Version 9.3

p/n 09-0862-002
Published May 1997

3Com Corporation
5400 Bayfront Plaza
Santa Clara, California
95052-8145

3Com Corporation reserves the right to revise this documentation and to make changes in content from time to time without obligation on the part of 3Com Corporation to provide notification of such revision or change.

3Com Corporation provides this documentation without warranty of any kind, either implied or expressed, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. 3Com may make improvements or changes in the product(s) and/or the program(s) described in this documentation at any time.

UNITED STATES GOVERNMENT LEGENDS:
If you are a United States government agency, then this documentation and the software described herein are provided to you subject to the following restricted rights:

For units of the Department of Defense:
Restricted Rights Legend: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) for Restricted Rights in Technical Data and Computer Software Clause at 48 C.F.R. 52.227-7013. 3Com Corporation, 5400 Bayfront Plaza, Santa Clara, California 95052-8145.

For civilian agencies:
Restricted Rights Legend: Use, reproduction, or disclosure is subject to restrictions set forth in subparagraph (a) through (d) of the Commercial Computer Software - Restricted Rights Clause at 48 C.F.R. 52.227-19 and the limitations set forth in 3Com Corporation's standard commercial agreement for the software. Unpublished rights reserved under the copyright laws of the United States.

If there is any software on removable media described in this documentation, it is furnished under a license agreement included with the product as a separate document, in the hard copy documentation, or on the removable media in a directory file named LICENSE.TXT. If you are unable to locate a copy, please contact 3Com and a copy will be provided to you.

Unless otherwise indicated, 3Com registered trademarks are registered in the United States and may or may not be registered in other countries.

3Com, Boundary Routing, NETBuilder, NETBuilder II, and SuperStack are registered trademarks of 3Com Corporation. 3TECH and OfficeConnect are trademarks of 3Com Corporation. 3ComFacts is a service mark of 3Com Corporation.

CompuServe is a registered trademark of CompuServe, Inc. IBM, AS/400, LAN Net Manager, OS/2, PS/2, Systems Network Architecture (SNA), and VTAM are registered trademarks of International Business Machines Corporation. Advanced Peer-to-Peer Networking and APPN are trademarks of International Business Machines Corporation. AppleTalk, Macintosh, and LaserWriter are registered trademarks of Apple Computer, Incorporated. XNS is a trademark of Xerox Corporation. VAX, DEC, and DECnet are registered trademarks of Digital Equipment Corporation. TeleVideo is a registered trademark of TeleVideo Corporation. NetWare and Novell are registered trademarks of Novell, Inc. Banyan and VINES are registered trademarks of Banyan Systems. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Ltd. Telenet is a trademark of Telenet Communications Corporation. SPARCsystem is a trademark of SPARC International, Inc. licensed exclusively to Sun Microsystems, Inc. Sun and Solaris are registered trademarks of Sun Microsystems, Inc. SunOS is a trademark of Sun Microsystems, Inc. Link level compression uses Stac LZS compression software, copyrighted by Stac Electronics, (© Stac Electronics, 1991-1995) and protected by one or more patents, including U.S. patent 5,126,739. Stac and LZS compression are registered trademarks of Stac Electronics. Computer Library is a trademark of Ziff Communications. HP is a registered trademark of Hewlett-Packard Company. Hayes and Ultra are registered trademarks of Hayes Microcomputer Products, Inc. Motorola is a registered trademark of Motorola Corporation. Cisco is a registered trademark of Cisco Systems. Wellfleet is a registered trademark of Wellfleet Communications, Inc. NCS is a registered trademark of National Computer Systems.

Other brand and product names may be registered trademarks or trademarks of their respective holders.

Guide written by Lianne Card, Mike Figone, Andrew Mann, Patty Nunn, Francisco Razo, Patrick Sullivan, and Carol Tatick. Edited by Amy Guzules and Pat Vaughn. Technical illustration by Debra Knodel. Production by Debra Knodel and Ramona Boersma.

Printed on recycled paper.

CONTENTS

26 CONFIGURING FRAME RELAY ACCESS DEVICE SUPPORT FOR SNA

27 CONFIGURING TUNNELS TO CONNECT PEER SNA NETWORKS

34 WIDE AREA NETWORKING USING PPP AND PLG

35 WIDE AREA NETWORKING USING ISDN

36 CONFIGURING THE NETBUILDER II TO USE A WAN EXTENDER

37 CONFIGURING PORT BANDWIDTH MANAGEMENT

38 CONFIGURING PROTOCOL RESERVATION

42 WIDE AREA NETWORKING USING FRAME RELAY

43 Configuring Wide Area Networking Using the ATM DXI

44 Configuring Wide Area Networking Using SMDS

45 CONFIGURING WIDE AREA NETWORKING USING X.25

51 CONFIGURING LOCAL ACCESS CONTROL

52 MANAGING SESSIONS FOR INCOMING EXTENDED CALLS

53 NETWORK MANAGEMENT

A SWAPPING NETBUILDER II HARDWARE MODULES

B DIAL-UP PROGRESS AND ERROR MESSAGES

C LOOPBACK TESTING

D INTERNET ADDRESSING

H STATISTICS DISPLAYS

I STATIC TABLES

J AUDIT TRAIL MESSAGES

K REGULAR EXPRESSIONS

L X.3 PARAMETERS AND PAD PROFILES

M WIDE AREA NETWORK SETUP INFORMATION

N APPN CONFIGURATION EXAMPLES

O IBM TRACE FACILITY

21 CONFIGURING THE LLC2 DATA LINK INTERFACE

This chapter describes the steps for configuring the Logical Link Control, type 2 (LLC2) data link interface. You may need to configure the LLC2 data link interface if you are configuring source route bridging, Advanced Peer-to-Peer Networking (APPN) routing, data link switching (DLSw), or Synchronous Data Link Control (SDLC).

Configuring LLC2 Data Link Interface

Logical Link Control, type 2 (LLC2) is a connection-oriented version of the LLC data-link layer protocol used to connect end devices. The LLC2 data link interface can be configured on the bridge/router. These parameters determine the session interaction between the LLC2 end systems and the bridge/router. The default settings should be sufficient for most network configurations.

For more information on LLC2 p-bits (poll bits) and I-frames (information frames) configured in this procedure, refer to the *IBM Token-Ring Network Architecture Reference* document.

To configure the LLC2 data link interface, follow these steps:

1 Configure the length of time the bridge/router waits for a response of an LLC2 p-bit command, or acknowledgment of an LLC2 I-frame using:

```
SETDefault !<port> -LLC2 TImerReply = <milliseconds>(5000–60000)
```

The default is 3000 milliseconds.

2 Configure the length of time that the bridge/router will wait before acknowledging the received I-frame using:

```
SETDefault !<port> -LLC2 TImerAck = <milliseconds>(0–500)
```

The default is zero.

3 Configure the time period that the bridge/router expects to receive a frame from the other end using:

```
SETDefault !<port> -LLC2 TImerInact = <milliseconds>(3000–180000)
```

The default is 60,000 milliseconds.

The bridge/router transmits a poll and activates the Reply Timer (configured in step 1) after the specified expiration time.

The TImerInact value should be at least five times the value entered for the TImerReply parameter.

4 Define the retry count, or the maximum number of times to retransmit after the reply timer expires using:

```
SETDefault !<port -LLC2 RetryCount = <retrys>(1–255)
```

The default is 7.

5 Configure the maximum frame size in bytes of the information field using:

```
SETDefault !<port> -LLC2 MaxFrame = <size>(128–4399)
```

The default is 1500 bytes.

6 Configure the receive window size for I-frames using:

```
SETDefault !<port> -LLC2 ReceiveWindow = <size>(1–128)
```

The default is 7.

7 Configure the transmit window size for I-frames using:

```
SETDefault !<port> -LLC2 TransmitWindow = <size>(1–128)
```

The default is 7.

8 Prepare for the number of LLC2 sessions you plan to have using:

```
SETDefault -SYS CONNectionUsage = [High | Medium | Low]
```

This command sets up the number of LLC2 and X.25 connection service sessions allowed at one time. The default for this parameter is “High” for systems using the Dual Processor Engine (DPE) and “Low” for all other systems. If the CONNectionUsage is set to Low, the setting may not be enough depending on how many LLC2 sessions you plan to have.

After setting this parameter, you must reboot the system for it to take effect.

You may need to change the setting to medium for some situations. For example, if you have a few LLC2 sessions running on the low setting, and you cannot get additional LLC2 sessions to connect, you may want to change the setting to medium. In addition, if you are accepting many incoming LLC2 tunnel sessions to a bridge/router serving an systems network architecture (SNA) host, you may want to change the setting to medium.

The number of LLC2 sessions possible with each CONNectionUsage setting depends on several factors, including:

- The hardware platform and the amount of available memory
- The number of X.25 connection service sessions being run

For more information on the CONNectionUsage parameter, refer to Chapter 58 in *Reference for NETBuilder Family Software*.

Displaying LLC2 Information

You can display information about specific LLC2 sessions or a log of LLC2 activity.

To display information regarding LLC2 data link interface sessions, enter:

```
SHow -LLC2 SESSions
```

The display is similar to the following:

```
---------------------------LLC2 Sessions---------------------------
.........LLC2 Active Source Mac Address:%02608C3C36AC.............
Source:%02608C3C36AC Sap:04  Dest:%02608C1A0CE7 Sap:04
Port:!2-ACTIVE RIF: 06F0 (SRF LF=0x38:92&3:258
```

In this display, “Source” refers to the media access control (MAC) address of the bridge/router where the LLC2 connection originated, and “Destination” refers to the MAC address of the bridge/router where the LLC2 connection is intended to go. For tunneling, the source address is the peer MAC address, and the destination address is the local MAC address that is configured with the TUNnelMAcadd parameter.

The source or destination depends on which tunnel peer bridge/router you are using. For example, if you entered the SHow -LLC2 SESSions command on the destination bridge/router shown in the preceding display, the MAC addresses would be reversed, as shown in the following display:

```
--------------------------LLC2 Sessions--------------------------
.........LLC2 Active Source Mac Address:%02608C1A0CE7..............
Source:%02608C1A0CE7 Sap:04  Dest:%02608C3C36AC Sap:04
Port:!2-ACTIVE RIF: Transparent Frame
```

For more information on the parameters in the LLC2 Service, refer to Chapter 34 in *Reference for NETBuilder Family Software*.

You can display a log of LLC2 activity by entering:

```
SHow -LLC2 Llc2LOG
```

The log displays a history of the most recent 256 log entries including the following actions:

- Session activation or deactivation
- Session failure

Configuring LLC2 with Other Services

IBM-related services such as DLSw and APPN are affected by parameter settings in the BRidge, SR, and LLC2 Services. NETBuilder token ring ports that send or receive LLC2 or NetBIOS packets must be configured properly to avoid token ring frame copy errors and to allow connectivity. Table 21-1 shows the required settings in source route (SR), source route transparent (SRT), and transparent bridging environments for each of the IBM-related services. 3Com recommends configuring token ring ports for source route only mode if possible.

In Table 21-1, tunneling refers to the 3Com proprietary method of LLC2 tunneling, DLSw refers to data link switching, and LNM refers to LAN Net Manager. The settings are shown in abbreviated form. 3Com-recommended configurations are shown in bold.

Table 21-1 IBM-Related Settings for Token Ring Ports

Services	**Port Config-uration**	**Source Route Bridging (-SR SRB)**	**Transparent Bridging (-BR TB)**	**Bridging (-BR CONT)**	**Route Discovery (-SR RD)**	**LLC2 CONTrol (-LLC2 CONT)**	**Frame Copy Errors**
Bridging only	**SR**	**SRB**	**NTB**	**BR**	**NoLLC2**	**Disable**	**None**
Bridging only	SRT	SRB	TB	BR	NoLLC2	Disable	*
Bridging only	T	NSRB	TB	BR	NoLLC2	Disable	*
LNM	**SR**	**SRB**	**NTB**	**BR**	**LLC2**	**Enable**	**None**
DLSw/ Tunneling	**SR**	**SRB**	**NTB**	**NBR | BR**	**LLC2**	**Enable**	**None**
DLSw/ Tunneling	SRT	SRB	TB	BR	LLC2	Enable	* †
DLSw/ Tunneling	T	NSRB	TB	BR	NoLLC2	Enable	* †
APPN	**SR**	**SRB**	**NTB**	**NBR | BR**	**LLC2**	**Disable**	**None**

(continued)

Table 21-1 IBM-Related Settings for Token Ring Ports

Services	Port Config-uration	Source Route Bridging (-SR SRB)	Transparent Bridging (-BR TB)	Bridging (-BR CONT)	Route Discovery (-SR RD)	LLC2 CONTrol (-LLC2 CONT)	Frame Copy Errors
APPN	SRT	SRB	TB	NBR | BR	LLC2	Disable	*
APPN	T	NSRB	TB	NBR | BR	LLC2	Disable	*
Default Setting	SRT	SRB	TB	NBR	NoLLC2	Disable	None

* In this configuration, end systems may generate a small number of token-ring MAC frame copy error report packets when the NETBuilder bridge/router is initializing or when it ages out a MAC address from its bridge table.

† In this configuration it is important for global bridging to be enabled, otherwise the token ring hardware does not filter transparent packets. This can generate many frame copy error reports and adversely effect performance. To prevent forwarding of bridge packets in this configuration, enter the following command: SETDefault -BRidge CONTrol = NoForward. The NoForward parameter allows DLSw and LLC2 tunneling to send and receive LLC2 SNA and NETBios packets, but prevents other packets from bridging.

The row in Table 21-1 labeled DLSw/Tunneling with port configuration SR represents DLSw or 3Com tunneling in a source-route-only port configuration. The entries in this row expand to the following NETBuilder software configuration syntax:

```
SETDefault -BRidge CONTrol = Bridge | NoBridge
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
SETDefault !<port> -BRidge TransparentBridge = NoTransparentBridge
SETDefault !<port> -SR RingNumber = <number> (1–4095)
SETDefault !<port> -SR RouteDiscovery = LLC2
SETDefault !<port> -LLC2 CONTrol = Enable
```

In this configuration, global bridging is enabled or disabled on one or more token ring ports. Transparent bridging is disabled, source routing and route discovery are configured, and LLC2 is enabled.

22 Configuring Synchronous Data Link Control Connectivity

This chapter describes how to provide Synchronous Data Link Control (SDLC) connectivity over local and wide area networks, how the SDLC works on the router, and gives guidelines for operating and managing your SDLC configuration.

For conceptual information, refer to "How SDLC Conversion Works" on page 22-9. For information about the parameters in the SDLC Service, refer to Chapter 51 in Reference for NETBuilder Family Software.

On the NETBuilder II system, SDLC is supported only on the HSS 3-Port modules.

Connection Methods

This section describes various SDLC connections. For configuration procedures, refer to "Configuring the Router for SDLC" on page 22-2.

Figure 22-1 shows an SDLC point-to-point configuration where remote PU2 devices use SDLC to connect to an SDLC- or token ring-attached host front end processor (FEP) through the WAN. In this configuration, the SNA and SDLC data is passed through the bridge/router using data link switching (DLSw).

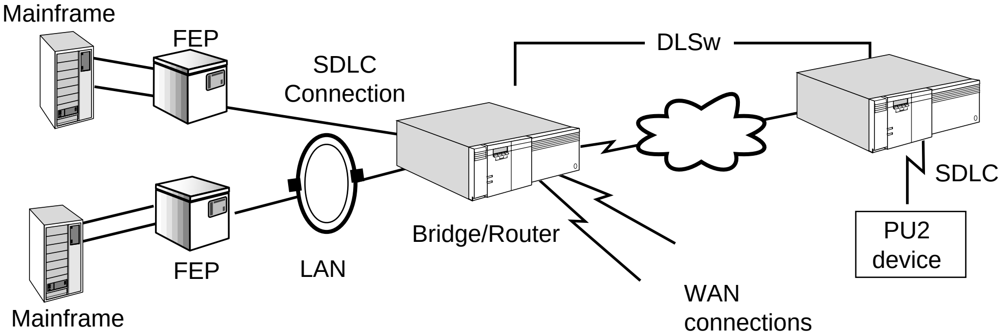


Figure 22-1 SDLC Point-to-Point Configuration

A multipoint configuration may consist of several remote SDLC devices using SDLC connections to a 3Com bridge/router to reduce the number of independent (SDLC and other) links required by the site. In this configuration, the SDLC data is passed through the bridge/router using DLSw. As shown in Figure 22-2, a remote site may be configured as an SDLC primary node talking to 3x74 cluster controllers and other SDLC secondary devices (486x).

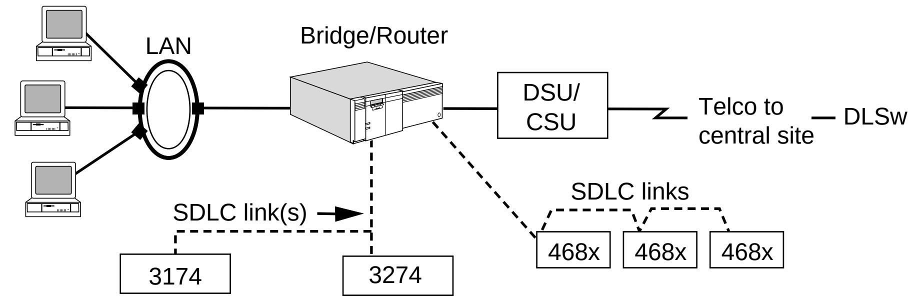


Figure 22-2 SDLC Multipoint Configuration

The SDLC connectivity of the NETBuilder II bridge/router also allows an SDLC-attached device to communicate with a local LAN-attached device or with a front-end processor (FEP) through Frame Relay (see Figure 22-3).

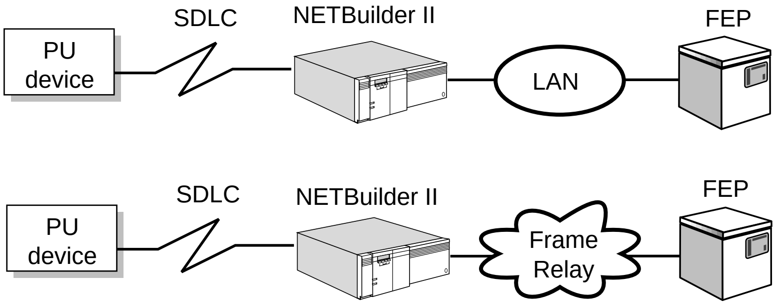


Figure 22-3 SDLC Connectivity through a LAN and Frame Relay

Configuring the Router for SDLC

This section describes how to configure the bridge/router for SDLC. After you complete the procedures in this section, proceed to "Configuring the CU Devices on the Link" on page 22-5.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the non-SDLC ports and paths of your bridge/router according to Chapter 1. For the token ring bridge refer to Chapter 5.
- Set up the ports for SDLC as described later in this guide. Ports being used for SDLC must have a one-to-one port-to-path mapping.
- Set up the LLC2 data link interface as described in Chapter 21.

SDLC is affected by parameter settings in other services. For more information, refer to "Configuring LLC2 with Other Services" on page 21-3.

- Configure IP and obtain the IP addresses of the local bridge/router and DLSw peers.
- Obtain the SNA device address and the Virtual Telecommunications Access Method (VTAM) address so you can match the addresses of the devices you are configuring.
- Allocate a media access control (MAC) and Service Access Point (SAP) configuration to represent the attached SDLC device.

Procedure To configure SDLC, perform the procedures in the following section on bridge/router A (see Figure 22-4), which has the attached SDLC devices.

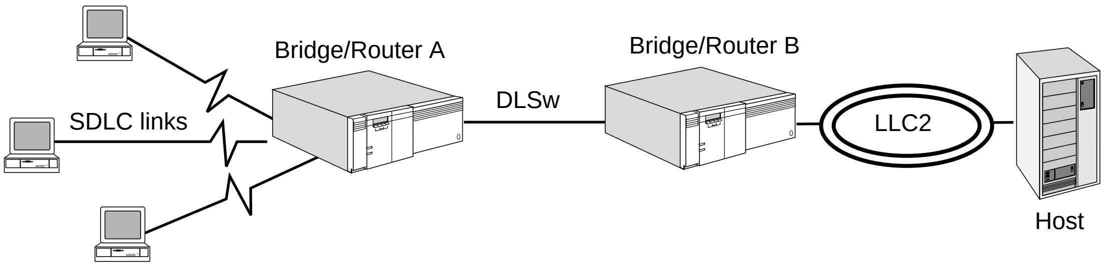


Figure 22-4 Configuring the Router for SDLC

Configuring the SDLC Port and Path Attributes

To configure the port attributes for SDLC, follow these steps:

1 For the port running SDLC, set the OWNer parameter using:

```
SETDefault !<port> -PORT OWNer = SDLC
```

2 Set the communication mode (using the PATH Service DUplex parameter) and the transmission encoding method (using the PATH Service ENCoding parameter) for the path assigned to the SDLC port.

For example, if the attached device requires half-duplex communication and nonreturn to zero (NRZ) encoding, use:

```
SETDefault !<path> -PATH DUplex = Half
SETDefault !<path> -PATH ENCoding = NRZ
SETDefault !<path> -PATH TxIdle = Mark
```

Make sure the parameter settings match the configuration of the device and that you configure the TxIdle parameter as shown. The Mark setting allows half-duplex operation to occur by setting up the system to receive the second half of the transmission without aborting.

3 Make sure the LineType and clocking parameters in the PATH Service are set correctly.

For a back-to-back or a null modem connection you must use external clocking. The NETBuilder II bridge/router does not provide an internal clock source. The LineType must be Leased and clocking must be External. Set the LineType and clocking parameters using:

```
SETDefault !<path> -PATH LineType = Leased
SETDefault !<path> -PATH Clock = External
```

Changes to these parameters do not take effect until you enable the ports and paths.

4 Disable Link Access Procedure, Balanced (LAPB) on the selected path using:

```
SETDefault !<path> -LAPB CONTrol = Disable
```

Because of the SDLC configuration, the port-to-path correlation must be mapped on a one-to-one basis. LAPB cannot be enabled when the port owner is SDLC.

Configuring LLC2 and Bridging Characteristics

If SDLC devices attached to the bridge/router need to communicate with LAN (LLC2) devices attached to the same bridge/router, the bridge/router must be set up to support LLC2 connections. You also must set up DLSw to perform internal switching. If this has not already been done, refer to Chapter 24 and Chapter 27.

If you want your NETBuilder II bridge/router to connect the SDLC devices to a LAN, the LLC2 ports must be configured as described for DLSw in Chapter 24.

Configuring the SDLC Protocol Characteristics

To configure SDLC for communication with the connected devices, follow these steps:

1 Define the control units (CUs) attached to the port.

Assign a name to the CU using this port using:

```
ADD !<port> -SDLC PortCU <CU name>
```

Assigning the CU to the port sets up the SDLC configuration of the port, which allows you to view and modify the port parameter settings. The CU name you assign has only local significance. CU names must be unique and can be no longer than 8 alphanumeric characters. A name longer than 8 characters is rejected and a warning message appears. However, maintaining name consistency between the NETBuilder II bridge/router and network control point (NCP) configurations may simplify configuration management. CU names must be unique within the bridge/router.

The SDLC parameters for a port are inaccessible. They cannot be viewed or modified until at least one CU is assigned to the port with the Port CU parameter.

2 Configure whether the port will act in a primary or secondary role in the connection.

In SDLC, a primary station controls the operation of other secondary stations. The role of the port applies to the port and all of the CUs configured on the port. The role must be set according to the CUs attached to the port, if the role of the CU is secondary, set the port as primary.

For example, to attach a CU device that is secondary, set the port as primary on the bridge/router using:

```
SETDefault !<port> -SDLC PROle = Primary
```

All SDLC ports that are attached to a primary (a host) should be set as secondary. All SDLC ports that are attached to a secondary CU (a 3174 downstream physical unit) should be set as primary.

3 Display the current parameter settings using:

```
SHow !<port> -SDLC PCONFig
```

4 Define whether the port operates in half- or full-duplex mode with the connected device.

To configure a port on the bridge/router for half-duplex operation, use:

```
SETDefault !<port> -SDLC PDatmode = Half
```

The port can be set for half-duplex (two-way alternating) or full-duplex (two-way simultaneous) communication to match the configuration of the SDLC devices on this port.

5 Set the maximum amount of data contained in a single frame (basic transmission unit (BTU) size) using:

```
SETDefault !<port> -SDLC PMaxData = 521
```

The value of this parameter should match the host.

6 Set the frame numbering method used by the CUs attached to this port using:

```
SETDefault !<port> -SDLC PMODulo = 128
```

The setting of this parameter must match the CU on this port.

Configuring the SDLC Protocol Timing Parameters

To configure the timing of the port, which affects how the port waits and responds to communication with the CUs attached to this port, follow these steps:

1 Set the number of times the bridge/router attempts to complete a protocol exchange with an SDLC connected device before considering that device as having failed using:

```
SETDefault !<port> -SDLC PT1Retry = 3
```

2 Set the no-response time-out waiting period for the port using:

```
SETDefault !<port> -SDLC PT1Timer = 400
```

This parameter is used on primary ports only. If the CU does not send a response to a poll or a message from the SDLC port before this timer expires, the transmission is retried until the retry count set in the previous step runs out. At this point, the bridge/router will terminate (disconnect) the SDLC connection and attempt to contact the CU again for a new connection.

3 Set the delay between attempts to connect the network data link (LLC2) partner for CUs whose mode is set to originate using:

```
SETDefault !<port> -SDLC PRetryTimer = 20
```

After you have completed this procedure, proceed to the next section to configure the link stations (CUs) attached to the bridge/router.

Configuring the CU Devices on the Link

This section describes how to configure the SDLC connection for the CU devices the bridge/router has configured on each SDLC port.

Prerequisites

Before beginning this procedure, complete the following steps:

- Configure the SDLC port parameters as described in the previous procedure.
- Obtain the MAC/SAP address pair for the CU. This includes the local MAC/SAP values used to represent this SDLC device in the LLC2 network environment and the remote MAC/SAP values, if the CUMOde parameter is set to Originate, which indicate the CUs partner (LLC2) device.

Procedure

To configure the link for the CU, follow these steps:

1 Define the type of CU you are configuring by entering:

```
SETDefault !LS22 -SDLC CUType = T1
```

This command specifies the CU named LS22 which is a type PU1 or T1 device.

2 Set the CU device identification if required using:

```
SETDefault !<CU name> -SDLC CUXId = 0179097C
SETDefault !<CU name> -SDLC CUXidDefined = Yes
```

This step is optional and depends upon the configuration of the attached CU and the requirements of the network partner device.

The CUXId is only required if the PRole parameter is set to primary and the PU type is set to type 2.0 or type 1 and the attached (secondary) CU will not respond to an exchange identification (XID) poll. The CUXidDefined parameter must also be set to enable use of the defined CUXId value.

Refer to Chapter 51 in *Reference for NETBuilder Family Software* for further information about the CUXId parameter.

3 Configure the poll address of the secondary CU using:

```
SETDefault !<CU name> -SDLC CUAddr = C2
```

If the bridge/router is configured as primary, the CU address must match the PU. If the bridge/router is set as secondary, the CU address must match the host configuration.

4 Configure the local MAC address for the CU using:

```
SETDefault !<CU name> -SDLC CULocalMac = %50004080C940
```

This value is the MAC address used within the LLC2/DLSw environment to communicate with the CU. LLC2 frames intended for this CU must use this value as the destination MAC address. When the NETBuilder II bridge/router sends LLC2/DLSw frames on behalf of this CU, this value is used as the source MAC address. The MAC addresses in this parameter are in noncanonical format.

Two CUs may not use the same CULocalMac and CULocalSap combination.

5 Configure the local SAP used by the CU using:

```
SETDefault !<CU name> -SDLC CULocalSap = 08
```

This value is the LLC2 SAP used for this CU in the LLC2/DLSw environment. LLC2 frames intended for this CU must use this value as the destination SAP. When the NETBuilder II bridge/router sends LLC2/DLSw frames on behalf of this CU, this value is used as the source SAP.

6 Set the maximum number of frames that may be transmitted before waiting for a response using:

```
SETDefault !<port> -SDLC CUMaxOut = 4
```

7 Set up the operating mode for the CU by entering:

```
SETDefault !<CU name> -SDLC CUMOde = Originate
```

The mode determines whether the bridge/router initiates sessions in the LLC2/DLSw environment on behalf of this CU, or whether the bridge/router simply responds to sessions initiated by other stations. The exact sequence of connection events also depends on the PRole parameter.

When CUMOde is set to Originate, the bridge/router initiates an LLC2 connection as soon as it has contacted the SDLC station; that is, when it has received an XID or set normal response mode (SNRM) (if PRole is secondary) or a response to an XID or SNRM (if PRole is Primary).

When CUMOde is set to Answer, the bridge/router will not initiate LLC2 sessions, but responds to LLC2 connection attempts by trying to establish contact with the SDLC station, sending XID or SNRM if PRole is Primary, and responding to XID or SNRM if PRole is secondary.

8 Configure the remote MAC address for the CU using:

```
SETDefault !<CU name> -SDLC CURemoteMac = %60003070C940
```

When CUMOde is Originate, the bridge/router uses this value as the destination MAC address when initiating an LLC2 connection on behalf of this CU.

9 Configure the remote SAP used by the CU using:

```
SETDefault !<CU name> -SDLC CURemoteSap = 08
```

When CUMOde is Originate, the bridge/router uses this value as the destination LSAP when initiating an LLC2 connection on behalf of this CU.

10 Enable SDLC for the port.

For example, to connect a CU to a bridge/router, enable the SDLC Protocol using:

```
SETDefault !<port> -SDLC PCOntrol = Enable
```

11 Check that there is a one-to-one port-to-path mapping by displaying the PORT Service PAths parameter by entering:

```
SHow !1 -PORT PAths
```

12 Enable the CU using:

```
SETDefault !<CU name> -SDLC CUCONTrol = Enabled
```

Both the port and the CU must be enabled for an SDLC link to operate. The port and the CU may be enabled in any order. Be sure you have enabled the CONTrol parameter in the PORT Service and PATH Service.

Verifying the Configuration

After you have configured SDLC, you can display SDLC port and CU configuration information. You can also add or delete CUs assigned to a port. Deleting all CUs assigned to a port also deletes the SDLC configuration of the port.

To display all of the SDLC port parameters configured for the specified port and the CU configuration for all CUs assigned to that port, enter:

```
SHow -SDLC PCONFig
```

To display the SDLC configuration for a specific port, for example port 1, enter:

```
SHow !1 -SDLC PCONFig
```

Display the value of all CU-related parameters for each CU by entering:

```
SHow !* -SDLC CUCONFig
```

To display a specific CU, use:

```
SHow !<CU name> -SDLC CUCONFig
```

You can display a log of SDLC activity by entering:

```
SHow -SDLC SdlcLOG
```

The log displays a history of the most recent 256 log entries including the following actions:

- Control unit activation or deactivation
- Control unit failure

Using Frame Relay Access

If SDLC devices attached to the bridge/router need to communicate with a FEP attached directly to the bridge/router through Frame Relay, the bridge/router must be set up to support the mapping of LLC2 traffic to Frame Relay. If you have not already configured Frame Relay, perform the procedures in Chapter 42.

APPN over SDLC

You can configure Advanced Peer-to-Peer Networking (APPN) traffic to run over SDLC. To configure the bridge/router network node to run over SDLC, you first configure the SDLC port and path attributes between the network node and the partner node using the procedures in this chapter. You then configure the APPN network node following the procedures in Chapter 10; the procedures are similar to configuring APPN over other data link control (DLC) types, except that you do the following tasks:

- When setting the -APPN PortDef parameter for APPN ports, set the DLC type to SDLC and optionally, set the DatMode and ROle values.
- Configure adjacent link stations using the -APPN SdlcAdjLinkSta parameter and configure Dependent LU Requestor (DLUr) link stations using the -APPN SdlcDlurLinkSta parameter.

For information about these APPN Service parameters, refer to Chapter 5 in *Reference for NETBuilder Family Software.*

Figure 22-5 shows a configuration in which APPN traffic is being sent over SDLC connections. The NETBuilder II bridge/router acting as a network node is shown sending APPN over SDLC connections to a peer NETBuilder II network node, to a network control program (NCP), and to a 3174 PU2.0 type node. The network node is also serving as a DLUr for the VTAM Dependent LU server (DLUs).

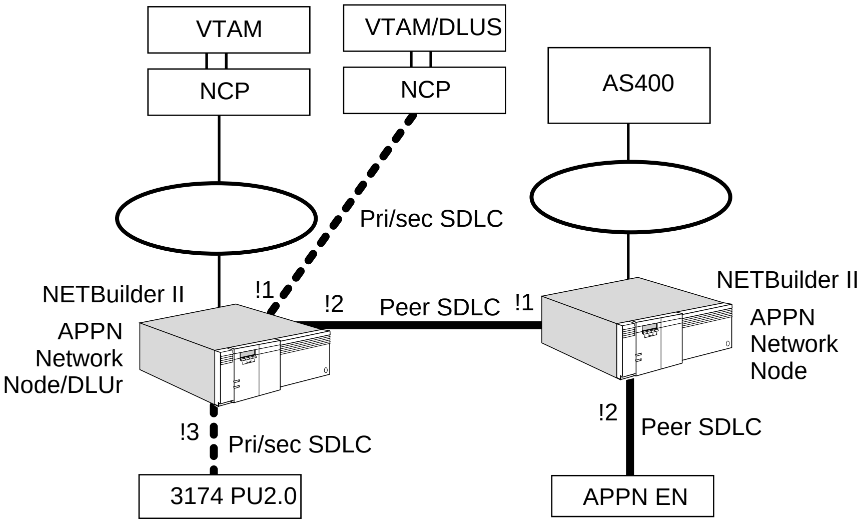


Figure 22-5 APPN Traffic over SDLC

You can send either APPN ISR traffic or APPN HPR traffic over SDLC connections. For information about configuring APPN Intermediate Session Routing (ISR) over SDLC, follow the procedures in Chapter 10. For information about configuring APPN High Performance Routing (HPR), refer to Chapter 11.

How SDLC Conversion Works

SDLC devices generally are referred to as physical units (PUs), control units (CUs), and linkstations. CU is used in parameter descriptions and names. The term PU is used in some examples and general discussion and when referring to device type. The term link station may also be used to refer to the device type. Except for parameter names or specific device types, these terms may be used interchangeably.

SDLC connectivity allows SDLC devices to communicate with local or remote non-SDLC (LLC2, Frame Relay devices, or other remote SDLC devices, using an SDLC connection to your bridge/router. SDLC polling and response occur locally between the SDLC device and the NETBuilder II bridge/router; the SNA data stream is tunneled through the network using the DLSw protocol. Figure 22-6 shows a typical SDLC/DLSw configuration.

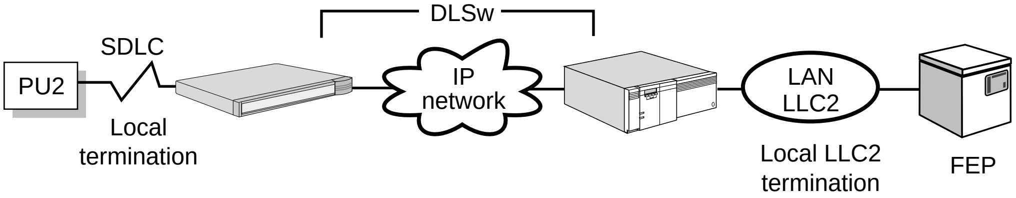


Figure 22-6 SDLC/DLSw Configuration

The NETBuilder II bridge/router provides SDLC connectivity by mapping the SDLC device to a “virtual” LLC2 device, or an LLC2 device to an SDLC device. Other systems in the network communicate with this LLC2 device, for example, through DLSw. The bridge/router passes the data to and from the SDLC device.

In addition to the configuration information required to communicate with the SDLC device, the bridge/router must be given the LLC2 information used to communicate with non-SDLC devices, for example, MAC and SAP values.

Operating the bridge/router with SDLC is accomplished by mapping (or conversion) of SDLC connections into LLC2 connections. This section describes the key aspects of this mapping, how the configuration parameters relate to (and affect) the mapping behavior, and illustrates the configuration for common applications.

Both SDLC and LLC2 are reliable data link protocols. They provide sequenced, acknowledged, and retransmitted delivery of data frames, and include special frames for session initiation and termination. Although similar, SDLC and LLC2 operate in different environments using different modes. Because of this different frame sequences are used for session initiation, and different addressing schemes are used in each service.

Data Link Switch is a protocol (defined by RFC 1434) used to link LLC2 sessions together across an internetworked reliable transport protocol. The protocol definition for DLSw includes addressing schemes and session startup sequences that readily correspond to LLC2 addressing and session startup.

To link SDLC sessions with LLC2 or DLSw sessions, the bridge/router must handle two major functions: address mapping and session initiation.

Address Mapping

In an LLC2 (LAN) or DLSw environment, addressing consists of MAC/SAP pairs. Each station has a MAC (LAN) address; every frame sent from one station (A) to another station (B) contains both a source and a destination MAC address to distinguish the sending and receiving stations on the shared-access medium. Each LLC2 frame also contains a source and destination LLC2 SAP (LSAP). A pair of LAN stations may have multiple sessions between them; the LSAP values are used to distinguish between frames belonging to different sessions.

In an SDLC environment, each frame carries only a single address value; the identifier of the secondary station that is to receive, or that sent, this frame.

To allow stations in an LLC2 environment to communicate with an attached SDLC station, the bridge/router maps a set of LLC2 (LAN) addresses to each CU. This address mapping is configured using the CULocalMac, CURemoteMac, CULocalSap, and CURemoteSap parameters in the SDLC Service. The bridge/router appears as a LAN-based CU mapped to the SDLC CU, which uses the CULocalMac and CULocalSap to communicate with other LAN stations. Figure 22-7 shows various types of address mapping for SDLC.

In the LLC2 or DLSw environment, the CUs supported by the bridge/router appear to be attached to the LAN: Frames can be sent to them using the MAC/SAP values assigned as CULocalMac and CULocalSap. Frames sent by the bridge/router on behalf of the CU use the CULocalMac and CULocalSap values as the source address values.

If the bridge/router is initiating an LLC2 session on behalf of a CU (refer to “Session Initiation” on page 22-12), it must know which LLC2 station to send the connection request to. This destination is determined by the CURemoteMac and CURemoteSap parameter values. These values are used as the destination for LLC2 frames when the bridge/router initiates such sessions.

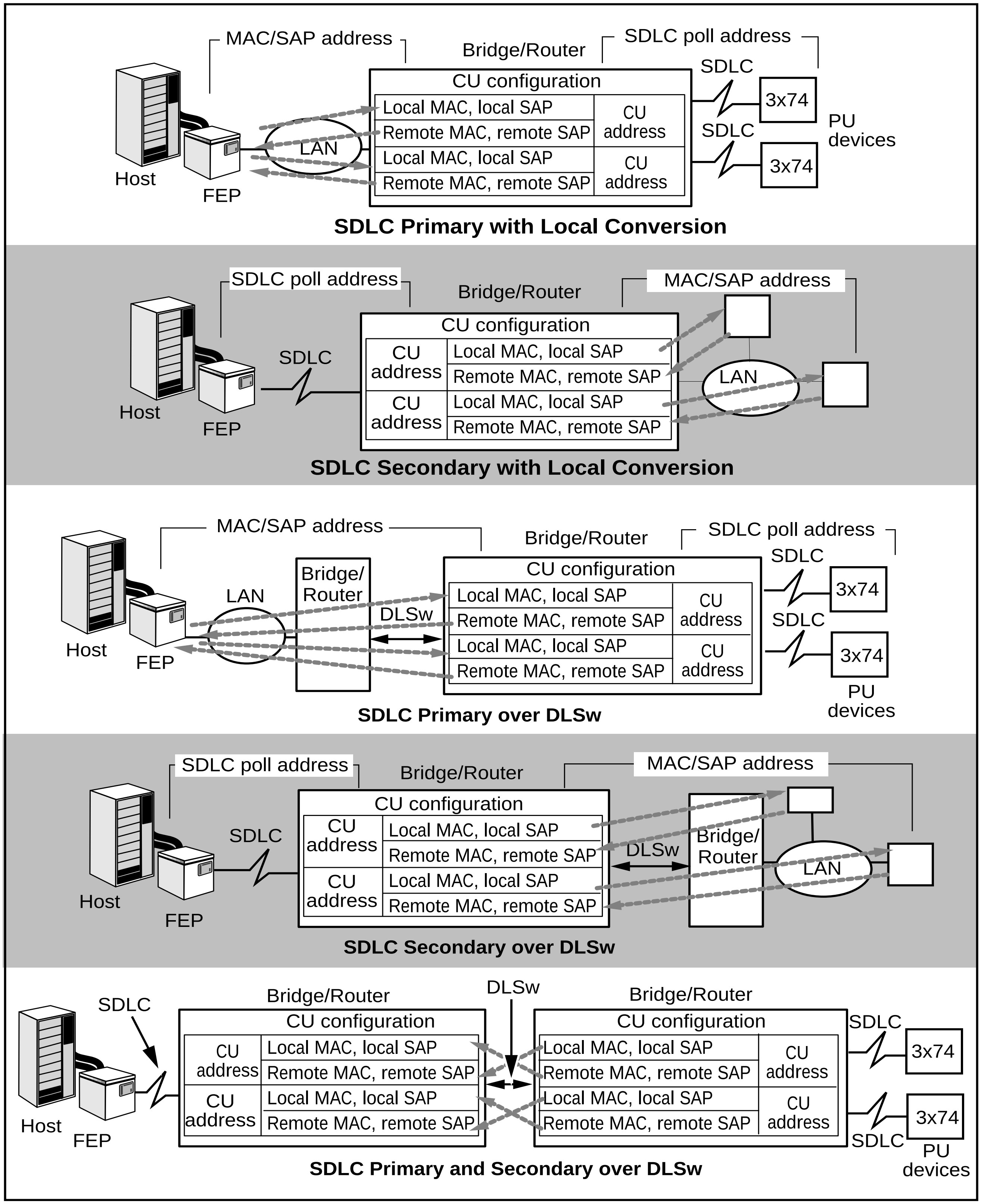


Figure 22-7 SDLC Address Mapping

Session Initiation

In addition to address mapping, the SDLC configuration determines how the bridge/router initiates SDLC and LLC2/DLSw connections. The CUMOde parameter for the specific CU, in conjunction with the SDLC role of the port (the PROle parameter) determines the system's behavior as follows:

- In an SDLC environment, datalink connections are initiated by the bridge/router acting as the primary station. The primary station controls the operation of other secondary stations. When acting as an SDLC secondary station, the bridge/router accepts connections only. Use the CUMode parameter to configure the bridge/router to either originate a network connection request or answer a connection request from the network.
- In an LLC2 or DLSw environment, either of the two systems involved in a session may initiate the datalink connection. The CUMode parameter also determines whether and when the NETBuilder II bridge/router initiates LLC2 or DLSw connections.

When CUMOde is set to Originate, the bridge/router initiates datalink connections in the LLC2/DLSw environment. Contact is achieved by sending an XID or SNRM and receiving a response. When the bridge/router is secondary, contact is achieved by receiving an XID or SNRM from the primary station. However, if the bridge/router is a secondary station in Originate mode, it will not respond until it has completed the LLC2/DLSw connection.

When the CUMOde is set to Answer, the bridge/router will only accept datalink connections from the LLC2/DLSw environment. When a connection request is received, the bridge/router attempts to set up the corresponding SDLC connection before accepting the LLC2/DLSw connection. When you set up the SDLC connection, an XID or SNRM is sent if the bridge/router is acting as a primary station; if the bridge/router is acting as a secondary station, an XID or SNRM response is made.

23 CONFIGURING SDLC AND HDLC TUNNELING FOR SNA NETWORKS

This chapter describes how to configure tunneling for synchronous data link control (SDLC) and high-level data link control (HDLC) traffic using the IBM Data Link Switching (DLSw) protocol.

For conceptual information on how SDLC and HDLC tunneling works, refer to "How SDLC and HDLC Tunneling Works" on page 23-5. For information about the SHDlc Service parameter, refer to Chapter 52 in Reference for NETBuilder Family Software.

Configuring SDLC and HDLC Tunneling

This section describes how to configure the bridge/router for SDLC and HDLC tunneling, by referring to Figure 23-1. The figure shows an IBM host connected to an IBM controller through NETBuilder bridge/routers and an IP network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the IP addressing and IP routing protocols on the appropriate ports according to Chapter 6.
- Obtain the IP addresses for bridge/routers on each side of the TCP/IP connection.

Procedure

Configure ports, paths, SDLC and HDLC tunneling, and data link switching by referring to the example in Figure 23-1 and completing the steps under "Configuring Router A" and "Configuring Router B." Table 23-1 lists the commands used in these steps.

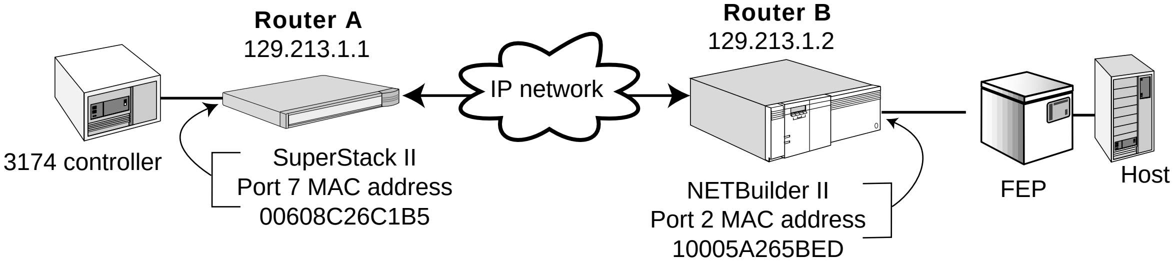


Figure 23-1 SDLC and HDLC Tunneling Example

Table 23-1 Commands to Configure SDLC and HDLC Tunneling and Data Link Switching

Commands Entered on Bridge/Router A	Commands Entered on Bridge/Router B
SHow -PORT CONFiguration	SHow -PORT CONFiguration
SETDefault !1 -IP NETaddr = 129.213.1.1	SETDefault !1 -IP NETaddr = 129.213.1.2
SETDefault !7 -PORT OWNer = SHDlc	SETDefault !2 -PORT OWNer = SHDlc
SETDefault !7 -PATH CLock = External	SETDefault !2 -PATH CLock = External
SETDefault !7 -PATH CONNector = RS232	SETDefault !2 -PATH CONNector = RS232
SETDefault !7 -PATH LineType = Leased	SETDefault !2 -PATH LineType = Leased
SETDefault !7 -PATH BAud = 19.2	SETDefault !2 -PATH BAud = 19.2
SETDefault !7 -PATH DUplex = Full	SETDefault !2 -PATH DUplex = Full
SETDefault -TCP CONTrol = KeepAlive	SETDefault -TCP CONTrol = KeepALive
SETDefault -TCP KeepALive = 3	SETDefault -TCP KeepALive = 3
SETDefault -DLSw Interface = 129.213.1.1	SETDefault -DLSw Interface = 129.213.1.2
SETDefault -DLSw CONTrol = (EnableSNA, DisableNetBIOS)	SETDefault -DLSw CONTrol = (EnableSNA, DisableNetBIOS)
SETDefault !7 -PATH CONTrol = Enable	SETDefault !2 -PATH CONTrol = Enable
ADD !1 -DLSw PEer 129.213.1.2	ADD !1 -DLSw PEer 129.213.1.1
SystemInfo	SHow -SYS MacAddrFormat
	SETDefault !2 -SYS MacAddrFormat = Noncanonical
SHow -PATH MacAddress	SHow -PATH MacAddress
SETDefault !7 -SHDlc PEer = %10005A265BED	SETDefault !2 -SHDlc PEer = %00608C26C1B5

Configuring Router A

To configure router A, follow these steps:

1 Display the port configuration by entering:

```
SHow -PORT CONFiguration
```

The display shows the ownership status of each port.

2 Define the IP address for the port through which the router is going to tunnel by entering:

```
SETDefault !1 -IP NETaddr = 129.213.1.1
```

3 Set the port ownership of serial port 7 to SHDlc by entering:

```
SETDefault !7 -PORT OWNer = SHDlc
```

When you use a WAN port, you need to configure the port owner. SDLC and HDLC tunneling use only WAN ports.

The number of SHDlc ports a NETBuilder II bridge/router can support is the number of WAN paths it can operate simultaneously.

4 Display attributes for all available paths by entering:

```
SHow -PATH CONFiguration
```

5 Set the attributes for the SHDlc line by entering:

```
SETDefault !7 -PATH CLock = External
SETDefault !7 -PATH CONNector = RS232
SETDefault !7 -PATH LineType = Leased
SETDefault !7 -PATH BAUD = 19.2
SETDefault !7 -PATH DUplex = Full
```

If you use a single- port WAN adapter, set the -PATH ENCoding parameter to NRZ.

SHDLC only supports full-duplex operation.

After configuring values using the PATH Service, you may receive a message telling you to re-enable the path. If you receive this message, re-enable the path with the SETDefault !<path> -PATH CONTrol = Enable command.

6 Enable the transmission of TCP keepalive packets by entering:

```
SETDefault -TCP CONTrol = KeepAlive
```

TCP keepalive packets notify the bridge/router when the TCP connection has ended. Without TCP keepalive packets, the bridge/router will not detect that the TCP connection is down due to an abnormal situation. The data link switching sessions may remain active even though the corresponding TCP session has ended.

7 Specify the number of contiguously missed keepalive packets that brings down the TCP session.

For example, if you want three retries, enter:

```
SETDefault -TCP KeepAliveLimit = 3
```

8 Configure an IP address to connect traffic to and from the router.

The address must be one that has been defined in the router using the SETDefault !<port> -IP NETaddr syntax. This address is the only address used for data link switching.

If you are configuring your NETBuilder II bridge/router as an IP router, the port associated with this IP address must be active before any packets can be sent to or received by this IP address. Select an IP address associated with a port that is always up or is the most reliable, such as a LAN port.

To map the specified DLSw tunnel to the local IP address of bridge/router A, enter:

```
SETDefault -DLSw Interface = 129.213.1.1
```

All Internet addresses for connected bridge/routers must be known in the local bridge/router's routing table, either dynamically through RIP or OSPF, or statically configured in the IP routing tables.

9 Enable data link switching for SNA traffic on the port by entering:

```
SETDefault -DLSw CONTrol = (EnableSNA, DisableNetBIOS)
```

This setting allows SNA traffic to flow through the data link switch and disables NetBIOS traffic. SNA traffic must be enabled for SDLC and HDLC tunneling to work.

If you are going to use the prioritization feature of DLSw, refer to "Prioritizing DLSw Traffic" in Chapter 24 before proceeding to the next step.

10 Configure the DLSw tunnel peer IP connection by entering the following command. You also must specify the tunnel ID, a peer network address, and optionally, a name for the tunnel connection.

```
ADD !1 -DLSw PEer 129.213.1.2
```

When configuring the Internet address for the tunnel peer, you do not specify the port number of the bridge/router where the connection will be made. When a tunnel connection is made, the bridge/router determines the port through which the peer Internet address can be reached. When a peer has been defined and enabled, the system continuously retries to connect to the peer until a TCP connection is established between the system and the peer.

11 From the peer router, display the format of the peer MAC address by entering:

```
SHow -SYS MacAddrFormat
```

12 To display the local MAC address on the SuperStack bridge/router, enter:

```
SystemInfo
```

13 To convert the MAC address to noncanonical format, you must enter the MacAddrConvert command on a NETBuilder II bridge/router. This command is not available on the SuperStack bridge/router.

14 Display the peer router MAC address by entering:

```
SHow -PATH MacAddress
```

With this display, you can obtain the peer router MAC address that you configure in the next step.

15 Set the MAC address for the peer serial port that the local SDLC port is communicating with by entering:

```
SETDefault !7 -SHDlc PEer = %10005A265BED
```

Configuring Router B

To configure router B, repeat steps 1–11 in the preceding procedure, then continue with the following steps (performed on a NETBuilder II bridge/router):

1 If the peer MAC address displayed is in canonical format, set it to noncanonical by entering:

```
SETDefault !7 -SYS MacAddrFormat - Noncanonical
```

2 Display the peer router MAC address by entering:

```
SHow -PATH MacAddress
```

With this display, you can obtain the peer router MAC address that you configure in the next step.

3 Set the MAC address for the peer serial port that the local SDLC port is communicating with using:

```
SETDefault !<port> -SHDLc PEer
```

Verifying the Configuration

After you have configured a tunnel connection using data link switching, you can display information to verify the connection.

To display complete configuration information, enter:

```
SHow -DLSw CONFiguration
```

The display shows the settings you have configured.

To display the peer information, enter:

```
SHow -SHDLc -PEer
```

The following display is an example of this information:

```
---------------------------SHDlc PEer----------------------------
 Local Port    Local MacAddress     Circuit State Peer MacAddress
     !4         %1000405011DC         CONNECTED    %100040605D8A
```

When shown in the display, SAP E8 represents an HDLC tunnel.

Displaying Circuits

To display the status of circuits, enter:

```
SHow -DLSw CIRcuits
```

Information similar to the following is displayed:

```
---------------------------------------Circuits----------------------------------
Local             DL Corr.  Port Peer              DL Corr.  State     Peer IP
Name/Address                     Name/Address                          Address
%00608C26C1B5  04 376B008E  4    %10005A265BED  04  37250047  CONNECTD  129.213.1.2
%0020AF00DCC8  04 171A00C9  6    %40000003172A  04  86FA0013  CONNECTD  200.200.1.254
%100040600B03  00 00000000  ?    %100040A0E8E1  00  00000000  DISC      200.200.1.254
%400001111111  08 00000000  ?    %400002222222  08  00000000  DISC      200.200.1.254
%400011600000  04 00000000  ?    %0020AF00B940  04  00000000  DISC      200.200.1.254
%400011600000  34 00000000  ?    %0020AFEE9630  34  00000000  DISC      200.200.1.254
```

How SDLC and HDLC Tunneling Works

The SDLC and HDLC tunneling features enable NETBuilder II bridge/routers to send SDLC or HDLC frames across IP networks through DLSw tunnels. Two bridge/routers interconnect a point-to-point SDLC or HDLC link. They encapsulate SDLC or HDLC frames sent between the two end points and tunnel them through an IP network.

A typical use of SDLC and HDLC tunneling is to connect a host computer and a remote terminal or controller. In Figure 23-1, two end points of an SDLC link (a 3174 controller and a host) are interconnected by two intermediary bridge/routers. The 3174 controller is connected to port 7 on router A, and the host FEP is connected to port 2 on router B.

24 CONFIGURING DATA LINK SWITCHING FOR SNA AND NETBIOS NETWORKS

This chapter describes how to configure data link switching on your system to connect networks running IBM Systems Network Architecture (SNA) and NetBIOS traffic over Transmission Control Protocol/ Internet Protocol (TCP/IP) using the IBM Data Link Switching (DLSw) Protocol.

For conceptual information on how data link switching works, refer to "How Data Link Switching Works" on page 24-27 and to RFC 1795. The 3Com implementation of DLSw is based on this standard. Also, to simplify configuration, you can use DLSw multicast. For more information, refer to Chapter 25.

Configuring for SNA

This section describes how to configure the bridge/router for both ends of an SNA connection using the DLSw protocol.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the token ring LAN as described in Chapter 5.
- Configure the IP addressing and IP routing protocols on the appropriate ports according to Chapter 6.
- Obtain the IP addresses for bridge/routers on each side of the TCP/IP connection.

Figure 24-1 shows a sample data link switching configuration for an SNA environment.

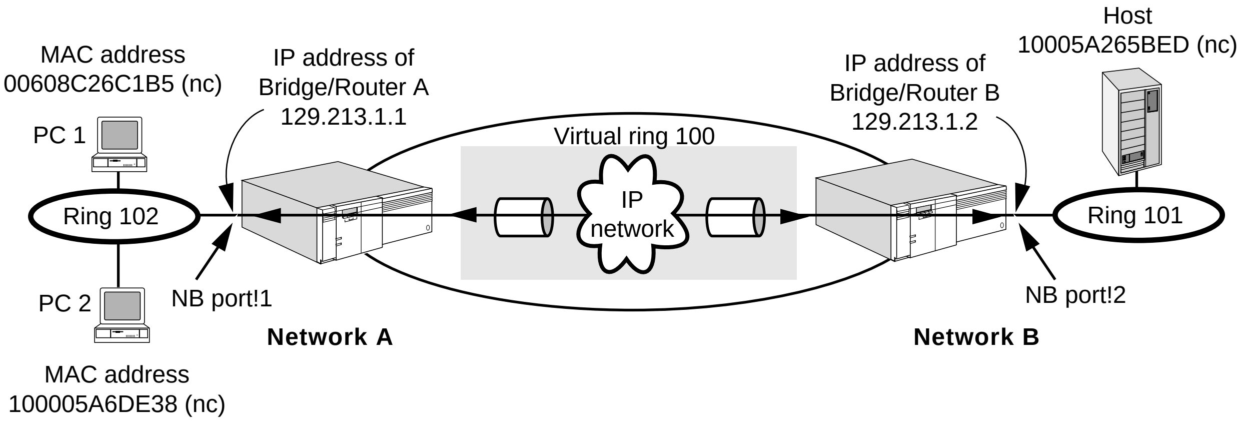


Figure 24-1 Configuring Data Link Switching for SNA

Table 24-1 lists the commands used to configure the example in Figure 24-1.

Table 24-1 Commands to Configure Data Link Switching for SNA

Commands Entered on Bridge/Router A	Commands Entered on Bridge/Router B
SETDefault -TCP CONTrol = KeepALive	SETDefault -TCP CONTrol = KeepALive
SETDefault -TCP KeepALive = 3	SETDefault -TCP KeepALive = 3
SETDefault !1 -LLC2 CONTrol = Enable	SETDefault !2 -LLC2 CONTrol = Enable
SETDefault !1 -SR RouteDiscovery = LLC2	SETDefault !2 -SR RouteDiscovery = LLC2
SETDefault -LLC2 TUNnelVRing = 100	SETDefault -LLC2 TUNnelVRing = 100
SETDefault -DLSw MOde = Secure	SETDefault -DLSw MOde = Secure
SETDefault -DLSw Interface = 129.213.1.1	SETDefault -DLSw Interface = 129.213.1.2
ADD !1 -DLSw PEer 129.213.1.2	ADD !1 -DLSw PEer 129.213.1.1
SETDefault -DLSw CONTrol = EnableSNA, DisableNetBios	SETDefault -DLSw CONTrol = EnableSNA, DisableNetBios

Procedure

To configure data link switching for SNA bridge/router A, see Figure 24-1 and follow these steps:

1 Enable transmission of Transmission Control Protocol (TCP) keepalive packets by entering:

```
SETDefault -TCP CONTrol = KeepAlive
```

TCP keepalive packets notify the bridge/router when the TCP connection has ended. Without TCP keepalive packets, the bridge/router will not detect that the TCP connection is down due to an abnormal situation. This can result in data link switching sessions being kept active even though the corresponding TCP session has ended.

2 Specify the number of contiguously missed keepalive packets that will bring down the TCP session.

For example, if you want three retries, enter:

```
SETDefault -TCP KeepAliveLimit = 3
```

3 Enable LLC traffic from a port to be tunneled through data link switching by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

This command enables LLC2 traffic on port 1.

Enable route discovery by entering:

```
SETDefault !1 -SR RouteDiscovery = LLC2
```

This command enables route discovery for LLC2 on port 1.

Repeat this step for each port you are configuring.

DLSw is affected by parameter settings in other services. For more information, refer to “Configuring LLC2 with Other Services” on page 21-3.

4 Assign a unique virtual ring number for the data link switching cloud.

This ring number is used by source routing and some data link switching Switch-to-Switch Protocol (SSP) messages. For example, to configure the virtual tunnel ring, enter:

```
SETDefault -LLC2 TUNnelVRing = 100
```

When using source routing, the internetwork becomes a virtual ring. If your end systems are using token ring source routing, the bridge/router and the IP tunnel appear to the end systems as a source route bridge with a token ring network attached to the other side of the bridge/router.

This virtual ring number must match on all peer bridge/routers used for tunneling and must be unique within the token ring network.

5 Configure the desired mode of operation.

To configure secure mode, enter:

```
SETDefault -DLSw MOde = Secure
```

The router accepts connections only from data link switches defined in the ADD PEer parameter.

To configure for default prioritization, enter:

```
SETDefault -DLSw MOde = Secure, DefaultPRioritized
```

You can also configure the mode to multicast. For more information, refer to Chapter 25.

6 Configure an IP address to connect traffic to and from the router.

The address must be one that has been defined in the router using the SETDefault !<port> -IP NETaddr syntax. This address is the only address used for data link switching.

If you are configuring your NETBuilder II bridge/router as an IP router, the port associated with this IP address must be active before any packets can be sent to or received by this IP address. Select an IP address associated with a port that is always up or is the most reliable, such as a LAN port.

To map the specified DLSw tunnel to the local IP address of bridge/router A, enter:

```
SETDefault -DLSw Interface = 129.213.1.1
```

All Internet addresses for connected bridge/routers must be known in the routing table of the local bridge/router, either dynamically through RIP or OSPF, or statically configured in the IP routing tables.

7 Enable data link switching for SNA traffic on the port by entering:

```
SETDefault -DLSw CONTrol = (EnableSNA, DisableNetBIOS)
```

This setting allows SNA traffic to flow through the data link switch and disables NetBIOS traffic.

If you are going to use the prioritization feature of DLSw, refer to "Prioritizing DLSw Traffic" on page 24-14 before proceeding to step 8.

8 Configure the DLSw tunnel peer IP connection by entering:

```
ADD !1 -DLSw PEer 129.213.1.2
```

When configuring the Internet address for the tunnel peer, you do not specify the port number of the bridge/router where the connection will be made. When a tunnel connection is made, the bridge/router determines the port through which the peer Internet address can be reached. When a peer has been defined and enabled, the system continuously retries to connect to the peer until a TCP connection is established between the system and the peer.

9 To configure bridge/router B, repeat steps 1–8.

Configuring for NetBIOS

This section describes how to configure data link switching for NetBIOS traffic.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the token ring LAN as described in Chapter 5.
- Configure the IP addressing and IP routing protocols on the appropriate ports according to Chapter 6.
- Obtain the IP addresses for both bridge/routers on either side of the TCP/IP connection.

Procedure

Figure 24-2 shows a sample DLSw configuration for a NetBIOS environment.

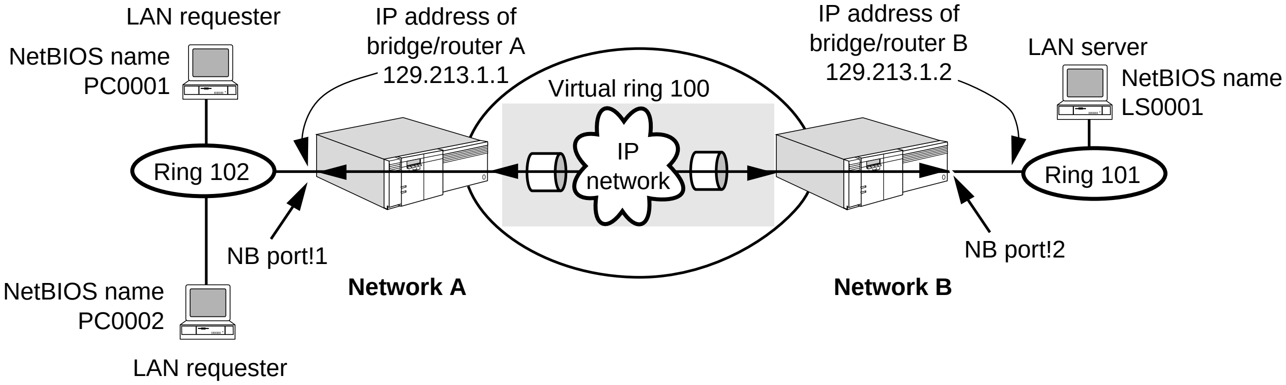


Figure 24-2 Configuring Data Link Switching for NetBIOS

Table 24-2 lists the commands used for this configuration.

Table 24-2 Commands to Configure Data Link Switching for NetBIOS

Commands Entered on Bridge/Router A	Commands Entered on Bridge/Router B
SETDefault -TCP CONTrol = KeepALive	SETDefault -TCP CONTrol = KeepALive
SETDefault -TCP KeepALive = 3	SETDefault -TCP KeepALive = 3
SETDefault !1 -LLC2 CONTrol = Enable	SETDefault !2 -LLC2 CONTrol = Enable
SETDefault !1 -SR RouteDiscovery = LLC2	SETDefault !2 -SR RouteDiscovery = LLC2
SETDefault !1 -SR RingNumber = 102	SETDefault !2 -SR RingNumber = 101
SETDefault -LLC2 TUNnelVRing = 100	SETDefault -LLC2 TUNnelVRing = 100
SETDefault -DLSw Interface = 129.213.1.1	SETDefault -DLSw Interface = 129.213.1.2
ADD !1 -DLSw PEer 129.213.1.2	ADD !1 -DLSw PEer 129.213.1.1
SETDefault -DLSw CONTrol = DisableSNA, EnableNetBios	SETDefault -DLSw CONTrol = DisableSNA, EnableNetBios

To configure data link switching for NetBIOS on bridge/router A, see Figure 24-2 and follow these steps:

1 Enable transmission of TCP keepalive packets by entering:

```
SETDefault -TCP CONTrol = KeepAlive
```

TCP keepalive packets notify the bridge/router when the TCP connection has ended. Without TCP keepalive packets, the bridge/router will not detect that the TCP connection is down due to an abnormal situation. The data link switching sessions may be kept active even though the corresponding TCP session has ended.

2 Specify the number of contiguously missed keepalive packets that will bring down the TCP session.

For example, if you want three retries, enter:

```
SETDefault -TCP KeepAliveLimit = 3
```

3 Enable LLC traffic from a port to be tunneled through data link switching by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

This command enables traffic on port 1.

Enable route discovery by entering:

```
SETDefault !1 -SR RouteDiscovery = LLC2
```

This command enables route discovery for LLC2 on port 1.

Assign the ring number to the local port by entering:

```
SETDefault !1 -SR RingNumber = 102
```

Repeat this step for each port you are configuring.

4 Assign a unique virtual ring number for the data link switching cloud.

This ring number is used by source routing and some data link switching SSP messages. For example, to configure the virtual tunnel ring, enter:

```
SETDefault -LLC2 TUNnelVRing = 100
```

When using source routing, the internetwork becomes a virtual ring. If your end systems are using token ring source routing, the bridge/router and the IP tunnel appear to the end systems as a source route bridge with a token ring network attached to the other side of the bridge/router.

This virtual ring number must match on all peer bridge/routers used for tunneling and must be unique within the token ring network.

5 Configure the desired mode of operation.

To configure secure mode, enter:

```
SETDefault -DLSw MOde = Secure
```

The router accepts connections only from data link switches defined in the ADD !<tunnelid> PEer parameter.

6 Configure an IP address to connect traffic to and from the router.

The address must be one that has been defined in the router using the SETDefault !<port> -IP NETaddr syntax. This address is the only address used for data link switching. To map the specified DLSw tunnel to the local IP address of bridge/router A, enter:

```
SETDefault -DLSw Interface = 129.213.1.1
```

All Internet addresses for connected bridge/routers must be known in the routing table of the local bridge/router, either dynamically through RIP or OSPF, or statically configured in the IP routing tables.

7 Enable data link switching for NetBIOS traffic on the port by entering:

```
SETDefault -DLSw CONTrol = (EnableNetBios, DisableSNA)
```

This setting allows NetBIOS traffic and disables SNA traffic from flowing through the data link switch.

If you are going to use the prioritization feature of DLSw, refer to “Prioritizing DLSw Traffic” on page 24-14 before proceeding to step 8.

8 Configure the DLSw tunnel peer IP connection by entering:

```
ADD !1 -DLSw PEer 129.213.1.2
```

When configuring the Internet address for the tunnel peer, you do not specify the port number of the bridge/router where the connection will be made. When a tunnel connection is made, the bridge/router determines the port through which the peer Internet address can be reached.

9 To configure bridge/router B, repeat steps 1–8, and enter the addresses of the session partners.

Verifying the Configuration

After you have configured a tunnel connection using data link switching, you can display information to verify the connection.

To display complete configuration information, enter:

```
SHow -DLSw CONFiguration
```

The display shows the settings you have configured.

Displaying Connections

To display the control status of your connections between two data link switched SNA networks, enter:

```
SHow -DLSw CONNections
```

The display shows whether data link switching has established a connection with the Peer IP address. For example, the following display shows the results of the sample configuration in Figure 24-1 on the bridge/router A side:

```
-------------------------Connections-------------------------
Peer IP Address       State               No. of Circuits
129.213.1.2           ACTIVE              0
```

To determine whether TCP has established connections, display TCP port information, and to verify whether the two data link switching port numbers are active or connected, enter:

```
SHow -TCP CONNections
```

This display shows the actual TCP connections. There are two connections for each DLSw tunnel.

```
---------------------TCP Connection Table-------------------------
Loc Address      Port    Rem address     Port    State      ConnID
129.213.1.1      2065    129.213.1.2     2067    estab     1966085
129.213.1.1      2067    129.213.1.2     2065    estab     1900550
```

Displaying Circuits

To display the status of circuits, enter:

SHow -DLSw CIRcuits

Information similar to the following is displayed:

```
---------------------------------------Circuits----------------------------------
Local             DL Corr.  Port Peer               DL Corr.  State     Peer IP
 Name/Address                     Name/Address                           Address
%00608C26C1B5  04 376B008E  4    %10005A265BED  04  37250047  CONNECTD  129.213.1.2
%0020AF00DCC8  04 171A00C9  6    %40000003172A  04  86FA0013  CONNECTD  200.200.1.254
%100040600B03  00 00000000  ?    %100040A0E8E1  00  00000000  DISC      200.200.1.254
%400001111111  08 00000000  ?    %400002222222  08  00000000  DISC      200.200.1.254
%400011600000  04 00000000  ?    %0020AF00B940  04  00000000  DISC      200.200.1.254
%400011600000  34 00000000  ?    %0020AFEE9630  34  00000000  DISC      200.200.1.254
```

For more information about the possible states, refer to the CIRcuits parameter in Chapter 19 of *Reference for NETBuilder Family Software*.

Displaying LLC Sessions

Logical link control (LLC) displays media access control (MAC) addresses in canonical format. Use the MacAddrConvert command to convert a MAC address in canonical format to noncanonical format. To display the status of configured sessions, enter:

SHow -LLC2 SESSions

Information similar to the following is displayed:

```
-------------------------LLC2 Sessions-------------------------
...........LLC2 Active Source Mac Address:%0020AF1D2C10
Source:%0020AF1D2C10 Sap:04 Dest:%02608C1A0CE7Sap:04 Port:!1
 -ACTIVE
RIF: Transparent Frame
```

Displaying Cache

You can also display the contents of the names in your NetBIOS names cache by entering:

SHow -DLSw NameCache

The cache displays both static and dynamic names:

```
-----------------------Netbios Names Cache-----------------------
Peer: IP Address            Netbios Name
129.213.1.2                 LANSERVER1
```

You can display the contents in the MAC addresses cache by entering:

SHow -DLSw MacCache

When verifying MAC addresses for the sample configuration shown in Figure 24-1, information similar to the following is displayed:

```
-------------------------Mac Addresses Cache---------------------
Peer: IP Address                  Mac Address
129.213.1.2                       %100051265BED
```

Displaying the DLSw Activity Log

You can display a log of DLSw activity by entering:

```
SHow -DLSw DlswLOG
```

The log displays a history of the most recent log entries including the following actions:

- Circuit activation or deactivation
- Circuit failure
- Tunnel activation or deactivation
- Tunnel failure
- Capabilities exchange accepted or rejected

The following display is an example of this log:

```
50  Tue May 28 17:51:54 1996 Capex Ack IP 192.100.2.3       Vectors: 81 82 83 86 84
#49  Tue May 28 17:51:54 1996 Capex Ack IP 192.100.100.1     Vectors: 81 82 83 86 84
#48  Tue May 28 17:51:54 1996 Tunnel UP IP 192.100.2.3
#47  Tue May 28 17:51:54 1996 Tunnel UP IP 192.100.100.1
#46  Tue May 28 17:09:11 1996 Circuit DOWN LMAC 100040501175 LSAP 0C RMAC 100040 5011DB
 RSAP 0C IP 192.100.2.3
#45  Tue May 28 17:09:11 1996 Circuit DOWN LMAC 100040501175 LSAP 08 RMAC 100040 5011DB
 RSAP 08 IP 192.100.2.3
#44  Tue May 28 17:09:11 1996 Circuit DOWN LMAC 100040501175 LSAP 04 RMAC 100040 5011DB
 RSAP 04 IP 192.100.2.3
```

Displaying the DLSw End-Station Topology

You can configure the bridge/router to collect end-station topology information for the DLSw network topology and display it to help troubleshoot the network. Before you can display the topology, you must first specify whether you want logical unit (LU) topology or physical unit (PU) topology information collected.

To collect end-station topology use:

```
SETDefault -DLSw SnaTopoCollect = (EnablePu | EnablePuLu |
  Disable)
```

To collect end-station topology PU and LU information, enter:

```
SETDefault -DLSw SnaTopoCollect = EnablePuLu
```

To collect end-station topology PU information only, enter:

```
SETDefault -DLSw SnaTopoCollect = EnablePU
```

To disable the collection of either LU or PU information, enter:

```
SETDefault -DLSw SnaTopoCollect = Disable
```

To display the DLSw topology map based on the end-station topology collection information, enter:

```
SHow -DLSw SnaTopoDisplay
```

The following display is an example showing the end-stations in a DLSw topology:

```
                    ----------------------------------------SNA End-Station Topology--------------------------------------------
PU Name : US3COMHQ.PU01BJ1  Node ID  : 05D 90100 Node Type: NN Dep. LU: Yes
MAC Addr: %00608C24F2F6 04  Port Num : 2       DLC Type : TR
Status  : ACTIVE         Active LU: 4       Bound LU : 1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
LU Name            Add      T   State    Pri. LU          | LU Name     Add  T   State    Pri. LU
--------           ---      -  -------- --------           +  --------     ---  -  --------  --------
LU01BJ1             2       2  BOUND   CNM01LU     |                                    3         ACTIVE
4                  ACTIVE                              |                                    5         ACTIVE
==================================================================================
PU Name :               Node ID  : 017 9079D   Node Type: 2.0  Dep. LU: Yes
MAC Addr: %10004060532C 04   Port Num : 7A        DLC Type : SDLC
Status  : ACTIVE          Active LU: 8         Bound LU : 1
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
LU Name            Add      T   State    Pri. LU          | LU Name     Add  T   State    Pri. LU
--------           ---      -  --------  --------          +  --------      ---  -   --------   --------
                    2         ACTIVE                 |                                    3         PNDACT
                    4         INACTIVE               |                                    5         ACTIVE
                    6         ACTIVE                 |                                    7         ACTIVE
                    8         ACTIVE                 | LUTRDSH                            9   2     BOUND     LU002MVS
==================================================================================
PU Name : US3COMHQ.DLSWWS1   Node ID  : 05D 00210  Node Type: EN   Dep. LU: No
MAC Addr: %00608C24F2F6 04   Port Num : 1        DLC Type : ETH
Status  : ACTIVE
==================================================================================
```

In this display, the end-stations shown are as follows:

- The first end-station with the PU name PU01BJ1is a token ring station with four dependent LUs in varying states. For example, the LU named LU01BJ1, is in the BOUND state with the primary LU (PLU) CNM01LU. The other LUs, which are not named, are active.

The second end-station is an SDLC 3174 with dependent LUs in varying states. It does not show a PU name because the XIDs exchanged were type 1 (and the name was not manually set).

- The third end-station, with the PU name DLSWWS1, is a PU 2.1 attached end-station through Ethernet, and has no dependent LUs.

In the display, each end-station description is separated by the double lines. This display shows both PU and LU topology information (obtained by specifying EnablePuLu for the SnaTopoCollect parameter.

If only PU information was collected, the display would not show the LU information.

For more information about this display, refer to the SnaTopoDisplay parameter description in Chapter 19 of *Reference for NETBuilder Family Software.*

When an SNMP Manager such as SunNet Manager or OpenView is used, more information about each end-station is displayed than is available through the NETBuilder SnaTopoDisplay parameter display.

Customizing the Configurations

This section describes how to customize data link switching configurations.

Defining a Non-Secure Host Configuration

By default, the bridge/router can accept DLSw tunnel connections from any other configured DLSw bridge/routers. By setting a bridge/router to a Secure state, unauthorized sites can be prevented from accessing a particular site. For less vital traffic, you can leave the bridge/router configured to accept tunnel connections from any remote bridge/router. If you plan to have terminal users at many different remote sites making tunnel connections to a site, you can use the NonSecure setting.

Figure 24-3 shows a configuration in which the bridge/router at a central host site accepts incoming tunnel connections from three branch offices to access the local site.

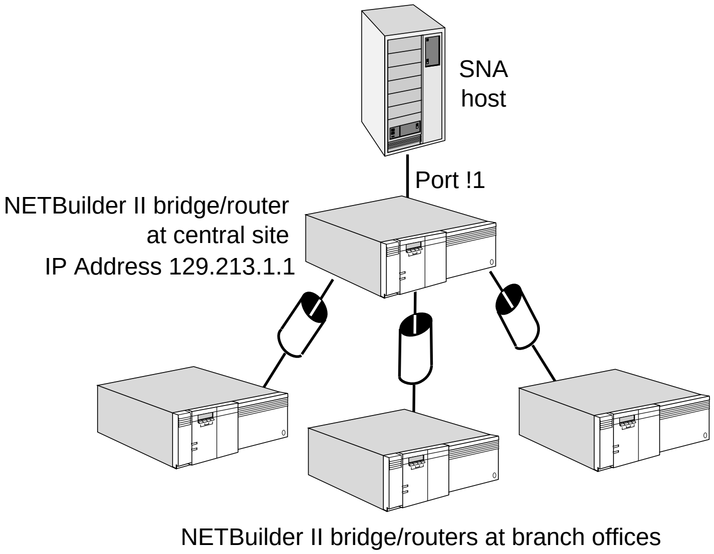

SNA
host
Port !1
NETBuilder II bridge/router
at central site
IP Address 129.213.1.1
NETBuilder II bridge/routers at branch offices

Figure 24-3 Data Link Switching Tunnel Configuration for Central bridge/router

Table 24-3 lists the commands used for this configuration.

Table 24-3 Commands to Configure Data Link Switching for a Central Site Bridge/Router

Commands Entered on the Central Site Bridge/Router
SETDefault -TCP CONTrol = KeepALive
SETDefault -TCP KeepALive = 3
SETDefault !1 -LLC2 CONTrol = Enable
SETDefault -LLC2 TUNnelVRing = 100
SETDefault -DLSw MOde = NonSecure
SETDefault -DLSw Interface = 129.213.1.1
SETDefault -DLSw CONTrol = EnableSNA, DisableNetBios

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the token ring LAN as described in Chapter 5.
- Configure the IP addressing and IP routing protocols on the appropriate ports according to Chapter 6.
- Obtain the IP addresses for both bridge/routers on either side of the TCP/IP connection.

Procedure

To configure a central site bridge/router to accept any incoming tunnel connection requests, follow these steps:

1 Enable transmission of TCP keepalive packets by entering:

```
SETDefault -TCP CONTrol = KeepAlive
```

TCP keepalive packets notify the bridge/router when the TCP connection has ended. Without TCP keepalive packets, the bridge/router will not detect that the TCP connection is down due to an abnormal situation. This can result in data link switching sessions being kept active even though the corresponding TCP session has ended.

This command will keep switched virtual circuits active even though there is no traffic across the link other than KeepAlive packets.

2 Specify the number of contiguously missed keepalive packets that will bring down the TCP session.

For example, if you want three retries, enter:

```
SETDefault -TCP KeepAliveLimit = 3
```

3 Enable LLC traffic from a port to be tunneled through data link switching by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

This command enables LLC2 traffic on port 1. Repeat this step for each port you are configuring.

4 Assign a unique virtual ring number for the data link switching cloud.

This ring number is used by source routing and some data link switching SSP messages. For example, to configure the virtual tunnel ring, enter:

```
SETDefault -LLC2 TUNnelVRing = 100
```

When using source routing, the internetwork becomes a virtual ring. If your end systems are using token ring source routing, the bridge/router and the IP tunnel appear to the end systems as a source route bridge with a token ring network attached to the other side of the bridge/router. The default ring number of this virtual ring is decimal 92.

This virtual ring number must match on all peer bridge/routers used for data link switching and must be unique within the token ring network. It also will minimize the risk of topology loops.

5 Configure an IP address to connect traffic to and from the router.

The address must be one that has been defined in the router, and will be the only address used for data link switching. To map the specified DLSw tunnel to the local IP address of the central site bridge/router, enter:

```
SETDefault -DLSw Interface = 129.213.1.1
```

All Internet addresses for connected bridge/routers must be known in the local bridge/router's routing table, either dynamically through RIP or OSPF, or statically configured in the IP routing tables.

6 Set the central site bridge/router to accept all DLSw connection requests (including requests from bridge/routers that are not configured as data link tunnel peers) by entering:

```
SETDefault -DLSw MOde = NonSecure
```

The difference between this host configuration and the example configuration for SNA shown in Figure 24-1 is that the host-located data link switch does not need to be configured with the IP address of its partners.

Setting Up DLSw Security Access Filters

You can configure data link switching with additional security beyond what is defined with DLSw peers and known IP addresses. With the -DLSw AccessAct parameter, you can configure the media address you are permitting access to for SNA traffic, or for NetBIOS traffic you can configure specific NetBIOS names of devices you are permitting access to.

Setting Up Filters for SNA Traffic

The following examples of setting up security access for SNA traffic refer to Figure 24-1. Examples 1 and 2 configure bridge/router B for security access.

Example 1 If you want to prevent PC1 from accessing the host, at the bridge/router on network B, enter:

```
SETDefault -DLSw AccessAct = RemoteSnaDiscard
ADD !1 -DLSw SnaRemAccess 00608C26C1B5 ffffffffffff 10005A265BED
  ffffffffffff
```

If you want to prevent PC1 from accessing any remote system at the bridge/router on network B, enter:

```
SETDefault -DLSw AccessAct = RemoteSnaDiscard
ADD !1 -DLSw SnaRemAccess 00608C26C1B5 ffffffffffff 000000000000
  000000000000
```

Example 2 If you want to allow only PC1 access to the host, but want to restrict access to all other systems at the bridge/router on network B, enter:

```
SETDefault -DLSw AccessAct = RemoteSnaForward
ADD !1 -DLSw SnaRemAccess 00608C26C1B5 ffffffffffff 10005A265BED
  ffffffffffff
```

Examples 3 and 4 configure bridge/router A for security access.

Example 3 If you want to prevent PC1 from accessing the host, at bridge/router A, enter:

```
SETDefault -DLSw AccessAct = LocalSnaDiscard
ADD !1 -DLSw SnaLocalAccess 00608C26C1B5 ffffffffffff
  10005A265BED ffffffffffff
```

If you want to prevent PC1 from accessing any system attached to bridge/router B, at bridge/router A, enter:

```
SETDefault -DLSw AccessAct = LocalSnaDiscard
ADD !1 -DLSw SnaLocalAccess 00608C26C1B5 ffffffffffff
 000000000000 000000000000
```

Example 4 If you want to allow only PC1 to access the host, but restrict access for all other local systems, at bridge/router A, enter:

```
SETDefault -DLSw AccessAct = LocalSnaForward
ADD !1 -DLSw SnaLocalAccess 00608C26C1B5 ffffffffffff
 10005A265BED ffffffffffff
```

Setting Up Filters for NetBIOS Traffic

The following examples of setting up security access for NetBIOS traffic refer to Figure 24-2. Examples 1 and 2 configure bridge/router B for security access.

Example 1 If you want to prevent PC001 from accessing the LAN server LS0001, at the bridge/router on network B, enter:

```
SETDefault -DLSw AccessAct = RemoteNBDiscard
ADD !1 -DLSw NBRemAccess PC0001 LS001
```

Example 2 If you want to allow only PC0001 access to LAN server LS0001, but want to restrict access to all other systems, at the bridge/router on network B, enter:

```
SETDefault -DLSw AccessAct = RemoteNBForward
ADD !1 -DLSw NBRemAccess PC0001 LS0001
```

Examples 3 and 4 configure bridge/router A for security access.

Example 3 If you want to prevent PC0001 from accessing LS0001, at bridge/router A, enter:

```
SETDefault -DLSw AccessAct = LocalNBDiscard
ADD !1 -DLSw NBLocalAccess PC0001 LS0001
```

Example 4 If you want to allow only PC0001 to access LS0001, but restrict access for all other local systems, at bridge/router A, enter:

```
SETDefault -DLSw AccessAct = LocalNBForward
ADD !1 -DLSw NBLocalAccess PC0001 LS0001
```

Disabling Data Link Switched Connections

You can disable tunneled data link switch peer connections for a specific peer by tunneling to and from an internetwork, or disabling all tunneling on the local bridge/router.

To disable tunneling from a switch to a peer network, enter:

```
SETDefault !1 -DLSw PEer = 129.213.1.2 Disable
```

This command disables a connection to a peer data link switch.

To disable all tunneling on the bridge/router, enter:

```
SETDefault -DLSw Interface = 0.0.0.0
```

Configuring Statically Defined Media Addresses

If your installation requires multiple DLSw tunnels, you can configure your data link switch connections to use statically defined media addresses. For example, to configure bridge/router A in the SNA example shown in Figure 24-1 with the host media address, enter the following command using noncanonical format for the address:

```
ADD !1 -DLSw PeerMacAdd 10005A265BED
```

Explorer type frames are then sent to the one predefined DLSw peer address.

Configuring Statically Defined NetBIOS Names

If your installation has routers with multiple DLSw tunnels, you can configure your data link switch connections to use statically defined NetBIOS names. For example, to configure bridge/router A in the NetBIOS example shown in Figure 24-2 with the host name, enter:

```
ADD !1 -DLSw PeerNBName LANSERVER1
```

This setting ensures that Name Query frames are not broadcast to all DLSw peer addresses, but are sent only to the predefined DLSw peer.

The section describes how to increase performance and reduce NetBIOS broadcasts.

The 3Com DLSw router at the receiving end of a NetBIOS broadcast sends only one NetBIOS broadcast across the data link switch. The remote DLSw Peer router receives the NetBIOS broadcast and resends this same frame as many times as are defined by the NBBdcastResend parameter at the configured time interval.

If you want to change the values for NetBIOS broadcasts, enter:

```
SETDefault -DLSw NBBcastResend = 5
SETDefault -DLSw NBBcastTimeout = 2
```

NBBcastResend and NBBcastTimeout are independent of each other. Setting one parameter does not effect the other parameter.

Prioritizing DLSw Traffic

This section describes how to assign priorities and allocate bandwidth percentage to traffic from individual workstations, allowing you to give higher priority to mission-critical applications. The address of workstations and host computers in this section are used for example purposes only. Be sure to use the correct addresses for your network.

If the physical port that DLSw is using is being used by another protocol, you also may want to configure data prioritization. For information about how to configure data prioritization, refer to Chapter 41.

How Prioritization and Bandwidth Allocation Work

DLSw allows you to allocate bandwidth to traffic coming from individual workstations on network segments directly attached to a NETBuilder II bridge/router. In addition, DLSw allows you to assign priority to traffic coming from workstations. By assigning priority, you specify the order in which packets from workstations are placed on the link between NETBuilder and WAN services. This effects traffic delays but not traffic throughput. By allocating bandwidth, you specify how much link bandwidth the packets receive.

Example 1 This example illustrates how prioritization and bandwidth allocation work together.

Workstation X is set to High priority and 20% bandwidth. Workstation Y is set to Low priority and 80% bandwidth. Both workstations are sending many packets to the same tunnel. Of every ten packets the tunnel sends, the first two are from workstation X and the last eight are from workstation Y.

Example 2 Figure 24-4 shows two NETBuilder II bridge/routers, router 1 and router 2, as DLSw peers connected by a Frame Relay circuit.

To use the prioritization feature with this network, enter the local workstation's MAC address, service access point (SAP), or LU address identifier. Enter the same information for the workstation's remote session partner. The terms local and remote refer to the router from which you are configuring. For example, in Figure 24-4, you are configuring from router 1, and the addresses for its devices are local. The addresses for devices attached to router 2 are remote. In the figure, each letter represents a different MAC address.

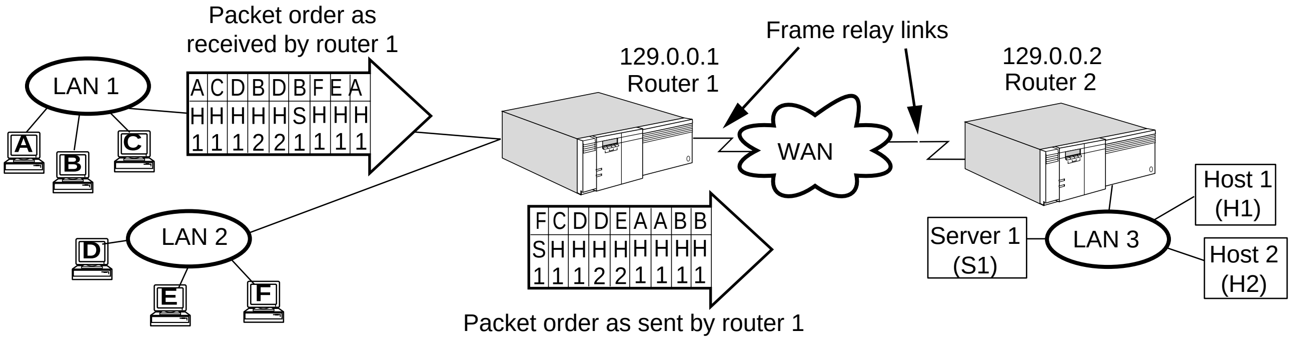


Figure 24-4 DLSw Prioritization and Bandwidth Allocation Example

There are six workstations, A through F, connected to router 1 through LAN 1 and LAN 2. There also are two SNA hosts, host 1 and host 2, and one NetBIOS server, server 1, connected to router 2. The following is the prioritization criteria defined for DLSw traffic going from router 1 to router 2:

- SNA traffic from workstation A to host 1 has a medium priority and 20% of the link bandwidth between router 1 and the Frame Relay service provider.
- SNA traffic from workstation B to host 1 has a high priority and 30% of the link bandwidth between router 1 and the Frame Relay service provider.
- SNA traffic from workstations C, D, E, and F to host 1, host 2, and server 1 has a medium priority and 40% of the link bandwidth between router 1 and the Frame Relay service provider.
- NetBIOS traffic from workstation F to server 1 has a low priority and 10% of the link bandwidth between router 1 and the Frame Relay service provider.
- As Figure 24-4 shows, packets coming from all the workstations get reordered for output on the Frame Relay link based on assigned priorities. For example, router 1 receives some packets from workstation F before it receives some packets from workstation A. However, because A has a high priority and F a low priority, A's packets are sent first because F's priority is lower than the other workstations. F's packets are sent last.

DLSw does not waste tunnel bandwidth. Bandwidth not allocated can be used by any workstations routed through the same tunnel.

Configuring Bandwidth Allocations and Priorities

DLSw allows you to allocate connection bandwidth and assign priorities to traffic from individual workstations. This section describes how to configure the example in Figure 24-4. The addresses of workstations and host computers in this section are not the addresses you are going to use for your network. Be sure to use the correct addresses for your network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the token ring LAN as described in Chapter 5.
- Configure the IP addressing and IP routing protocols on the appropriate ports in Chapter 6.
- Obtain the IP addresses for both bridge/routers on either sides of the TCP/IP connection.
- Configure DLSw for both bridge/routers.
- Set the default DLSw mode to DefaultPRioritized if most of your tunnels are going to be prioritized.
- Add a PEer definition for the remote router and set it to Disable. In Figure 24-4, tunnel ID number 1 is used. When you configure your network, you can use any number between 1 and 256 for tunnel numbers. The peer needs to be defined PRioritized if the default mode is not set to DefaultPRioritized.

Procedure

To configure the example in Figure 24-4, follow these steps:

1 Add a prioritization criterion for workstation A and its session partner, host 1, to the DLSw prioritization database, by entering:

```
ADD !3 -DLSw PRiorityCRiteria 1 20 Medium A SNA H1 SNA
```

Workstation A and host 1 are added to instance ID 3 in the DLSw prioritization database. In addition, workstation A's packets are allocated 20% of the tunnel bandwidth of tunnel ID 1 on router 1 and given a medium priority. The SAP address for workstation A and host 1 is SNA. In this command, the letter's A and H1 represent real MAC addresses; this also applies to the letters in the commands entered in steps 2 and 3.

2 Add a prioritization criterion for workstation B and its session partner, host 2, to the DLSw prioritization database, by entering:

```
ADD !4 -DLSw PRiorityCRiteria 1 30 High B SNA H2 SNA
```

3 Add a prioritization criterion for workstation F and its session partner, server 1, to the DLSw prioritization database, by entering:

```
ADD !5 -DLSw PRiorityCRiteria 1 10 Low F NB S1 NB
```

4 Add a prioritization criteria for all remaining session partners to the DLSw prioritization database, by entering:

```
ADD !6 -DLSw PRiorityCRiteria 1 40 Medium * SNA * SNA
```

5 Enable the connection between the devices attached to router 1 and the data link switch by entering:

`SETDefault -DLSw PEer = 129.0.0.2 Enable`

After you configure session pairs from router 1, you need to configure session pairs from router 2 if you have set the -DLSw MOde parameter to SECure.

Examples of Other Commands

Example 1 To delete an instance ID from the DLSw prioritization database, enter:

`DELete !3 -DLSw PRiorityCRiteria 1`

Instance ID 3 is deleted from tunnel ID 1.

CAUTION: *This command deletes every attribute defined for each device associated with the instance ID and tunnel ID.*

Example 2 To display information in the DLSw prioritization database, enter:

`SHow -DLSw PRiorityCRiteria 1`

All the instance IDs associated with tunnel ID 1 in this example are displayed.

Example 3 To change prioritization criterion number 3 to 60% bandwidth, enter:

`SETDefault !3 -DLSw PRiorityCRiteria 1 60`

If the percentages do not add up to 100%, DLSw normalizes them to 100%.

All devices connected to router 1 that also are associated with instance ID 3 are now allocated 43% of the tunnel bandwidth. DLSw performs the following normalization calculation: 60%/(60% + 30% + 10% + 40%) = 43%.

Example 4 To display prioritized statistics for tunnels on the local router, enter:

`SHow -DLSw PRioritySTATistics`

The following display is an example of these statistics:

```
-------------DLSw PRioritizationSTATistics 192.0.60.10-------------
Tid CurBw     BytesPassed
1   8000      16073
------------------------CriteriaStatistics------------------------
Cid Config% History% BytesPassed  HoldQSize
1   30      0        0                              0
2   20      34       5484                           0
3   30      56       9035                           0
4   20      10       1730                           0
33  0       0        0                              0
```

Example 5 To reset statistics for tunnels on the remote router, enter:

`FLush -DLSw PRioritySTATistics`

Statistics for router 2 tunnels are cleared.

For more information about prioritizing tunnel traffic., refer to Chapter 19 in *Reference for NETBuilder Family Software.*

Prioritizing DLSw Packets

To set the traffic priority of DLSw packets, use:

```
SETDefault -LLC2 TUNnelPRiority = <H | M | L | DEFault>
```

Using this parameter, you set the priority of the packets to high, medium, or low. If this parameter is set to default, the system uses the -IP QueuePriority setting. For more information about the TUNnelPRiority parameter, refer to Chapter 34 in *Reference for NETBuilder Family Software.*

The priority you set using the -LLC2 TUNnelPRiority parameter is different from the priority criteria set using the -DLSw PriorityCriteria parameter. The latter parameter only sets the criteria for prioritizing SNA traffic versus NetBIOS traffic.

When setting prioritization for DLSw packets, UDP explorer frames are automatically set to high priority regardless of the -LLC2 TUNnelPRiority parameter setting. The priority of all other types of UDP packets is set using -LLC2 TUNnelPRiority.

Circuit Balancing

This section describes how to configure DLSw to distribute sessions evenly over multiple DLSw connections and use alternate routes (tunnel paths) for sessions.

If DLSw multicast is being used, circuit balancing is not necessary.

How Circuit Balancing Works

The circuit balancing feature of DLSw allows you to use more than one route between end-stations. When you enable circuit balancing, DLSw considers all available routes between end-stations before assigning a session to a tunnel. DLSw also distributes sessions evenly across all available routes. For example, if there are two routes, and one route has two sessions and the other has three, DLSw assigns the next incoming session to the first route. If a connection fails, DLSw disruptively reroutes end-station and host sessions to an available route (users have to reestablish their sessions with host applications).

Figure 24-5 shows a SuperStack II NETBuilder bridge/router (router 1) with one token ring LAN attached. The LAN also has six workstations attached. router 1 has WAN connections to two NETBuilder II bridge/routers (router 2 and router 3) attached to a front-end processor (FEP) at a host site. Traffic between end-stations (the workstations) and the host travels through DLSw tunnels, and the circuit balancing feature of DLSw is enabled.

When router 1 is configured for circuit balancing, DLSw distributes sessions evenly between Connection 1 and Connection 2. If one of the connections fails, DLSw disruptively reroutes sessions between workstations on the LAN and the host by moving them to the other tunnel.

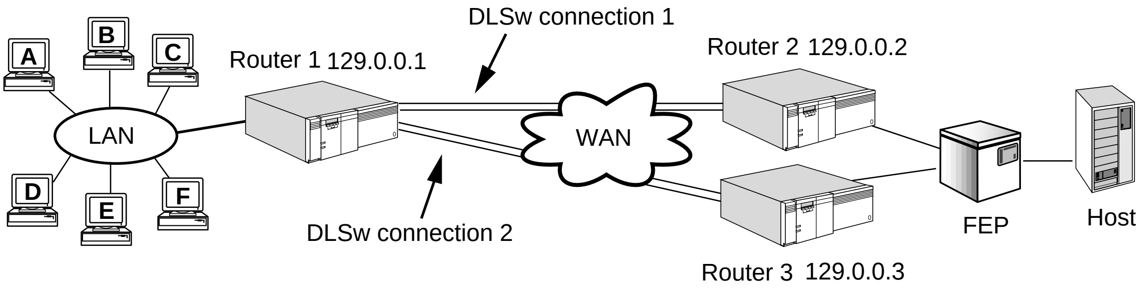


Figure 24-5 Circuit Balancing Example

For circuit balancing to function properly, WAN links must be the same speed. If the WAN links shown in the figure are different speeds (for example, one link is T1 and the other is 64K), then the router with circuit balancing learns the route from the T1 link before the learning the route from the 64K link. All circuits are directed to the DLSw connection on the T1 link instead of being distributed on both the 64K and T1 DLSw connections. Only after alternate routes are in the cache of the circuit balanced router, is the subsequent session establishment balanced (for example, an SNA session to the same MAC address destination is deactivated and then reactivated again).

Configuring Circuit Balancing

This section describes how to configure the example in Figure 24-5.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to each bridge/router with Network Manager privilege.
- Set up the ports and paths of the bridge/routers according to Chapter 1.
- Configure the token ring LAN as described in Chapter 5.
- Configure the IP addressing and IP routing protocols on the appropriate ports as described in Chapter 6.
- Obtain the IP addresses for the three bridge/routers.
- Configure DLSw for the three bridge/routers.
- Set the default DLSw mode to DefaultPRioritized if most of your tunnels are going to be prioritized.
- Add a -DLSw PEer definition for the remote router and one for each host router, and set them to Disable. The peer needs to be defined NoPRioritized if the default mode is not set to DefaultPRioritized.

Procedure

To configure circuit balancing for SNA bridge/router 1, see Figure 24-5 and follow these steps from the router 1 console. Be sure to use the addresses and commands appropriate for your network.

1 Enable circuit balancing for traffic between router 1 and router 2, and between router 1 and router 3 by entering:

```
SETDefault -DLSw CircuitBal = Enable
```

Unless you specify a <cache refresh timeout> value in the SETDefault command, DLSw defaults to 60 minutes. Cache refresh timeout is the interval between each route discovery broadcast.

2 Confirm that circuit balancing is enabled by entering:

```
SHow -DLSw CircuitBal
```

Examples of Other Circuit Balancing Commands

Example 1 To set the interval between each route discovery broadcasting between router 1 and router 2 to 100 minutes, enter:

```
SetDefault -DLSw CircuitBal = Enable 100
```

Example 2 To prevent DLSw from assigning any new circuits (sessions) to Tunnel 1, enter:

```
SETDefault !1 -DLSw PEer = 129.0.0.2 Enable
SET -DLSw CONNections = 129.0.0.2 Quiesce
```

Example 3 To prevent DLSw from sending broadcast or explorer packets on Tunnel 2, enter:

```
SETDefault !2 -DLSw PEer = 129.0.0.3 Enable NoBroadcast
```

Router 1 still accepts and answers explorer packets from router 2 and router 3. The NoBroadcast setting prevents circuits from being initiated from this side.

Configuring Local Switching and Port Groups

You can use local switching and port groups to design DLSw topologies over remote connections for the following situations:

- When you need to translate from one type data link control (such as LLC2) to a different type (such as SDLC), and when you need to concentrate traffic from multiple input ports to one output port locally. This translation is done implicitly and no user configuration is required. Refer to “Using Local Switching to Translate Different DLC Traffic Types” next.
- When you need to convert LAN LLC2 traffic at a branch office to Frame Relay LLC2 traffic (conforming to RFC 1490) that feeds into a remote NETBuilder bridge/router at a regional office, which in turn sends the traffic over DLSw connections to another bridge/router at the central site. This method reduces the number of incoming DLSw connections to the central site. Refer to “Configuring Port Groups for Funneling Many Remote Connections Into Fewer DLSw Connections” on page 24-21.

Local switch port grouping is supported over Frame Relay links only. Also, local switch port grouping cannot be used for BSC traffic.

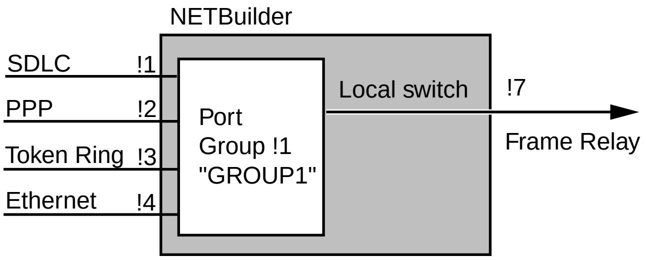
Using Local Switching to Translate Different DLC Traffic Types

You can configure local switching port groups to funnel connections from many LANs into a single bridge/router and in turn funnel these multiple connections through a single data link switch to reach the central site host. With this capability, you can switch incoming traffic of one type to outgoing traffic of the same type or another type on the same bridge/router.

Local switching port groups can be used in specific network topologies where Frame Relay Access Device (FRAD) functionality is desired but is not efficient. For example, port groups can be used in configurations in which IBM traffic is forwarded from an LLC2 or RFC 1490 domain to a Frame Relay circuit that connects to a central site in RFC 1490 format but without the MAC address translation. The central site in this configuration usually hosts so many stations that configuring each remote MAC address into the mapping table is impractical. If you use the FRAD capability, you are required to configure these remote MAC addresses. Local switching port groups enable you to set up such a large network without having to configure hundreds of remote MAC addresses. For more information about FRAD and BAN, refer to Chapter 26.

Port groups configured using this feature are known as explicit port groups. Ports defined as SDLC, FRAD, BAN, or LLC2 (for ports that are LAN encapsulated) are known as implicit port groups. The local switching feature enables you to switch traffic from a port group to other port groups.

Figure 24-6 is an example of configuring port grouping to enable local switching on the bridge/router. In the figure, ports 1 through 4 are incoming ports over a variety of media. These four ports are grouped into port group 1 on the bridge/router; all incoming traffic over the four ports are switched to Frame Relay and are then sent to the Frame Relay WAN over port 7.



ADD !1 -DLSw PortGroup 1 2 3 4 "GROUP1"

Figure 24-6 Port Group Local Switching

You can configure up to eight external port groups on a single bridge/router. Figure 24-7 is an example of multiple port groups on a bridge/router, with each port group forwarding the traffic from its port group to a different host.

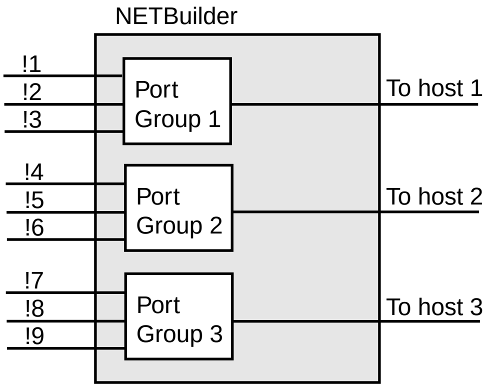


Figure 24-7 Multiple Port Groups on a Bridge/Router

Configuring Port Groups for Funneling Many Remote Connections Into Fewer DLSw Connections

Using port groups, you can reduce the number of Frame Relay connections needed, which enables greater scalability for larger networks. Figure 24-8 is an example of how port grouping can be used to funnel many connections on a large network down to one. In the example, many branch office LANs are connected across a Frame Relay network to a NETBuilder bridge/router at a regional site; the regional site in turn is connected to the host front-end-processor at the central site across another Frame Relay network. By setting up the port groups, you can group the multiple LAN connections from the branch offices and funnel them to the central site across the single data link switch tunnel.

Using this approach, you can avoid having hundreds of DLSw tunnels from each branch office terminating at the central site bridge/router. By combining all SNA traffic from the branch offices into a single Frame Relay DLSw tunnel, you can greatly reduce the number of tunnels at the central site, enabling greater scalability for large networks.

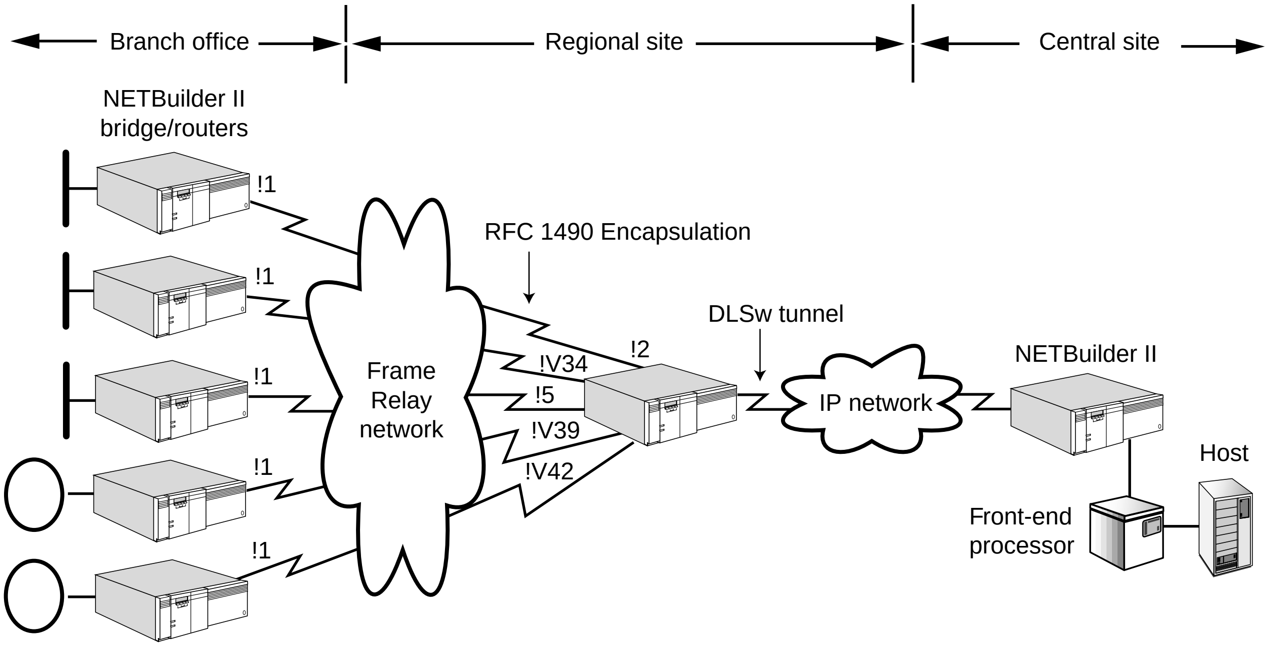


Figure 24-8 DLSw Port Group Topology

You configure the port groups on the NETBuilder bridge/router at each branch office. No special configuration is required at the regional site other than the normal DLSw and port and path configurations.

You cannot have local switch port grouping enabled while bridging is enabled. Before configuring port groups, make sure bridging is disabled.

To configure a port group on the branch office NETBuilder bridge/router in the example, follow these steps on each branch office bridge/router:

3 Enable LLC2 on port 1 by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

4 On the branch office NETBuilder bridge/router, enable source route bridging on port 1 by entering:

```
SETDefault !1 -SR SrcRouBridge = SrcRouBridge
```

5 Define the regional office NETBuilder bridge/router as the DLCI neighbor using:

```
ADD !<port> -BRidge DlciNeighbor = <dlci> (16-991)
```

For example, to define the DLCI neighbor as 20 for port 1, enter:

```
ADD !1 -BRidge DlciNeighbor = 20
```

When you configure the regional office NETBuilder bridge/router, you must also define the DLCI neighbor as 20, so the two bridge/routers can send and receive traffic over Frame Relay.

6 Set the DLCI throughput using:

```
SETDefault !<port> -FR DLCIR = <dlci> <cir>
```

Using this command, define the throughput using the <cir> value based on your service provider's requirements. For example, to define this parameter for port 1 for DLCI number 20 with a <cir> value of 64 (for 64 kbps), enter:

```
SETDefault !1 -FR DLCIR = 20 64
```

When you configure the regional office NETBuilder bridge/router, you must also define this parameter with the same value so the two bridge/routers can send and receive traffic over Frame Relay.

7 Define the port group using:

```
ADD !<port_group_id> -DLSw PortGroup <port> [,...] [“<string>”]
```

For example, to create port group 1 and assign port 1 to it, enter:

```
ADD !1 -DLSw PortGroup 1
```

Using the PortGroup parameter, you can assign up to 16 ports to a port group, and you can also assign a string to the port group. For example, to assign ports 2, 3, 4, and 5 to the port group and assign the string PG1 to it, enter:

```
ADD !1 -DLSw PortGroup 2, 3, 4, 5 “PG1”
```

8 Repeat the previous steps for each branch office bridge/router that will be accessing the same host, assigning the specific ports as necessary.

The port group number only needs to be unique on the local bridge/router. The port group number does not need to match on other bridge/routers.

Table 24-4 lists the commands you need to enter on both the branch office NETBuilder bridge/router and the bridge/router at the regional site for port groups to work.

Table 24-4 Commands to Configure Local Switch Port Groups on Both Bridge/Routers

Commands Entered on the Branch Office Bridge/Routers	Commands Entered on the Regional Office Bridge/Router (entered on the WAN ports to the branch office bridge/routers)
SETDefault !1 -LLC2 CONTrol = Enable	SETDefault !<port> -LLC2 CONTrol = Enable
SETDefault !1 -SR SrcRouBridge = SrcRouBridge	SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
ADD !1 -BRidge DlciNeighbor = 20	ADD !<port> -BRidge DlciNeighbor = 20
SETDefault !1 -FR DLCIR = 20 64	SETDefault !<port> -FR DLCIR = 20 64
ADD !1 -DLSw PortGroup 2, 3, 4, 5 “PG1”	

The following restrictions relate to the use of port groups:

- You cannot use redundant links with port groups.
- SHDLC links are not supported on port groups.

To delete ports in a port group or an entire port group, use:

```
DELete !<port_group_id> -DLSw PortGroup [<port> [,...] | ALL]
```

For example, to delete ports 4 and 5 in port group 1, enter:

```
DELete !1 -DLSw PortGroup 4, 5
```

To delete all ports in port group 1 (and thus delete port group 1), enter:

```
DELete !1 -DLSw PortGroup ALL
```

Network Design Issues for Port Grouping

You can use port grouping to solve the following DLSw network design issues:

- Scaling large DLSw networks
- Scaling large meshed DLSw networks

The following sections describe these issues.

Using Port Groups to Scale Large DLSw Networks. Figure 24-9 is an example in which NETBuilder bridge/routers at six separate branch offices each have a DLSw connection across an IP network into a NETBuilder II bridge/router at a central site. Because there are six DLSw connections, the central site must deal with the overhead and processing for each connection.

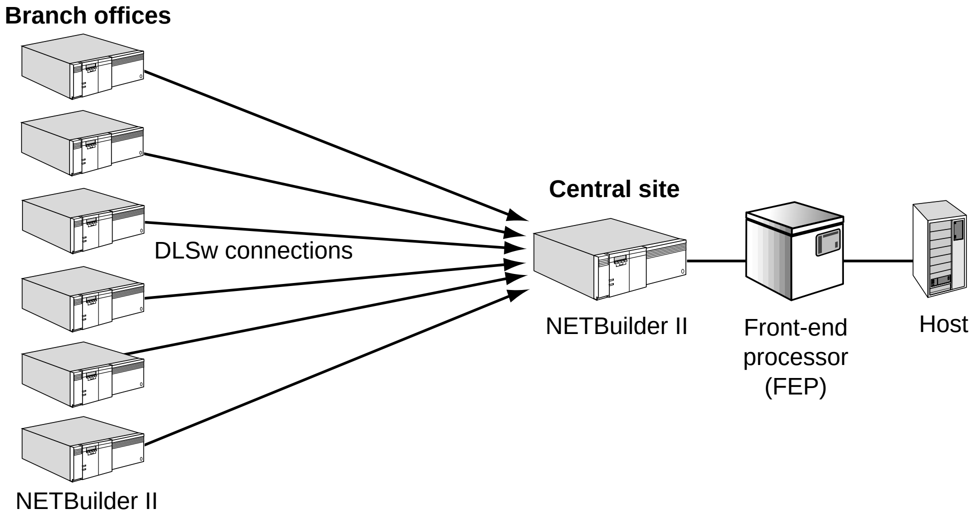


Figure 24-9 DLSw Connections to Remote Offices (Before Port Grouping)

In Figure 24-10, port groups have been configured on NETBuilder bridge/routers at regional offices. Each port group has three remote site branch offices assigned to it, with the three remote connections funneled through a single DLSw connection to the central site. By assigning port groups in this way, you can reduce the number of incoming DLSw connections to the central site from six to two.

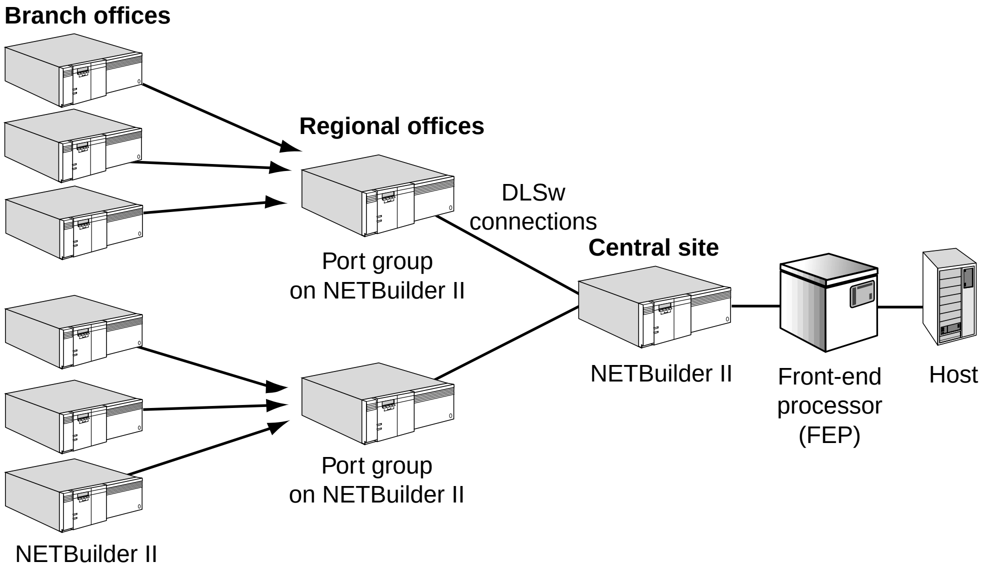


Figure 24-10 DLSw Connections to Remote Offices (After Port Grouping)

Using Port Groups to Scale DLSw Meshed Networks. Figure 24-11 is an example of a DLSw meshed network in which there are bridge/routers at nine remote sites, each configured with DLSw connections so that every site can communicate directly with every other site. Such meshed topologies create additional overhead of large numbers of Frame Relay circuits and TCP connections and create problems with topology update broadcasts.

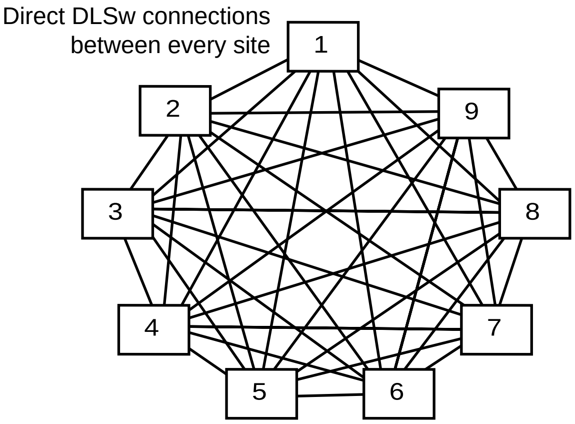


Figure 24-11 DLSw Meshed Network (Before Port Grouping)

Figure 24-12 shows the same meshed network with port groups configured at intermediate regional offices. By configuring port groups on each of the remote sites funneling into three regional offices, each remote site can connect with every other remote site. By assigning port groups in this way, you can reduce the number of DLSw connections from 35 to three. On the regional office bridge/routers, you must either have bridging enabled, or you can configure a port group on each regional office bridge/router for the ports incoming from the remote sites.

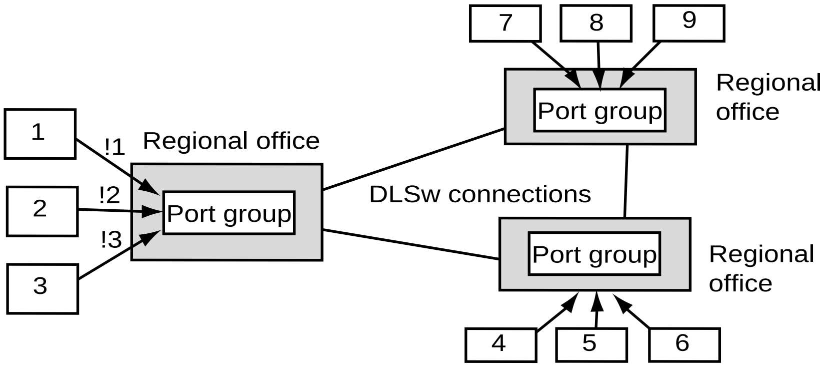


Figure 24-12 DLSw Meshed Network (After Port Grouping)

Configuring DLSw for Dual-TIC Topologies

Host topologies are often designed so that the same MAC address is assigned to multiple token ring interface Cards (TICs) on front-end-processors. This configuration, referred to as a dual-TIC topology, provides greater backup, redundancy, and load balancing across the dual interface cards. The NETBuilder DLSw implementation supports dual-TIC topologies in source-routed environments.

Figure 24-13 is an example in which dual TICs are set up on two token rings. The TICs on the front-end-processors have redundant MAC addresses. For example, the MAC address 10005A265BED is mapped to TIC #A on both 3745A and to TIC #A on 3745B, while MAC address 00608C26C1B5 is mapped to TIC #B on both front end processors.

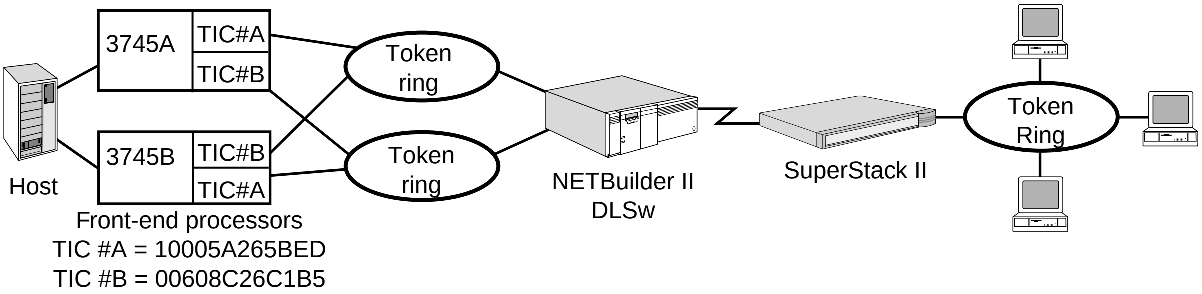


Figure 24-13 DLSw in a Source Route Dual-TIC Topology

In this configuration, the NETBuilder II bridge/router supports the dual-TIC environment. No special configuration is required to support dual-TIC except for the following steps:

- You must configure ring numbers for the token rings accessing the front-end-processor.s
- You must turn off transparent bridging (use source route bridging only)

For more information about source route bridging, refer to Chapter 5.

Converting SNA Alerts to SNMP Traps

This section describes how the SnaAlertsToTraps feature of DLSw converts SNA alerts to SNMP traps so that SNMP managers, such as SunNet Manager, NetView AIX, or HP OpenView, can process them. When SNA devices detect a problem, they can send SNA alerts to a focal point (usually NetView) where they are processed and displayed to an operator. The alerts contain information describing the problem and possible actions to be taken.

How SNA-Alerts-To-Traps Works

DLSw allows you to interconnect devices such as OS/2 workstations and 3174 cluster controllers to SNA hosts using NETBuilder II bridge/routers. The SnaAlertsToTraps feature of DLSw enables SNMP management platforms to manage SNA devices (end-stations) by converting their SNA alerts to SNMP traps and sending the traps to the SNMP manager.

Figure 24-14 shows an end-station and an IBM host connected by a SuperStack II bridge/router (router 1) and a NETBuilder II bridge/router (router 2) over an IP network. The end-station sends SNA alerts to router 1, which passes them to the IBM host, where NetView processes and displays them to an operator. router 1 converts the SNA alerts to SNMP traps and sends the traps to the SNMP manager.

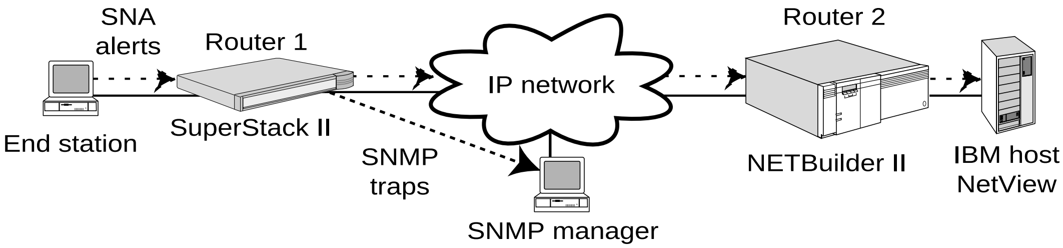


Figure 24-14 SnaAlertsToTraps Example

Configuring SnaAlertsToTraps

To configure the SnaAlertsToTraps feature, follow these steps from the SuperStack II console:

1 Set the trap option for the SNMP Service by entering:

```
SETDefault -SNMP CONTrol = Trap
```

The SnaAlertsToTraps feature does not work unless trap is set using the SNMP Service. For more information about how to configure the NETBuilder II bridge/router so that it can be controlled by an SNMP manager, refer to Chapter 53 and Chapter 55 in Reference for NETBuilder Family Software.

2 Enable the SnaAlertsToTraps feature by entering:

```
SETDefault -DLSw SnaAlertsToTraps = Send
```

When you enable SnaAlertsToTraps, the SuperStack II bridge/router processes SNA alerts for every attached LU and PU.

You can use two other values instead of Send. The SendAlert value encapsulates the entire SNA alert (the Network Management Vector Transport (NMVT)) inside an SNMP trap protocol data unit (PDU), and sends it to the SNMP manager. The Disabled value tells the SuperStack II bridge/router to ignore all SNA alerts.

To verify the current state of the SnaAlertsToTraps feature, enter:

```
Show -DLSw SnaAlertsToTraps
```

A display indicates whether the SnaAlertsToTraps feature is enabled.

For more information about configuring the SnaAlertsToTraps feature, refer to Chapter 19 in *Reference for NETBuilder Family Software*.

How Data Link Switching Works

DLSw supports SNA and NetBIOS in multiprotocol routers. SNA and NetBIOS provide connection-oriented services. SNA and NetBIOS use IEEE 802.2 LLC2 protocol over LANs. DLSw also provides SNA connectivity over WAN links for devices attached by SDLC peripheral links. For conceptual information on how data link switching works for LANs, refer to RFC 1795. The NETBuilder bridge/router family of hardware and software fully implements this standard.

Figure 24-15 shows a typical network configuration using data link switching with SNA and NetBIOS traffic to connect three bridge/routers across an IP internetwork. Each connection is a tunnel, which consists of two TCP ports: one to send data (port #2067) and one to receive data (port #2065).

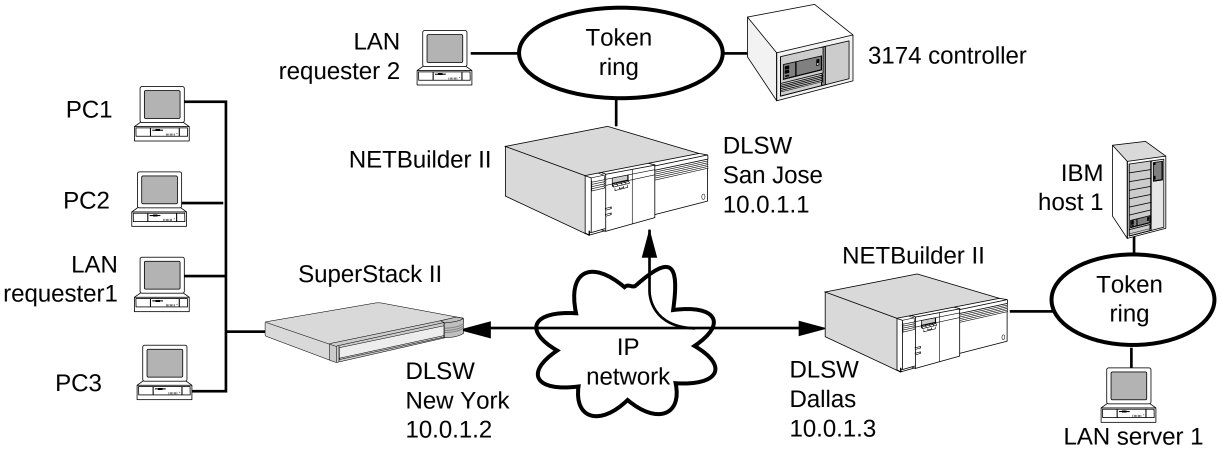


Figure 24-15 Simple Data Link Switching Configuration

Multiple sessions between different ports are multiplexed onto a single tunnel. For instance, if there is a session connecting LAN server 1 and LAN requester 1, and a concurrent session connecting PC1 and host 1, traffic is multiplexed onto a single tunnel between the NETBuilder II bridge/router at Dallas and the SuperStack II NETBuilder bridge/router at New York.

Media Addressing and NetBIOS Name Caching

When the 3Com DLSw router receives an explorer or NetBIOS name type frame, the router first checks the statically defined table for the existence of a predefined route. The router also checks the DLSw caching tables for a match. If a match is found, the frame is forwarded on the static or cached DLSw tunnel. If no match exists, then the frame is forwarded to each DLSw tunnel. When the DLSw router receives a DLSw explorer or NetBIOS name type frame, the router adds the media address or the NetBIOS name to its caching tables.

A cached item is deleted when the DLSw router uses a cached route to forward an explorer frame but fails to get a response. The result is that the first explorer or query frame is sent using the cache tunnel. When that frame fails to get a response, the cached item is deleted and the query is resent on all tunnels.

DLSw Configuration and STP

DLSw is not aware of the Spanning Tree Protocol (STP). Because of this limitation, you must avoid configuring a second data path that can loop SNA and NetBIOS traffic back to an originating router. Do not configure either bridge or tunnel paths as second data paths. Avoid the topology shown in Figure 24-16 because it may duplicate packets and cause failure.

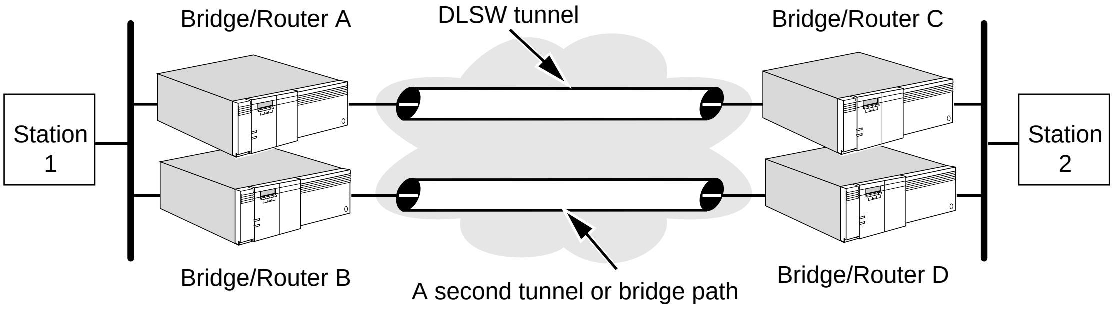


Figure 24-16 Illegal DLSw Tunneling Configuration

Your site may require redundancy in a DLSw environment. If you need DLSw bridge/router redundancy, contact your network supplier for planning.

In token ring topologies, DLSW or LLC2 tunneling can support parallel paths in a source-routed-only environment.

Data Link Switching Terms

Data link switching terminology uses tunneling terms that have specific meanings defined in RFC 1795, and are relevant for the DLSw environment.

data link switching	A method for forwarding SNA and NetBIOS traffic between routers.
initial bandwidth	An option that allows you to define initial tunnel bandwidth.
instance ID	A number that identifies an entry in a table.
peer	A relationship between a local and remote router, usually referring to a remote router with a remote address, which is the peer IP address.
prioritization	An option that allows you to allocate tunnel bandwidth to data traffic coming from devices associated with a specific priority criterion.
Switch-to-Switch Protocol (SSP)	The protocol used between two communicating data link switches.
tunnel	A connection between two routers using two IP addresses, one in each router. Both routers must be using the data link switching Switch-to-Switch Protocol. Multiple tunnels between multiple routers can be configured.
tunnel ID	A local identifier that defines tunnels to peer devices.
tunneling	The encapsulation of SNA and NetBIOS traffic in a TCP/IP packet, using the Data Link Switching Protocol.

25

CONFIGURING MULTICAST DATA LINK SWITCHING FOR NETBIOS AND SNA NETWORKS

This chapter describes how to configure your system to perform multicast data link switching (DLSw). Multicast DLSw allows easier scalability of large DLSw networks while reducing the number of configuration steps required. Multicast DLSw provides an enhancement to the RFC 1795-compliant DLSw described in Chapter 24. Multicast DLSw provides the following enhancements:

- Reduced configuration for data link switches

  With RFC 1795-compliant DLSw, each data link switch in partially meshed or fully meshed networks must be configured for one or more peers so that TCP connections can be established between the DLSw peers. With multicast DLSw, IP multicast addresses are used for exploration, which eliminates the requirement that DLSw peers must be configured.

- Reduced WAN backbone traffic

  With RFC 1795-compliant DLSw, each data link switch sends out broadcast CANUREACH_ex Switch-to-Switch Protocol (SSP) requests on every TCP connection. With multicast DLSw, only one multicast packet is sent out by a data link switch, which reduces WAN backbone traffic.

- Reduced TCP Overhead

  With RFC 1795-compliant DLSw, each data link switch has two TCP connections with each of its peers, whether or not a circuit is established between end systems through the DLSw peers. With multicast DLSw, TCP connections are brought up only if a circuit needs to be established between the data link switches. The TCP connections are brought down when all circuits using the connection have ended.

Before you configure multicast DLSw, MOSPF must be configured. For information on configuring MOSPF, refer to Chapter 9. For information on MOSPF parameters, refer to Chapter 37 in Reference for NETBuilder Family Software.

Configuring Multicast DLSw

This section describes how to configure multicast DLSw for NetBIOS or SNA traffic. Multicast DLSw is useful in the following network topologies:

- Configurations in which stations are communicating using NetBIOS, where logical meshed network connectivity is desirable.
- SNA networks in which TCP connections between the PU2 client and the host server is always required. Multicast DLSw is useful in demand-based situations where the sessions between the clients and the host do not need to be up all the time. As a result, TCP connections do not need to be kept up all the time, saving processing overhead.

For configuration procedures for NetBIOS, refer to the next section. For configuration procedures for SNA client and server environments, refer to "Configuring Multicast DLSw for SNA Client and Server Environments" on page 25-3.

DVMRP is not supported with DLSw multicast.

Configuring DLSw Multicast for NetBIOS Mesh Environments

This section describes how to configure multicast DLSw for NetBIOS meshed environments. In this configuration, DLSw bridge/routers can use the default multicast address for both transmit and receive traffic.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the IP addressing and IP routing protocols on the appropriate ports according to Chapter 6.
- Configure MOSPF according to the procedures in Chapter 9.
- Obtain the IP addresses for bridge/routers on each side of the TCP/IP connection.
- Configure the DLSw peer configuration according to the procedures described in "Configuring for NetBIOS" on page 24-4.

Figure 25-1 is an example of a meshed NetBIOS environment. In the figure, four DLSw bridge/routers are participating in the multicast environment.

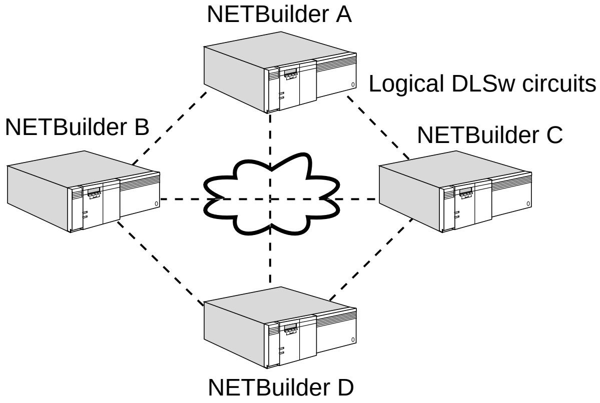


Figure 25-1 DLSw Multicast Example (Meshed NetBIOS Environment)

To configure multicast DLSw as shown in the figure, follow these steps on each bridge/router:

1 Set the DLSw mode on the bridge/router to multicast by entering:

```
SETDefault -DLSw MOde = Multicast
```

2 Enable NetBIOS for DLSw by entering:

```
SETDefault -DLSw CONTrol = EnableNetBios
```

3 Enable IP routing by entering:

```
SETDefault -IP CONTrol = Route
```

4 Enable multicast IP routing by entering:

```
SETDefault -MIP CONTrol = Enable
```

5 Enable OSPF on the DLSw port using:

```
SETDefault !<port> -OSPF CONTrol = Enable
```

6 Enable multicast OSPF on the DLSw WAN port using:

```
SETDefault !<port> -MOSPF CONTrol = Enable
```

After you follow these steps on each bridge/router, the routers send multicast requests onto the meshed network, and each router can reach every other router without configuring static DLSw peers.

Configuring Multicast DLSw for SNA Client and Server Environments

This section describes how to configure multicast DLSw for SNA client and server environments. In these configurations, one data link switch router is connected to an SNA host and a second data link switch router is connected to clients (PU2). You need to configure the appropriate multicast DLSw addresses on the client and the server routers.

Because you need to configure the multicast DLSw address on both sides, the benefit of using DLSw multicast for SNA client and server environments is limited. The primary benefit of using DLSw multicast instead of RFC 1795-compliant DLSw is that the TCP connections come up dynamically as needed and go down when the circuit becomes idle.

Prerequisites

Before beginning this procedure, complete these tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the IP addressing and IP routing protocols on the appropriate ports according to Chapter 6.
- Configure MOSPF according to the procedures in Chapter 9.
- Obtain the IP addresses for bridge/routers on each side of the TCP/IP connection.
- Configure the DLSw peer configuration according to the procedures described in “Configuring for SNA” on page 24-1.

To configure multicast DLSw on the SNA client, follow these steps:

1 Set the DLSw mode on the bridge/router to multicast by entering:

```
SETDefault -DLSw MOde = Multicast
```

2 Delete the default multicast address by entering:

```
DELete -DLSw MulticastAddr DEFault
```

This command deletes the default multicast address 224.0.10.0, which allows you to configure the multicast address in the next step.

You can restore the default multicast address by entering the ADD -DLSw MulticastAddr command and specifying DEFault.

The default multicast address is configured as TxRx, which is acceptable for fully meshed configurations, but is not suitable for client-server configurations.

3 Define the Class D multicast address that the client bridge/router will *receive* SNA traffic on using:

```
ADD -DLSw MulticastAddr <IP multicast address> SNA Rx
```

When entering the IP multicast address, you can enter any Class D address, from 224.0.0.0 to 239.255.255.255. The range of valid multicast addresses for DLSw multicast only is from 224.0.10.0 to 224.0.10.255.

For example, to add the IP multicast address 224.0.10.100 to receive traffic on, enter:

```
ADD -DLSw MulticastAddr 224.0.10.100 SNA Rx
```

4 Define the Class D multicast address that the client bridge/router will *transmit* SNA traffic on using:

```
ADD -DLSw MulticastAddr <IP multicast address> SNA Tx
```

For example, to add the IP multicast address 224.0.10.200 to receive traffic on, enter:

```
ADD -DLSw MulticastAddr 224.0.10.200 SNA Tx
```

5 Enable IP routing by entering:

```
SETDefault -IP CONTrol = Route
```

6 Enable multicast IP routing by entering:

```
SETDefault -MIP CONTrol = Enable
```

7 Enable OSPF on the port using:

```
SETDefault !<port> -OSPF CONTrol = Enable
```

8 Enable multicast OSPF on the port using:

```
SETDefault !<port> -MOSPF CONTrol = Enable
```

To configure multicast DLSw on the SNA server, follow the steps in the previous procedure except for steps 3 and 4. In steps 3 and 4 configure the multicast addresses used to send and receive traffic, but reverse the addresses configured for those steps. On the server, configure the sending and receiving multicast addresses by entering:

```
ADD -DLSw MulticastAddr 224.0.10.200 SNA Rx
ADD -DLSw MulticastAddr 224.0.10.100 SNA Tx
```

Figure 25-2 is an example of an SNA configuration in which multicast DLSw is used. Table 25-1 lists the commands to configure on each DLSw client and server bridge/router to allow multicast DLSw to work.

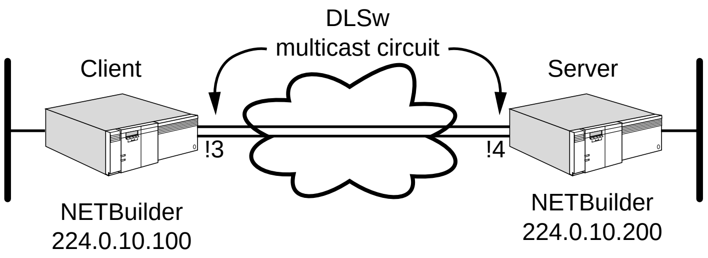


Figure 25-2 DLSw Multicast Example (SNA Configuration)

Table 25-1 Commands to Configure Multicast DLSw for SNA

Commands Entered on Client Bridge/Router	Commands Entered on Server Bridge/Router
SETDefault -DLSw MOde = Multicast	SETDefault -DLSw MOde = Multicast
DELete -DLSw MulticastAddr DEFault	DELete -DLSw MulticastAddr DEFault
ADD -DLSw MulticastAddr 224.0.10.100 SNA Rx	ADD -DLSw MulticastAddr 224.0.10.200 SNA Rx
ADD -DLSw MulticastAddr 224.0.10.200 SNA Tx	ADD -DLSw MulticastAddr 224.0.10.100 SNA Tx
SETDefault -IP CONTrol = Route	SETDefault -IP CONTrol = Route
SETDefault -MIP CONTrol = Enable	SETDefault -MIP CONTrol = Enable
SETDefault !3 -OSPF CONTrol = Enable	SETDefault !4 -OSPF CONTrol = Enable
SETDefault !3 -MOSPF CONTrol = Enable	SETDefault !4 -MOSPF CONTrol = Enable

Customizing the DLSw Multicast Configuration

This section describes how to customize the multicast DLSw configuration.

Tuning DLSw Multicast Parameters

You can tune the retry interval and retry count for the number of times that the SSP frames sent on multicast are retried. The default retry interval is 3 seconds for SNA and 1 second for NetBIOS, and the default retry count is 0 (no retries). To change the retry interval and retry count, use:

```
SETDefault -DLSw McastRetry = <SNA | NetBios> <retry interval
 (1-5)> <retry count (0-5)>
```

You must also specify whether the change is for SNA or NetBIOS traffic.

You can also specify the number of minutes that a TCP connection between multicast DLSw peers will stay up without any circuit using the connection. To change the TCP idle time, use:

```
SETDefault -DLSw McastTcpIdle = <timer duration (1-255)>
```

The default is 3 minutes.

Restoring the Default Multicast Address

If you want to restore the default multicast address (224.0.10.0) on the bridge/router after previously configuring an address for multicasting purposes, enter:

```
ADD -DLSw MulticastAddr DEFault
```

When you specify DEFault, the traffic type defaults to ALL and the usage defaults to TxRx.

Disabling DLSw Multicast

To disable DLSw multicast on the bridge/router, enter:

```
SETDefault -DLSw MOde = NoMulticast
```

The bridge/router stops sending out multicasts to DLSw stations on the network.

26 CONFIGURING FRAME RELAY ACCESS DEVICE SUPPORT FOR SNA

This chapter describes how to configure the bridge/router as a Frame Relay Access Device (FRAD) node to provide Frame Relay access support for Systems Network Architecture (SNA). The FRAD functionality implements Logical Link Control, type 2 (LLC2) encapsulation over Frame Relay, and uses data packet encapsulation methods based on the RFC 1490 frame format. In SNA environments, FRAD provides similar functionality to the IBM Frame Relay Boundary Network Node (BNN) and Boundary Access Node (BAN).

For more information about how FRAD works, refer to "How the Frame Relay Access Device Works" on page 26-7.

Configuring the NETBuilder as a FRAD Node

This section describes how to configure the bridge/router as a FRAD node for both BAN and BNN configurations. Table 26-1 shows how BAN and BNN support FRAD nodes.

Table 26-1 BAN and BNN Capabilities

BAN	BNN
RFC 1490 bridged token ring format.	RFC 1490 routed SNA format.
Static addressing not needed because the end-station MAC address is provided in every frame.	Static addressing using end-station MAC and SAP address.
Load balancing.	No load balancing.
Supports LAN, SDLC, and APPN.	LAN, SDLC, and APPN.

Configuring FRAD for LAN-Attached End Stations

You can configure FRAD for LAN-attached end stations for either Boundary Access Node (BAN) or for Boundary Network Node (BNN). This section is divided into two procedures, one for BAN-attached end stations and one for LAN-attached end stations using BNN.

Configuring the FRAD Node for a BAN-Attached End Station

Figure 26-1 is an example of a NETBuilder II bridge/router acting as a FRAD node with a BAN-attached end station. The FRAD provides Frame Relay access to the remote host front-end processor (FEP).

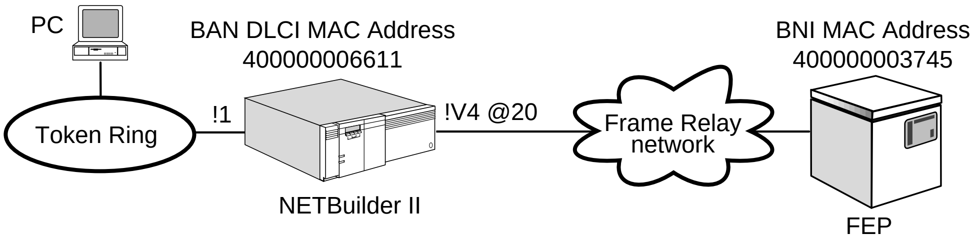


Figure 26-1 FRAD Node Configuration (BAN-attached End Station)

Prerequisites

Before beginning this procedure, complete the following tasks:

- Set up the ports and paths of your bridge/router according to the procedures described in Chapter 1.
- To control which BAN ports are active at a given time, you must configure the ports accessing Frame Relay as virtual ports.
- Configure the Frame Relay interface. For more information on configuring Frame Relay, refer to Chapter 37.

To configure the NETBuilder as a FRAD node for a BAN-attached end station, follow these steps:

1 Configure the physical Frame Relay port by entering:

```
SETDefault !4 -PATH BAud = 56
SETDefault !4 -PORT OWNer = FrameRelay
```

2 Add the virtual port by entering:

```
ADD !v1 -PORT VirtualPort 4@40
```

3 Configure the address mapping for Frame Relay connections to the FEP using:

```
ADD !<port> -DLSw BoundaryAccessNode <ban dlci mac addr> [<bni mac addr>]
```

With this syntax you map the source MAC to the FEP MAC and assign the boundary node indicator (BNI) MAC address. For example:

```
ADD !v1 -DLSw BAN 400000006611 400000003745
```

For more information about the mapping rules that apply to the FradMap parameter, refer to "How the Frame Relay Access Device Works" on page 26-7. For more information about the FradMap parameter, refer to Chapter 16 in *Reference for NETBuilder Family Software*.

4 Enable LLC2 control on the Frame Relay port by entering:

```
SETDefault !v1 -LLC2 CONTrol = Enabled
```

5 Enable LLC2 control on the local port to enable host-initiated activation by entering:

```
SETDefault !1 -LLC2 CONTrol = Enabled
```

6 For the SuperStack II NETBuilder Token Ring platforms or source-routing-only environment, you need to configure the end station support for LLC2 and source routing for the LAN port.

Configuring the FRAD Node for a LAN-Attached End Station Using BNN

Figure 26-2 is an example of a NETBuilder II bridge/router acting as a FRAD node with a LAN-attached end station. The FRAD provides Frame Relay access to the remote host FEP.

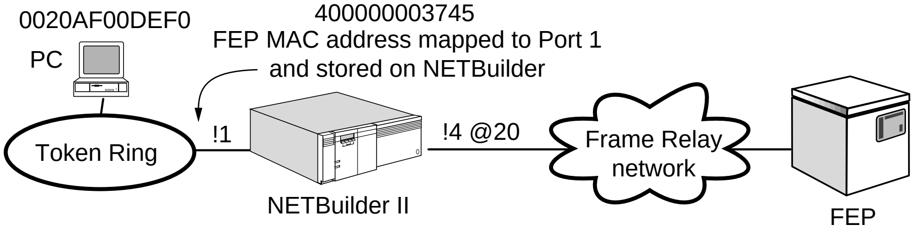


Figure 26-2 FRAD Node Configuration (LAN-attached End Station using BNN)

Prerequisites

Before beginning this procedure, complete the following tasks:

- Set up the ports and paths of your bridge/router according to the procedures described in Chapter 1.
- Configure the Frame Relay interface. For more information on configuring Frame Relay, refer to Chapter 37.

To configure the NETBuilder as a FRAD node for a LAN-attached end station using BNN, follow these steps:

1 Configure the physical Frame Relay port by entering:

```
SETDefault !4 -PATH BAud = 56
SETDefault !4 -PORT OWNer = FrameRelay
```

The baud rate for the path should match the speed of the Frame Relay line.

2 For LAN-attached end stations using BNN, disable bridging on the Frame Relay port by entering:

```
SETDefault !4 -BR TransparentBRidge = NoTransparentBridge
SETDefault !4 -SR SrcRouBridge= NoSrcRouBridge
```

3 To configure the address mapping for the Frame Relay connections to the FEP use:

```
ADD !<port> -DLSw FradMap <src mac> <src sap> <fep mac> <fep sap>
  <DLCI> <code point>
```

With this syntax, you map the source MAC and SAP to the FEP MAC and SAP and assign the DLCI and the code point. For example:

```
ADD !4 -DLSw FradMap 0020AF00DEF0 4 400000003745 4 20 82
```

For more information about the mapping rules that apply to the FradMap parameter, refer to “How the Frame Relay Access Device Works” on page 26-7. For more information about the FradMap parameter, refer to Chapter 16 in *Reference for NETBuilder Family Software*.

4 Enable LLC2 control on the Frame Relay port by entering:

```
SETDefault !4 -LLC2 CONTrol = Enabled
```

5 To enable LLC2 control on the local port to enable host-initiated activation enter

```
SETDefault !1 -LLC2 CONTrol = Enabled
```

6 For the SuperStack II NETBuilder Token Ring bridge/routers or in a source-routing-only environment, you need to configure the end station support for LLC2 and source routing for the LAN port

Configuring FRAD for SDLC-Attached End Stations

You can configure FRAD for SDLC-attached end stations for either Boundary Access Node (BAN) or for Boundary Network Node (BNN). This section is divided into two procedures, one for BAN and one for BNN.

Configuring the FRAD Node for an SDLC-Attached End Station Using BAN

Figure 26-3 is an example of a NETBuilder II bridge/router acting as a FRAD node with an SDLC-attached end station. The addressing shown is for a BNN configuration.

For a BAN configuration, the FEP address 100040609D88 is seen internally at the NETBuilder II (FRAD node) and is mapped to the BAN BNI MAC address 4FFF00000000. The FRAD provides Frame Relay access to the remote host FEP.

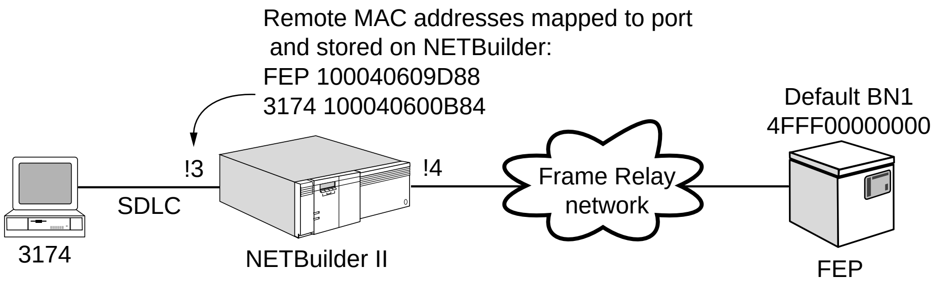


Figure 26-3 FRAD Node Configuration (SDLC-attached End Station) for BAN

Prerequisites

Before beginning this procedure, complete the following tasks:

- Set up the ports and paths of your bridge/router according to the procedures described in Chapter 1.
- Configure the Frame Relay interface. For more information on configuring Frame Relay, refer to Chapter 37.
- Configure the SDLC conversion for the SDLC-attached end station. This configuration must be done before configuring the NETBuilder II as a FRAD node. For more information about configuring SDLC conversion, refer to Chapter 23.

Procedure

To configure the NETBuilder II bridge/router as a FRAD node for an SDLC-attached end station for BAN, follow these steps:

1 To configure the SDLC physical port and path attributes enter:

```
SETDefault !3 -PORT OWNer = SDLC
SETDefault !3 -PATH DUplex = Full
SETDefault !3 -PATH ENCoding = NRZI
```

2 To configure the SDLC logical port and path attributes enter:

```
ADD !3 -SDLC PortCU PU31741
SETDefault !3 -SDLC PDatMode = Full
SETDefault !3 -SDLC PROle = Primary
SETDefault !3 -SDLC PCONTrol = Enabled
```

3 To configure the SDLC CU (adjacent link station) attributes enter.

```
SETDefault !PU31741 -SDLC CUAddr = C1
SETDefault !PU31741 -SDLC CULocalMac = 100040600B84
SETDefault !PU31741 -SDLC CURemoteMac = 100040609D88
SETDefault !PU31741 -SDLC CULocalSap = 4
SETDefault !PU31741 -SDLC CURemoteSap = 4
SETDefault !PU31741 -SDLC CUCONTrol = Enabled
```

4 To configure address mapping for the Frame Relay connections to the FEP use.

```
ADD !<port> -DLSw BoundaryAccessNode <ban dlci mac addr> [<bni mac
  addr>]
```

With this syntax, you map the source MAC to the FEP MAC and assign the BNI MAC address. For example, assuming that the DLCI is 40:

```
ADD !v4 -PORT VirtualPort 4@40
ADD !v1 -DLSw BAN 100040600D88
```

If the BNI DLCI address is different, you must add the BNI DLCI address to the command. For example:

```
ADD !v1 -DLSw BAN 100040600D88 4FFF00037451
```

For more information about the mapping rules that apply to the FradMap parameter, refer to “How the Frame Relay Access Device Works” on page 26-7. For more information about the FradMap parameter, refer to Chapter 16 in *Reference for NETBuilder Family Software*.

5. Enable LLC2 control on the Frame Relay port by entering

```
SETDefault !4 -LLC2 CONTrol = Enabled
```

6. Enable LLC2 control on the local port to enable host-initiated activation by entering:

```
SETDefault !1 -LLC2 CONTrol = Enabled
```

Configuring the FRAD Node for an SDLC-Attached End Station Using BNN

Figure 26-4 shows an example of a NETBuilder II bridge/router acting as a FRAD node with an SDLC-attached end station for BNN.

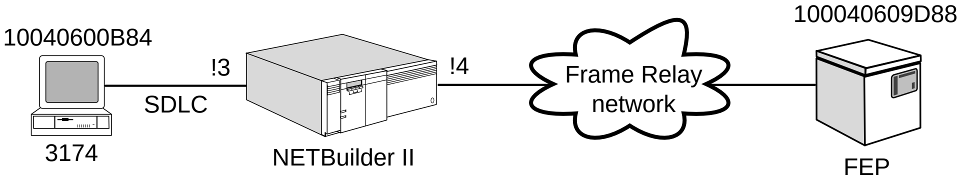


Figure 26-4 FRAD Node Configuration (SDLC-attached End Station) for BNN

Prerequisites

Before beginning this procedure, complete the following tasks:

- Set up the ports and paths of your bridge/router according to the procedures described in Chapter 1.
- Configure the Frame Relay interface. For more information on configuring Frame Relay, refer to Chapter 37.
- Configure the SDLC conversion for the SDLC-attached end station. This configuration must be done before configuring the NETBuilder II bridge/router as a FRAD node. For more information about configuring SDLC conversion, refer to Chapter 23.

Procedure

To configure the NETBuilder II bridge/router as a FRAD node for an SDLC-attached end station for BNN, follow these steps:

1. Configure the physical Frame Relay port by entering:

```
SETDefault !4 -PATH BAud = 56
SETDefault !4 -PORT OWNer = FrameRelay
```

The baud rate for the path should match the speed of the Frame Relay line.

2. Configure the SDLC physical port and path attributes by entering:

```
SETDefault !3 -PORT OWNer = SDLC
SETDefault !3 -PATH DUplex = Full
SETDefault !3 -PATH ENCoding = NRZI
```

3 Configure the SDLC logical port and path attributes by entering:

```
ADD !3 -SDLC PortCU PU31741
SETDefault !3 -SDLC PDatMode = Full
SETDefault !3 -SDLC PROle = Primary
SETDefault !3 -SDLC PCONTrol = Enabled
```

4 Configure the SDLC CU (adjacent link station) attributes by entering:

```
SETDefault !PU31741 -SDLC CUAddr = C1
SETDefault !PU31741 -SDLC CULocalMac = 100040600B84
SETDefault !PU31741 -SDLC CURemoteMac = 100040609D88
SETDefault !PU31741 -SDLC CULocalSap = 4
SETDefault !PU31741 -SDLC CURemoteSap = 4
SETDefault !PU31741 -SDLC CUCONTrol = Enabled
```

5 For BNN-attached end stations, disable bridging on the Frame Relay port by entering:

```
SETDefault !4 -BR TransparentBRidge = NoTransparentBRidge
SETDefault !4 -SR SrcRouBridge = NoSrcRouBridge
```

6 Configure the address mapping for Frame Relay connections to the FEP using:

```
ADD !<port> -DLSw FradMap <src mac> <src sap> <fep mac> <fep sap>
  <DLCI> <code point>
```

With this syntax, you map the source MAC and SAP to the FEP MAC and SAP and assign the DLCI and the code point. For example:

```
ADD !4 -DLSw FradMap 100040600B84 4 100040609D88 4 20 82
```

For more information about the mapping rules that apply to the FradMap parameter, refer to "How the Frame Relay Access Device Works" on page 26-7. For more information about the FradMap parameter, refer to Chapter 16 in *Reference for NETBuilder Family Software*.

7 Enable LLC2 control on the Frame Relay port by entering:

```
SETDefault !4 -LLC2 CONTrol = Enabled
```

8 Enable the LLC2 control on the local port to enable host-initiated activation by entering:

```
SETDefault !1 -LLC2 CONTrol = Enabled
```

Deleting Frame Relay Address Mappings

To delete a Frame Relay address mapping for BAN, use:

```
DELete !<vport> -DLSW BoundaryAccessNode <bnn dlci mac addr>
```

For example, to delete an address mapping on virtual port 1 with a FEP MAC address of 400000006611, enter:

```
DELete !v2 -DLSW BoundaryAccessNode 400000006611
```

To delete a Frame Relay address mapping for BNN use:

```
DELete !<port> -DLSw FradMap <src mac> <src sap> <fep mac> <DLCI>
  <code point>
```

For example, to delete an address mapping on port 2 with a source MAC address of 00608C2C61B5, a source SAP of 04, and a FEP MAC address of 40005A65BED, enter:

```
DELete !2 -DLSw FradMap 00608C2C61B5 04 40005A65BED
```

Displaying Frame Relay Address Mappings

To display Frame Relay address mappings, use:

```
SHow [!<port>|!*] -DLSw FradMap
```

How the Frame Relay Access Device Works

The 3Com FRAD implements LLC2 encapsulation over Frame Relay based on the RFC 1490 frame format. RFC 1490 describes how FRAD carries LLC2 frames over Frame Relay.

The specific values of SNA and NetBIOS are documented in the Frame Relay Forum *FRF.3 Multiprotocol Encapsulation Implementation Agreement* document, as well as in the ANSI *T1.617 Annex F.* These values are referred to as *code points* in these documents. The following are the code points currently defined for SNA and NetBIOS:

0x81- SNA Subarea (FID4)

0x82 - SNA Peripheral (FID2)

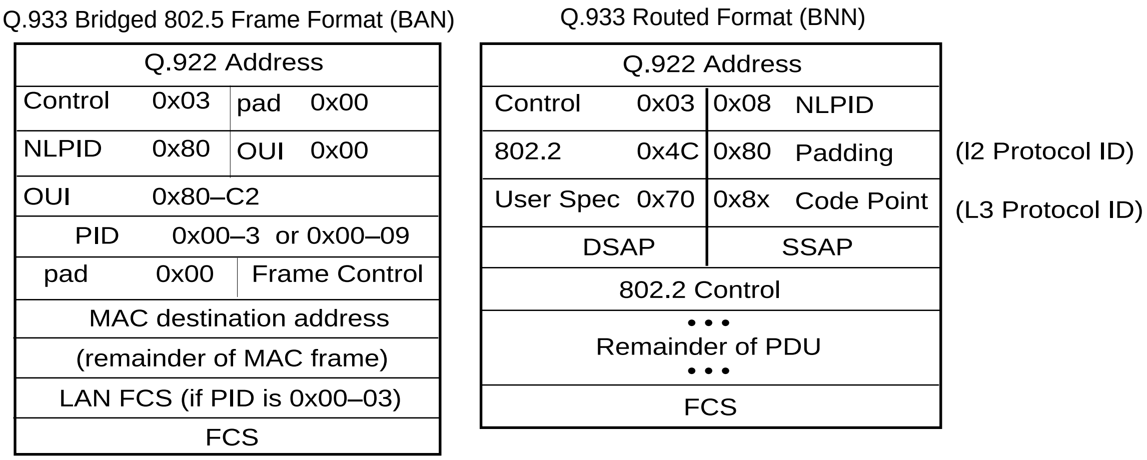
0x83 - SNA APPN (FID2)

0x84 - NetBIOS

These code points are used only by BNN for routed frames. BAN uses MAC address mapping and does not require code points.

The 3Com FRAD implementation does not support 0x81 and 0x84 data traffic. For 0x82, NCP processes the data as non-APPN peripheral data. For 0x83, NCP processes the data as APPN peripheral data.

Figure 26-5 shows the RFC 1490 encapsulation format for SNA and NetBIOS.



Q.933 Bridged 802.5 Frame Format (BAN)

Q.922 Address			
Control	0x03	pad	0x00
NLPID	0x80	OUI	0x00
OUI	0x80–C2		
PID	0x00–3 or 0x00–09		
pad	0x00	Frame Control	
MAC destination address			
(remainder of MAC frame)			
LAN FCS (if PID is 0x00–03)			
FCS			

Q.933 Routed Format (BNN)

Q.922 Address				
Control	0x03	0x08	NLPID	
802.2	0x4C	0x80	Padding	(I2 Protocol ID)
User Spec	0x70	0x8x	Code Point	(L3 Protocol ID)
DSAP		SSAP		
802.2 Control				
• • • Remainder of PDU • • •				
FCS				

Figure 26-5 RFC 1490 Encapsulation Format for BAN and BNN Implementation

BNN Configuration

RFC 1490 encapsulation eliminates the bridge/router on the host side; the SNA over the Frame Relay connection is terminated directly by the FEP. Figure 26-6 shows a configuration in which the Frame Relay connection is terminated directly to the FEPs in a BNN configuration.

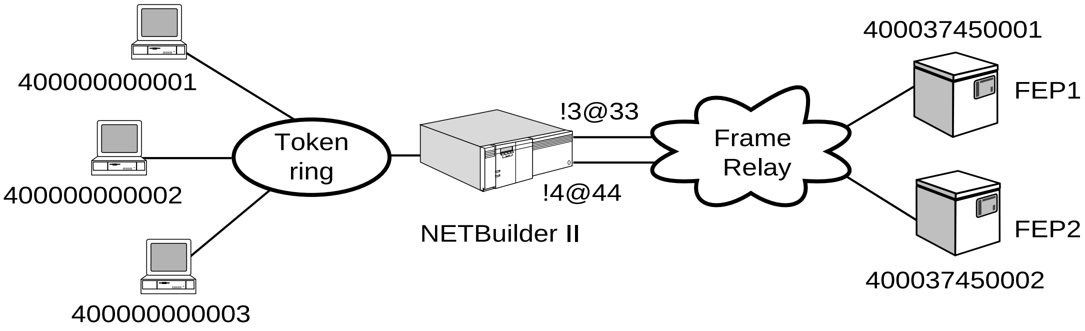


Figure 26-6 SNA Over Frame Relay Terminated by FEPs BNN Implementation

RFC 1490 encapsulation is implemented in NCP V7R1. With this implementation, the FEP can be attached directly to the Frame Relay network. This reduces the number of conversion points in the network, and can result in fewer network failure points for SNA traffic. BAN uses NCP V7R3. Updates are available for NCP V7R1 and V7R2.

Another advantage of RFC 1490 encapsulation is that it carries only LLC2 information (without the bridging frames), which places less overhead on the network because no broadcasting occurs (that is, no fan-out of frames to other bridging ports in the bridge/router). By eliminating bridging and broadcast frames, performance can be improved.

Since no broadcasting occurs, the NETBuilder bridge/router must specifically map the incoming MAC and SAP address to a specific outbound datalink connection identifier (DLCI), and vice versa. This mapping is performed using the -DLSw FradMap parameter syntax as follows:

```
ADD !<port> -DLSw FradMap <src mac> <src sap> <fep mac> <fep sap>
  <DLCI> <code point>
```

In the syntax, <src mac> and <src sap> are the MAC and SAP addresses of the SNA end station.

For RFC 1490 encapsulation, the <fep sap> value, in conjunction with the <src sap> and <code point> values, is required. For outbound (host bound) traffic, the value specified for <fep sap> will be placed in the source service access point (SSAP) field, while the value specified for <src sap> will be placed in the destination service access point (DSAP) field (see Figure 26-5). The value specified for <fep sap> must match the address specified in the DLCADDR keyword on the VTAM PATH definition statement; for more information, refer to the *IBM VTAM Resource Definition Reference* and the *NCP/SSP/EP Resource Definition Guide*.

The <src mac> and <fep mac> values must be specified in noncanonical format. The <src sap> and <fep sap> values must be in the range of 0-FC and divisible by 4.

For outbound (host bound) traffic, the bridge/router uses <src mac>, <src sap>, and <fep mac> to find the mapped Frame Relay partner in the mapping table. For inbound traffic (from the host), the bridge/router uses the combination of <fr port>, <dlci>, and <fep sap> to find the mapped LLC2 partner in the mapping table.

The maximum number of mapping entries allowed in the mapping table is 250.

When configuring the bridge/router for FRAD, follow these mapping rules:

- The combination of <src mac>, <src sap>, and <fep mac> in the mapping table must be unique.
- The combination of <fr port>, <fep sap>, and <dlci> in the mapping table must be unique.
- The mapping between (<src mac>, <src sap>, <fep mac>) and (<FR port>, <fep sap>, <dlci>) must be one-to-one.

By following these mapping rules, you can multiplex more than one SNA Link connection on a single DLCI.

The FRAD uses the local switching feature of DLSw, and inherits the advantages of DLSw such as local termination of data link traffic. However, the FRAD is also subject to the same limitations as DLSw. For example, FEP-to-FEP (FID4) traffic is not supported. For more information on DLSw, refer to Chapter 25.

BAN Configuration

BAN uses the RFC 1490 encapsulation implemented in NCP V7R3. BAN works similarly to BNN. The difference is that BNN uses setup coding to recognize each FEP and BAN uses MAC addressing.

Figure 26-7 shows a configuration in which the Frame Relay connection is terminated directly to the FEPs in a BAN configuration.

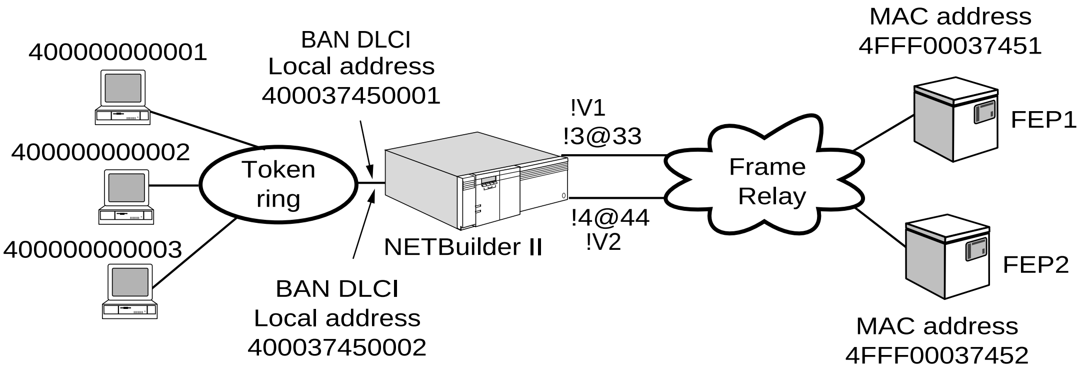


Figure 26-7 SNA Over Frame Relay Terminated by FEPs, BAN Implementation

Each Frame Relay BAN PVC is assigned a MAC address that is known as the BAN DLCI MAC address. This MAC address is assigned a virtual port and the BAN device listens for this MAC address on the LAN ports. The FEP recognizes the MAC address as a BNI. Both the BAN DLCI MAC and the BNI address may be the same. If they are different, the BAN device maps the two addresses.A test frame is sent to the BAN DLCI address. The first device that answers is the device to which the connection is made. This connection occurs when LLC2 receives the frames for a LAN device being sent to the BAN DLCI MAC address. Since no device is registered for the address, the frame is then forwarded to datalink switching (DLSw), which forwards the frame to all its tunnels including the one for local switching. Since the frame was received on a LAN port, it will not be forwarded out of any LAN port.

27 CONFIGURING TUNNELS TO CONNECT PEER SNA NETWORKS

This chapter describes the procedures for configuring tunnels on your system to connect peer networks running IBM's Systems Network Architecture (SNA).

The term "peer" as used in this chapter has a different meaning from what is usually used in SNA environments (such as in "peer-to-peer communications").

Using tunnels, you can configure a connection from a 3Com bridge/router on one SNA network to a 3Com bridge/router on a second SNA network, sending the connection across an internetwork. You also can configure a tunnel connection between a token ring network in a source route bridging domain with an Ethernet network in a transparent bridging domain. This tunnel may be configured within one bridge/router so that all IP processing is performed internally requiring no external internetwork.

For conceptual information and terminology, refer to "How SNA Tunnel Connections Work" on page 27-13.

Configuring Tunnels for Terminal-to-Host SNA Sessions

LLC2 tunnels are static configurations that map the Logical Link Control (LLC) address of the SNA device with the IP address used to transport the connection across the IP internetwork. You can configure each tunnel statically on both sides of the IP tunnel when operating in secure mode, or you can allow the destination side of the tunnel to accept connections from any source tunnel.

Most SNA applications involve a terminal user at one location accessing a host at another location. This section describes how to configure a bridge/router on a terminal network and a bridge/router on a host network so that terminal users can initiate tunnel connections to the host.

3Com bridge/routers at both ends of the tunnel must be running software with LLC2 functionality. Although the figures in this chapter show only token ring networks, you can use all media types that the bridge/router supports (including Ethernet and Fiber Distributed Data Interface (FDDI)).

When you set up a tunnel, the order in which you define tunnel elements is important. For example, you need to define the tunnel interface before the tunnel media access control (MAC), tunnel peer, or tunnel control. In addition, you need to define a tunnel peer before the MAC address.

Figure 27-1 is an example of a bridge/router using LLC2 tunneling to connect a terminal user on an attached token ring network with an SNA host on a peer network across the IP internetwork. Table 27-1 lists the commands you must enter for the bridge/routers at both the terminal side and the host side. Each of these procedures are described in sections that follow the figure.

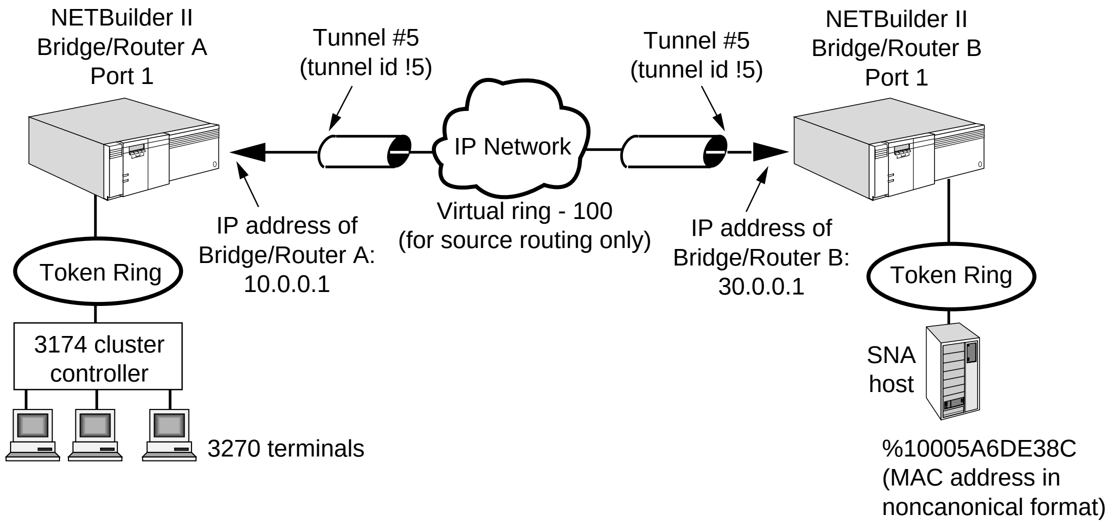


Figure 27-1 LLC2 Tunnel Configuration for Standard SNA Host Sessions

Table 27-1 Commands to Configure LLC2 Tunneling for SNA

Commands Entered on bridge/router A	Commands Entered on bridge/router B
SETD -TCP CONTrol=KeepALive	SETD -TCP CONTrol=KeepALive
SETD -TCP KeepALive=3	SETD -TCP KeepALive=3
SETD !1 -LLC2 CONTrol=Enable	SETD !1 -LLC2 CONTrol=Enable
ADD !5 -LLC2 TUNnelInterface 10.0.0.1	ADD !5 -LLC2 TUNnelInterface 30.0.0.1
ADD !5 -LLC2 TUNnelPeer 30.0.0.1	ADD !5 -LLC2 TUNnelPeer 10.0.0.1
MAC 10005A6DE38C	SETD -LLC2 TUNnelVRRing =100
ADD !5 -LLC2 TUNnelMacAaddr 08005AB6C731	
SETD -LLC2 TUNnelVRRing =100	

Configuring the Tunnel for the Terminal End

This section describes how to configure the tunnel on the bridge/router at the terminal end of the SNA host session. For a terminal-to-host SNA session across a tunnel, you must also configure the tunnel on the bridge/router at the host end of the session. After you complete the procedures in this section, proceed to "Configuring the Tunnel for the Host End" on page 27-5.

Prerequisites

Before beginning this procedure, complete the following steps:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the token ring LAN as described in Chapter 5.
- Configure the LLC2 data link interface as described in Chapter 21.
- Obtain the IP address of the local bridge/router.
- Obtain the IP address of the tunnel peer (the remote bridge/router).
- Obtain the MAC address of the host on the remote network (the peer MAC address).

Procedure

To configure the tunnel for the terminal end of the tunnel, follow these steps on bridge/router A serving the network on which the terminal resides:

1 Enable transmission of transmission control protocol (TCP) keepalive packets.

TCP keepalive packets notify the bridge/router when the TCP connection has ended. Without TCP keepalive packets, the bridge/router will not detect that the TCP connection is down due to an abnormal situation such as link failure. This can result in LLC2 sessions being kept active even though the corresponding TCP session has ended.

a Enable TCP keepalive packets by entering:

```
SETDefault -TCP CONTrol = KeepAlive
```

b Specify the number of keepalive packets to be transmitted using:

```
SETDefault -TCP KeepAliveLimit = <retry>(1–15)
```

2 Enable the LLC2 Protocol on the local bridge/router.

For example, if network A is connected to port 1 on bridge/router A, enable the LLC2 Protocol by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

The specified port can be a LAN or WAN port that maintains the LLC2 connections with the LLC2 end station.

3 Configure an IP tunnel on the local bridge/router.

The number of tunnels you can configure depends on the hardware platform you are using. With the TUNnelInterface parameter, you can map the specified tunnel to the local IP address of bridge/router A.

If you are configuring your NETBuilder bridge/router as an IP router, the port associated with this IP address must be active before any packets can be sent to or received by this IP address. Select an IP address associated with a port that is always up or is the most reliable, such as a LAN port.

For example, to map tunnel 5 shown in Figure 27-1 to bridge/router A's IP address (10.0.0.1), enter:

```
ADD !5 -LLC2 TUNnelInterface 10.0.0.1
```

This example creates the tunnel number 5 mapped to the local IP address 10.0.0.1 on bridge/router A. The tunnel is mapped to the address of the bridge/router, and not to an address of an individual port. For more information on the possible values of the TUNnelInterface parameter, refer to Chapter 34 in *Reference for NETBuilder Family Software*.

All Internet addresses for peer bridge/routers must be known in the routing table of the local bridge/router, either dynamically through routing information protocol (RIP) or open shortest path first (OSPF), or statically configured in the IP routing tables. For more information on IP routing, refer to Chapter 6.

4 Configure the tunnel peer (the bridge/router at the remote end of the tunnel), specifying the peer IP address of the tunnel peer and, if desired, a name for the tunnel peer.

For example, to configure the tunnel peer named “SJose” for tunnel number 5 to bridge/router B’s IP address (30.0.0.1), enter:

```
ADD !5 -LLC2 TUNnelPeer 30.0.0.1 “SJose”
```

This command sets the bridge/router at address 30.0.0.1 as the tunnel peer for tunnel number 5, and assigns the name “SJose” to that tunnel peer. The tunnel peer name is optional. The character string for the tunnel peer name is limited to 16 characters.

You can configure more than one tunnel on the bridge/router, but you must configure a separate tunnel for each tunnel peer.

When configuring the Internet address for the tunnel peer, you do not specify the port number of the bridge/router where the connection will be made. When a tunnel connection is made, the bridge/router determines the port through which the peer Internet address can be reached.

5 Obtain the MAC address of the host on the remote network (the peer MAC address).

6 Convert the peer MAC address to canonical (Ethernet) format by entering the MacAddrConvert <MacAddress> command.

This step is necessary because media addresses in SNA environments are presented in noncanonical (token ring) format. However, when configuring the MAC address of the host for the tunnel in the next step, you must enter the address in canonical (Ethernet) format.

For example, to convert the MAC address of the remote network host shown in Figure 27-1 from noncanonical format to canonical format, enter:

```
MacAddrConvert %10005A6DE38C
```

The system displays the address in canonical format as follows:

```
%08005AB6C731
```

This address is used in the next step.

7 Configure the MAC address of the host on the remote network (the peer MAC address).

For example, to configure the MAC address of the remote host for tunnel number 5, enter:

```
ADD !5 -LLC2 TUNnelMACadd 08005AB6C731
```

CAUTION: *It is not always easy to determine whether a MAC address is in canonical or noncanonical format. If you use a noncanonical address when configuring the MAC address, the tunnel connection will not be made, and you will not receive an error message.*

To check if tunnel connections are taking place, enter:

```
SHow -LLC2 SESSions
```

or

```
SHow -SYS STATistics -LLC2
```

If the statistics displayed show no values tabulated, this may indicate you configured the incorrect MAC address format. For more information on LLC2 statistics, refer to Appendix H.

CAUTION: *If the destination MAC address of an end system has been added to a tunnel on a bridge/router, all supported LLC2 session traffic to that destination will be tunneled. For example, you cannot bridge NetBIOS packets and tunnel LLC2 SNA packets to the same destination MAC address through the same bridge/router.*

8 If you are using source routing, you must define the internetwork with a unique ring number.

When source routing is used, the internetwork becomes a virtual ring. If your end systems are using token ring source routing, the bridge/router and the IP tunnel appear to the end systems as a source route bridge with a token ring network attached to the other side of the bridge/router. The default ring number of this virtual ring is decimal 92.

If necessary, use the SETDefault -LLC2 TUNnelVRing = <Number> (1-254) syntax to change the ring number for the internetwork, in which both token ring networks on the peer internetworks interpret the internetwork as an intermediate token ring network. This virtual ring number must match on all peer bridge/routers used for tunneling and must be unique within the token ring network.

For example, to configure the virtual tunnel ring as ring number 100, enter:

```
SETDefault -LLC2 TUNnelVRing = 100
```

After you have completed the above procedure, go to the next section to configure the tunnel for the host end of the SNA session.

Configuring the Tunnel for the Host End

This section describes how to configure the tunnel for the host end of the tunnel.

Prerequisites

Before beginning this procedure, complete the following steps:

- Configure the tunnel for the terminal end as described in the previous procedure.
- Obtain the IP address of the remote bridge/router.
- Obtain the IP address of the tunnel peer.

Procedure

To configure the tunnel for the host end of the tunnel, follow these steps on bridge/router B serving the network on which the host resides:

1 Enable the LLC2 Protocol on the remote bridge/router.

For example, if network B is connected to port 1 on bridge/router B, enable the LLC2 Protocol by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

The specified port can be a LAN or WAN port that maintains the LLC2 connections with the LLC2 end station.

2 Configure an IP tunnel on the remote bridge/router.

For example, to map tunnel 5 to bridge/router B’s IP address (30.0.0.1) shown in Figure 27-1, enter:

```
ADD !5 -LLC2 TUNnelInterface 30.0.0.1
```

This example creates the tunnel number 5 mapped to local network address 30.0.0.1. The tunnel number does not have to match the number used on the bridge/router for the terminal network side. The tunnel number is used for identification on the local bridge/router only.

For more information about the TUNnelInterface parameter values, refer to Chapter 34 in *Reference for NETBuilder Family Software*.

3 Configure the tunnel peer (the bridge/router at the remote end of the tunnel), specifying the network address of the tunnel peer and, if desired, a name for the tunnel peer.

For example, to configure bridge/router A as a tunnel peer for tunnel number 5, enter:

```
ADD !5 -LLC2 TUNnelPeer 10.0.0.1 "SDiego"
```

This command sets the bridge/router at address 10.0.0.1 as the tunnel peer for tunnel number 5, and assigns the name "SDiego" to that tunnel peer. The tunnel peer name is optional. The character string for the tunnel peer name is limited to 16 characters.

4 If you are using source routing, you must define the internetwork with a unique ring number.

When using source routing, the internetwork has become a virtual ring. If your end systems are using token ring source routing, the bridge/router and the IP tunnel appear to the end systems as a source route bridge with a token ring network attached to the other side of the bridge/router. The default ring number of this virtual ring is decimal 92.

If necessary, use the SETDefault -LLC2 TUNnelVRing = <Number> (1-254) syntax to change the ring number, in which both token ring networks on the peer internetworks interpret the internetwork as an intermediate token ring network. The virtual ring number must match on all peer bridge/routers used for tunneling and must be unique within the token ring network.

For example, to configure the virtual tunnel ring as ring number 100, enter:

```
SETDefault -LLC2 TUNnelVRing = 100
```

After you have completed this procedure, a user on a terminal at one end of the tunnel should be able to initiate a connection to the host at the other end of the tunnel. For information on disabling tunnels and tunnel connections from the bridge/router, refer to "Disabling Tunnels and Tunnel Connections" on page 27-10.

Additional terminals can now connect to the same host without additional changes.

Verifying the Configuration

After you have configured a tunnel, you can display information about the tunnel, including associated addresses on the peer internetwork.

To display complete tunnel configuration information, enter:

```
SHow -LLC2 CONFiguration
```

To display the control status of ports (meaning whether a port is enabled or disabled) being used for tunnel connections, enter:

```
SHow -LLC2 CONTrol
```

The ! syntax in this display indicates the port number. In subsequent displays in this section, the ! syntax indicates the tunnel identification number.

Displaying Tunnel Status

To display the control status of tunnels (meaning whether a tunnel is enabled or disabled), enter:

`SHow -LLC2 TUNnelControl`

To display the configuration status of all local tunnels, enter:

`SHow -LLC2 TUNnelInterface`

The display is similar to the following:

```
--------------------------Tunnel Interfaces--------------------------
Tunnelid  Local IP Address  Tunnel Port  Tunnel Transport  Type
!4        129.213.240.214   2049         TCP               LOCAL_TERM
!6        129.213.240.250   2049         TCP               LOCAL_TERM
```

In this display, the Tunnel Port is set with the ADD -LLC2 TUNnelInterface <local network IP address> [tunnel transport port] syntax. The type “LOCAL_TERM” indicates that the LLC2 session for that peer is terminated locally. TRANSPARENT type indicates that no local acknowledgment of the peer connection takes place; all data is passed to the other side of the tunnel as is.

Displaying Tunnel Peers and MAC Addresses

To display tunnel peers for the tunnels you have configured, enter:

`SHow -LLC2 TUNnelPeer`

The display is similar to the following:

```
=========================== SHow -LLC2 TUNnelPeer ===================
-------------------------------Tunnel Peers-----------------------
Tunnelid    Local IP Address       Peer IP Address      Peer Name
!4          129.213.240.214        139.200.108.164
!6          129.213.240.250        139.200.225.100
```

To display a list of MAC addresses for tunnel peer end stations, enter:

`SHow -LLC2 TUNnelMACadd`

The display similar to the following appears, which shows the peer Internet address and the MAC addresses of the end stations configured for that peer:

```
----------------------Tunnel Peer Mac Addresses-------------------
Tunnelid    Peer IP Address       Peer Mac Address       Saps
!5          129.213.240.214       %02608C3C36AB          00 to 0C
-----------------------------Tunnel Peers---------------------------
Tunnelid    Local IP Address      Peer IP Address        Peer Name
!5          139.80.20.5           129.213.240.214        frank
```

Displaying Tunnel Sessions

To display the status of current tunnel sessions, enter:

`SHow -LLC2 TUNnelDisplay`

The display is similar to the following:

```
----------------------------------Tunnel Display Sessions-----------------------------------
Id  Status   Local IP Addr      Peer IP Addr       Peer Mac         Sap  Local Mac        Sap
!4  UP       129.213.240.214    129.213.240.215    %02608C3C36AC    04   %02608C3C36AB    04
                                                   %02608C1A0CE7    04   %02608C3C36AB    04
!6  DOWN     129.213.240.250    UNDEFINED
```

This display shows both the local and peer IP address for a tunnel (the IP addresses on the peer bridge/routers), and both the local and peer MAC addresses for the tunnel (the MAC addresses for the SNA client and server nodes).

The MAC addresses shown in this display are the MAC addresses for the SNA client and server. To display the MAC addresses of the bridge/router tunnel peers, enter:

```
SHow -LLC2 SESSions
```

Customizing Tunnels

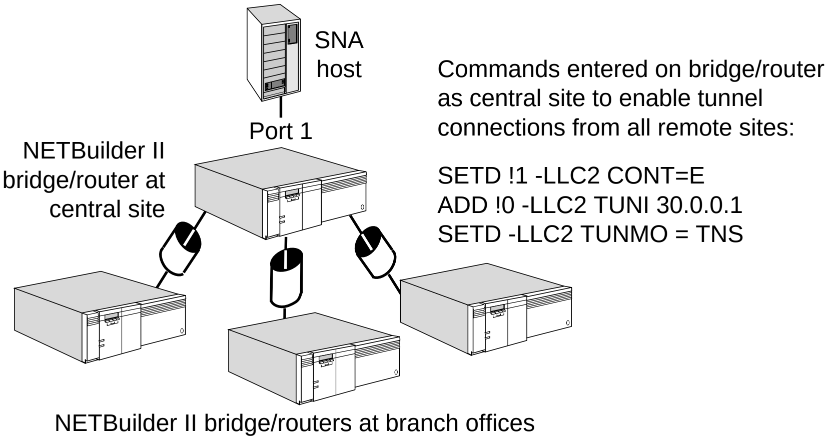
This section describes how to customize your tunnel configuration. Procedures in this section include:

- Configuring the host side so that the bridge/router can accept tunnel connections from any remote bridge/router.
- Configuring tunnels for peer-to-peer (LU6.2) SNA sessions.
- Displaying tunnels and tunnel connections.
- Deleting tunnels, tunnel peers, and peer end stations.

Configuring the Bridge/Router for Remote Connections

By default, the bridge/router can only accept tunnel connections from configured tunnel peer bridge/routers. This helps prevent users from unauthorized sites from accessing the host. However, you can configure the bridge/router on the host side so that the bridge/router can accept tunnel connections from any remote bridge/router. This can be useful if you plan to have terminal users at many different remote sites making tunnel connections to the host.

Figure 27-2 shows a configuration in which the bridge/router at the central site accepts incoming tunnel connections from three branch offices to access the central site host.


SNA host
Port 1
NETBuilder II bridge/router at central site
NETBuilder II bridge/routers at branch offices

Commands entered on bridge/router as central site to enable tunnel connections from all remote sites:

```
SETD !1 -LLC2 CONT=E
ADD !0 -LLC2 TUNI 30.0.0.1
SETD -LLC2 TUNMO = TNS
```

Figure 27-2 LLC2 Tunnel Configuration for Central Site Bridge/Router

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/router according to Chapter 1.
- Configure the LLC2 data link interface as described in Chapter 21.
- Obtain the IP address of the central site bridge/router.

Procedure

To configure a central site bridge/router to accept any incoming tunnel connection request, follow these steps:

1 Enable tunneling so that system can receive connections from the host.

For example, if the SNA host is connected to port 1 of the bridge/router at the central site, enable tunneling by entering:

```
SETDefault !1 -LLC2 CONTrol = Enable
```

2 Set the central site bridge/router to accept multiple tunnel connections, specifying the tunnel as !0 and the IP address of the bridge/router by entering:

```
ADD !0 -LLC2 TUNnelInterface 30.0.0.1
```

The !0 syntax simplifies configuration because multiple tunnels do not need to be configured at the central site. The !0 syntax can only be used at the central site to listen for incoming tunnel connections.

If you set the central site tunnel interface to !0, the remote peer bridge/router must configure this IP address as a tunnel peer using a nonzero tunnel interface.

3 Set the central site bridge/router to accept all tunnel connection requests (including requests from bridge/routers that are not configured tunnel peers) by entering:

```
SETDefault -LLC2 TUNnelMOde = TunnelNonSecure
```

With this configuration, you do not need to configure any other tunnel information on the central site bridge/router. However, you can configure your central site bridge/router in different ways to allow incoming tunnel connections from either configured tunnel peers or from any bridge/router, and over any port or over all ports. Table 27-2 lists the different options for controlling incoming tunnel connection requests.

Table 27-2 Controlling Incoming Connection Requests to bridge/router Serving the SNA Host

Command Options	Result

`ADD !0 -LLC2 TUNnelInterface 30.0.0.1`
`SETDefault -LLC2 TUNnelMOde = TunnelNonSecure`
	Incoming tunnel connection requests are allowed over all ports, and all tunnel connection requests are accepted, including requests from bridge/routers that are not configured tunnel peers.

`ADD !4 -LLC2 TUNnelInterface 30.0.0.1`
`ADD !4 -LLC2 TUNnelPeer 10.0.0.1`
	Incoming tunnel connection requests are allowed over the specified tunnel only (tunnel number 4 in this example), and are only accepted from the configured tunnel peer (IP address 10.0.0.1 in this case).

`ADD !4 -LLC2 TUNnelInterface 30.0.0.1`
`SETDefault -LLC2 TUNnelMOde = TunnelNonSecure`
	Incoming tunnel connection requests are allowed over the specified port only, but one tunnel connection request made over this port is accepted, including requests from bridge/routers that are not configured tunnel peers.

Configuring Tunnels for Peer-to-Peer (LU6.2) SNA Sessions

The terminal user is replaced by an Advanced Peer-to-Peer Communications (APPC) client and the SNA host is replaced by an APPC server. The bridge/router configuration requirements are exactly the same because the SNA terminal or the APPC client can initiate the LLC2 connection with the SNA host or the APPC server.

The destination MAC address of the SNA host or the APPC server must be mapped to a tunnel using the ADD |<tunnelid> -LLC TUNnelMAcadd <peer mac address> command. This mapping allows the bridge/router that receives the LLC2 connection request to forward the request over the IP internetwork to the tunnel peer bridge/router, which then forwards the LLC2 connection request to the SNA host or APPC server. If the SNA host or APPC server is enabled, it will respond to this LLC2 connection request, which establishes an LLC2 session.

You do not need to map the destination MAC addresses of the SNA terminal or the APPC client because neither the SNA host nor the APPC server initiates the LLC2 connection request.

If there is a reason for either side to initiate the LLC2 connection request, for example, APPC nodes that act as both client and server, then the appropriate destination MAC address must be mapped in both tunnel peer bridge/routers.

For more information on Peer-to-Peer (LU6.2) SNA sessions, refer to “Configuring Tunnels for Terminal-to-Host SNA Sessions” on page 27-1.

Disabling Tunnels and Tunnel Connections

You can disable tunneling for a specific tunnel, tunneling to and from a specific peer internetwork, or all tunneling on the local bridge/router.

To disable tunneling from a tunnel to a peer network, use:

```
SETDefault !<tunnelid> -LLC2 TUNnelControl = Disable
```

To disable all tunneling, enter:

```
SETDefault -LLC2 TUNnelMOde = TunnelDisable
```

Deleting Tunnels, Tunnel Peers, and Peer End Stations

You can delete tunnels, tunnel peer internetworks, and individual end stations on tunnel peer internetworks.

To delete a tunnel, enter the tunnel number of the tunnel to be deleted and the local network IP address using:

```
DELete !<tunnelid> -LLC2 TUNnelInterface <IPaddress>
```

CAUTION: *If you delete a tunnel and later decide you want to establish it again, you will have to reconfigure it from the beginning. Instead you can temporarily disable the tunnel. For instructions, refer to “Disabling Tunnels and Tunnel Connections” on page 27-10.*

To delete a tunnel peer (in other words, the peer internetwork for a specific tunnel), enter the tunnel ID and the peer network IP address using:

```
DELete !<tunnelid> -LLC2 TUNnelPeer <IP address>
```

To delete an end station from a peer internetwork, enter the tunnel ID and the peer MAC address, specifying the MAC address (in canonical format) of the end station using:

```
DELete !<tunnelid> -LLC2 TUNnelMAcadd <peer MAC address>
```

Enhancing Tunnel Performance

The section describes how to increase performance and handle the following situations:

- Tunneling for high traffic loads on a token ring LAN
- Multiple tunnels between two systems
- Slow file transfers and excessive LLC2 flow controlling
- Excessive LLC2 rejects

Tunneling for High Traffic Loads on a Token Ring LAN

When a bridge/router token ring port is configured to support LLC2 end systems and there is high traffic loads between end systems on that local LAN (including bridged traffic), then this bridge/router port should be configured for transparent bridging to improve performance and reduce frame-copied error reports. This action is necessary because when LLC2 is enabled on a port using the SETDefault | <port> -LLC2 CONTrol = Enable syntax, the bridge/router places that port in promiscuous mode. If you also configure this port for transparent bridging, the bridge/router token ring hardware filters local traffic; otherwise, this filtering is done by software, which can impact performance.

Procedure

To configure tunneling for high traffic loads, follow these steps:

1 Enable global bridging by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

After global bridging has been enabled, the bridge/router forwards bridged packets on any port that has transparent bridging enabled as well as performs the desired hardware filtering of local traffic.

2 Disable transparent bridging on ports over which you do not intend to bridge.

Because transparent bridging is enabled on all ports by default, if you do not intend to bridge, you should disable transparent bridging on all ports except those ports in which you have LLC2 enabled. To disable transparent bridging on all ports except port 1 (the LLC2 port that connects to the LLC2 end system), use:

```
SETDefault !<port> -BRidge TransparentBRidge = NoTransparentBRidge
```

For the bridge/router to learn addresses on the token ring network, it copies the packet with the unknown destination address and sets the address-recognized (A) and frame-copied (C) bits in the Frame Status (FS) field. End systems will generate frame-copied error reports when the FS(A) and (C) bits have been set and the destination of the frame is a local end system.

Multiple Tunnels Between Two Systems

If you have a configuration with multiple possible tunnels between two systems, A and B, how you configure the tunnels determines whether the connections can be made.

For example, suppose system A has tunnels configured in the following way:

TunnelID	Local IP	Peer IP	TunnelControl Parameter
!0	System A	Undefined	Enabled
!1	System A	Undefined	Disabled
!2	System A	System B	Disabled

In this situation, on an incoming connection request from B, tunnel !2 is tried first (since both the local and peer IP addresses are defined); the connection fails because tunnel !2 is disabled. System B will *not* retry the other tunnels.

In a situation in which only tunnels !0 and !1 are configured, on an incoming connection request from B, !1 is tried first (non !0 tunnel). The connection fails also because the tunnel is disabled; however, it then retries !0, and succeeds.

Slow File Transfers and Excessive LLC2 Flow Controlling

To speed up file transfers and reduce LLC2 flow controlling, and to increase performance in general, increase the TCP window size by entering:

```
SETDefault -TCP WINdow = 4096
```

Increase the maximum segment size that the TCP layer of the bridge/router can receive by entering:

```
SETDefault -TCP MaxSegmentSize = 4096
```

Excessive LLC2 Rejects

When excessive rejects happen, reduce the LLC2 transmit window size on the bridge/router port or the attached device.

To reduce the transmit window on a port of the bridge/router, use:

```
SETDefault !<port> -LLC2 TransmitWindow = <1–3>
```

LLC2 Configuration

3Com tunneling does not recognize the Spanning Tree Protocol (STP). Because of this limitation, you must avoid configuring a second data path that can loop SNA and NetBIOS traffic back to an originating router. Do not configure either bridge or tunnel paths as second data paths. Avoid the topology shown in Figure 27-3 because it may duplicate packets and cause failure.

Your site may require redundancy in a data link switching (DLSw) environment. If you need tunnel redundancy or DLSw bridge/router redundancy, contact your network supplier for planning assistance.

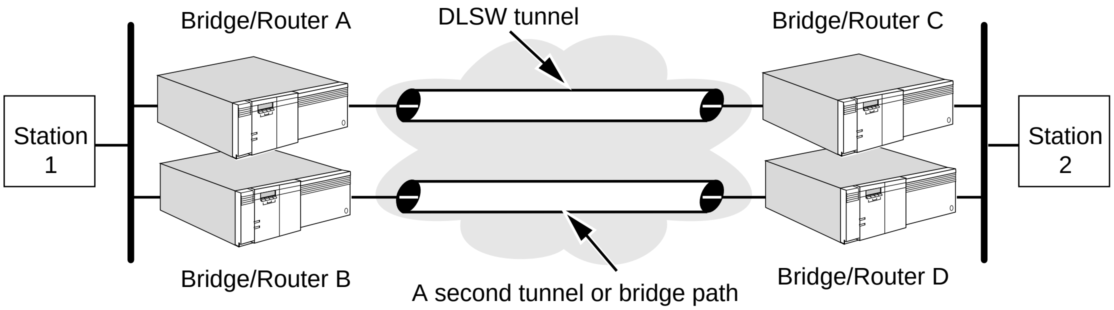


Figure 27-3 Illegal LLC2 Tunneling Configuration

Source Route Transparent Bridging, LLC2 Tunneling

Source route transparent (SRT) bridging, SRT DLSw switching, and SRT LLC2 tunneling require newly configured SRT ports to be reenabled before the configuration takes effect. After configuring the SRT port, enter:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

How SNA Tunnel Connections Work

Tunneling is a method of connecting peer networks that can be reached only across an internetwork using a different routing protocol than is being used on the peer networks. With tunneling, the complete network packet is encapsulated within a foreign protocol and routed over the foreign internetwork to the peer network.

Using the LLC2 tunneling over IP feature, the bridge/router encapsulates SNA LLC2 frames within Transmission Control Protocol/Internet Protocol (TCP/IP) packets for transmission over an IP internetwork to remote SNA networks. The frames are routed through the IP network and are treated as normal TCP/IP network frames. At the other end, the TCP/IP packets received from the IP network are disassembled into LLC2 frames for transmission over the local SNA network media. (The LLC2 frames can be terminated locally to eliminate LLC2 polling traffic across the IP internetwork.)

Figure 27-4 is an example of a tunneled connection from a bridge/router across the IP Internet to a bridge/router on a peer SNA network. In this example, the end station on the local network can access the SNA host on the peer network.

For information on configuring tunnels, refer to “Configuring Tunnels for Terminal-to-Host SNA Sessions” on page 27-1. For information on configuring the LLC2 data link interface, refer to Chapter 21.

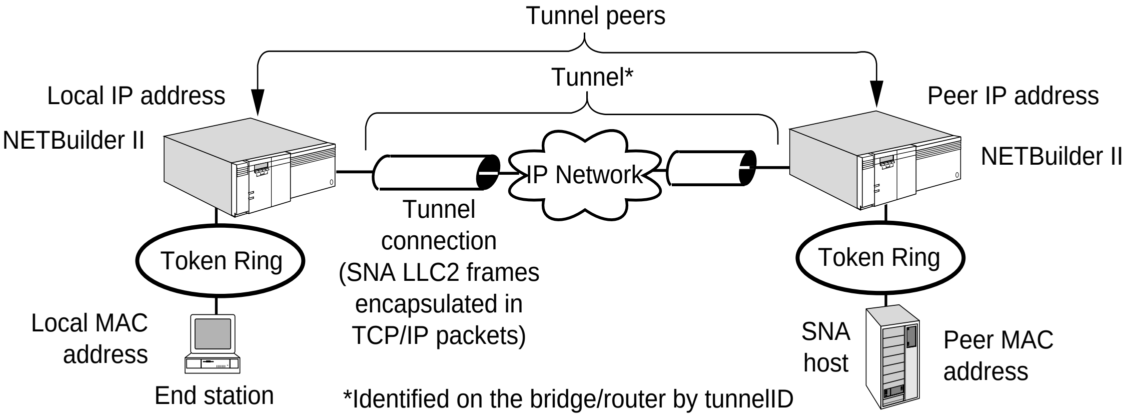


Figure 27-4 Using LLC2 Tunneling to Route SNA Packets Across an Internetwork

LLC2 Tunneling Terms

A list of important terms that are used in this chapter is provided to explain LLC2 tunneling concepts; see Figure 27-4 to show where the explained terms logically reside on the configuration. The terms are shown in the figure and are explained from the viewpoint of the “local” bridge/router on the left.

tunnel	A logical connection across an IP internetwork between two bridge/router tunnel peers that enable an SNA session to take place. The tunnel is defined by the path from the local IP address to the peer IP address.

tunnel ID
A local identifier that identifies the tunnel only on the bridge/router where the tunnel is configured. The tunnel ID number is *not* transmitted across the tunnel. The tunnel ID on the bridge/router at one end of a tunnel does not need to match the tunnel ID at the other end of the tunnel.

tunnel peer
The bridge/router on the other end of the tunnel. For a successful tunnel connection, the user at the bridge/router must configure the bridge/router on the receiving end as its tunnel peer.

Local IP address
The IP address of the bridge/router at the end of the tunnel where the tunnel is configured.

Peer IP address
The IP address of the bridge/router at the other end of the tunnel.

Local MAC address
The media access control (MAC) address of the SNA node (such as a terminal controller) making a connection request to the host. Although the MAC address may be in noncanonical (token ring) format, it must be configured in canonical (Ethernet) format.

Peer MAC address
The MAC address of the SNA node (normally a host) receiving the connection request. Although the MAC address may be in noncanonical (token ring) format, it must be configured in canonical (Ethernet) format.

28 CONFIGURING LAN ADDRESS ADMINISTRATION

This chapter describes how to use LAN Address Administration (LAA) to assign a media access control (MAC) address to a physical path or to the Communications Engine Card (CEC) interface, overriding the MAC address burned in the PROM on the physical interface.

Assigning MAC addresses to a path or CEC interface is supported for token ring ports only.

By assigning a MAC address to a path, you can use the same MAC address for multiple paths for load splitting purposes in Systems Network Architecture (SNA) environments. By assigning a MAC address to a path that is different from the MAC address burned in the physical interface PROM, you can hotswap modules on a port and still maintain the same MAC address.

CAUTION: *Using LAA on paths being used to route DECnet network traffic can cause problems in DECnet environments. For more information, refer to "Using LAA with DECnet" on page 28-4.*

Assigning a MAC Address to a Physical Path

To assign a MAC address to a physical path, follow these steps:

1 To assign a MAC address to a physical path (a different address from the one burned on the module PROM), use:

```
SETDefault !<path> -PATH MacAddress = %<MAC address> | Mac <MAC
  address> | Ncmac <MAC address>
```

You can enter the MAC address in one of two formats: canonical or noncanonical. To enter the address in canonical format, precede the address with the prefix "Mac" or the percent symbol (%). To enter the address in noncanonical format, precede the address with the prefix Ncmac. If you precede the MAC address with Mac or Ncmac, enter a space between the prefix and the address. If you precede the MAC address with the percent symbol (%), do not enter a space between the symbol and the address.

Bits 0 and 1 of the first byte must be set to 0 and 1 respectively. Bit 1 is the universally and locally administered bit. This limits the choice of addresses to the following set (where x can have any value).

In canonical format:

```
x2xx xxxx xxxx
x6xx xxxx xxxx
xAxx xxxx xxxx
xExx xxxx xxxx
```

In noncanonical format:

```
4xxx xxxx xxxx
5xxx xxxx xxxx
6xxx xxxx xxxx
7xxx xxxx xxxx
```

For example, to assign the canonical address 020002033D76 to path 2, enter:

```
SETDefault !2 -PATH MacAddress = Mac 020002033D76
```

To assign the noncanonical address 400040C0BC6E to path 2, enter:

```
SETDefault !2 -PATH MacAddress = NcMac 400040C0BC6E
```

To assign the MAC address 020002030EF2 in canonical format for token ring, enter:

```
SETDefault !2 -PATH MacAddress = %020002030EF2
```

To convert a MAC address from canonical format to noncanonical format and vice-versa, use the MacAddrConvert command. For more information, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

CAUTION: *Do not assign a multicast address for the MAC address. Also, do not assign a MAC address that is either a smart filtering MAC address or one of the bridge BPDU addresses.*

2. After you have reassigned the MAC address, re-enable the path using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

After you re-enable the path, the new MAC address assigned to the path will be shown when MAC addresses for any protocol are displayed. The new address remains assigned to the interface until you specifically reset the address. The new address remains assigned after you reboot the bridge/router.

3. If the LAA address is used by Advanced Peer-to-Peer Networking (APPN), you must deactivate and then activate the node control for the new address to be effective by entering:

```
SETDefault -APPN CONTrol = Deactivate
SETDefault -APPN CONTrol = Activate
```

4. Verify that the new MAC address has been assigned by entering:

```
SHow !* -PATH MacAddress
```

After you have assigned a MAC address to a path, you can reassign the path back to the MAC address burned on the PROM using:

```
SETDefault !<path> -PATH MacAddress = Reset
```

When you reset the MAC address, the address you previously assigned is deleted.

Figure 28-1 shows how you can use duplicate MAC addresses by reassigning an existing address. In the figure, you reassign the MAC address on path 3 to duplicate the address on path 2. The MAC address burned in the PROM of the module on path 3 still exists, but is not used for any connections. The MAC address burned in the PROM is transparent until the MAC address is reset.

You cannot set duplicate MAC addresses on the same ring in token ring environments. If you set duplicate MAC addresses on the same bridge/router, each path must be connected to different rings.

CAUTION: *Setting duplicate MAC addresses is recommended only for SNA and other connection-oriented protocols. In addition, setting duplicate MAC addresses will work only in source routing LAN environments. As a result, setting duplicate MAC addresses is not recommended on transparent bridges or source route transparent bridges.*

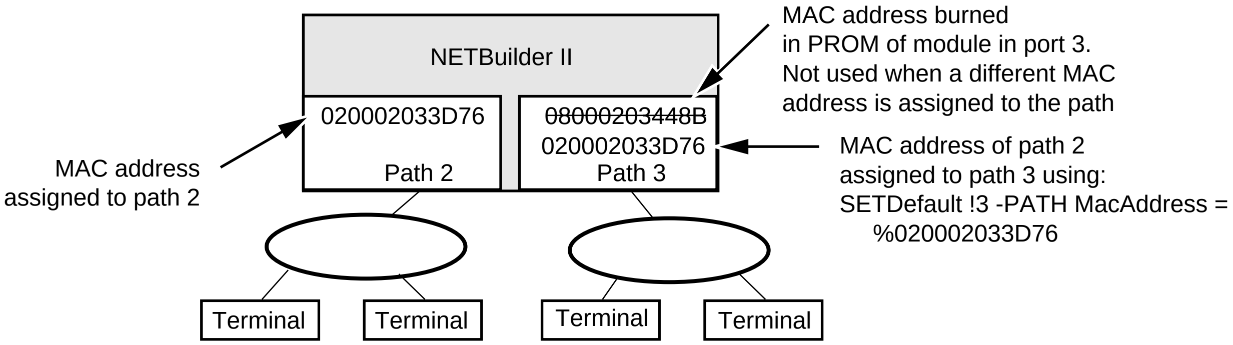

NETBuilder II
020002033D76
Path 2
~~08000203448B~~
020002033D76
Path 3
MAC address burned in PROM of module in port 3. Not used when a different MAC address is assigned to the path
MAC address of path 2 assigned to path 3 using: SETDefault !3 -PATH MacAddress = %020002033D76
MAC address assigned to path 2
Terminal
Terminal
Terminal
Terminal

Figure 28-1 Setting Duplicate MAC Addresses Using LAA

Assigning a MAC Address to a CEC Interface

You can assign a MAC address to the CEC interface on a NETBuilder II bridge/router. This can be useful in assigning a bridge/router to act as a backup network node in APPN environments.

To assign a MAC address to the CEC interface on a NETBuilder II bridge/router, follow the procedures in "Assigning a MAC Address to a Physical Path" on page 28-1. However, when setting the MacAddress parameter, instead of specifying a path number, specify !0 to represent the CEC interface. For example, to assign the noncanonical address 400040C0BC6E to the CEC interface, enter:

```
SETDefault !0 -PATH MacAddress = NcMac 400040C0BC6E
```

After you change the MAC address of the CEC interface, you must reboot the bridge/router.

Using Duplicate MAC Addresses for SNA Load Balancing

You can set duplicate MAC addresses to set up load balancing for SNA environments. In Figure 28-2, NETBuilder C and NETBuilder D are APPN network nodes in which duplicate MAC addresses are used on both so that connections to the host from the terminals at the bottom of the figure can go through either bridge/router.

In this example, MAC addresses A and B are duplicated on NETBuilder C and NETBuilder D. In this environment, the end stations at the bottom of the figure must configure the address of the host. Half of the end stations can configure MAC address A as the address to the host, and the other end stations can configure MAC address B. Two backbone rings are required because you cannot have two stations with the same MAC address on the same ring.

Using source routing, the end stations send out a discovery packet for the host address (either address A or B). The discovery packets are bridged through both bridges (NETBuilder X and NETBuilder Y) to the backbone rings. The discovery packet flows on both backbone rings. NETBuilder C and D both respond. The workstation chooses the first path to respond. When the traffic on both bridges and rings is “load balanced,” if one bridge or ring goes down, the end stations can rediscover a new path to the host without reconfiguring.

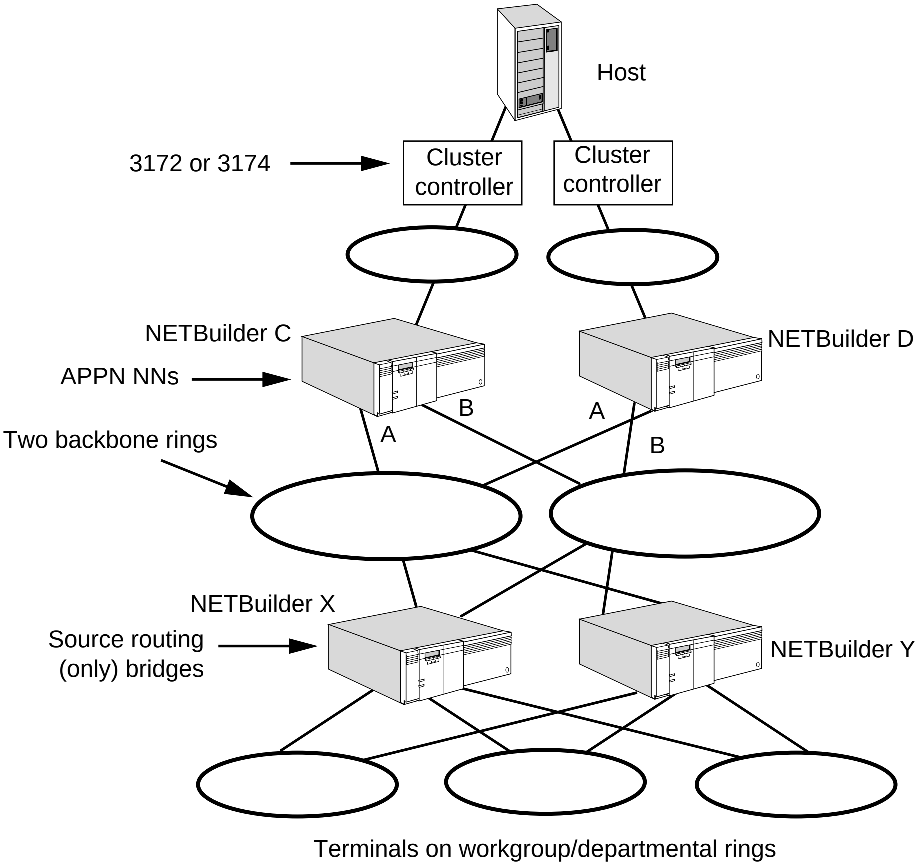


Figure 28-2 Using LAA for SNA Load Balancing

Using LAA with DECnet

Because both LAA and DECnet involve overwriting MAC addresses, you must be careful that any changed MAC addresses are not overwritten when you configure LAA and DECnet together. Depending on whether you configure LAA or DECnet first, you can overwrite a previously configured address. The difference in the results is as follows:

- If you configure LAA first on a path and then enable DECnet over that same path, the MAC address you configured using LAA will be overwritten by the DECnet address.
- If you enable DECnet first on a path and then try to reassign the MAC address of that path using LAA, you will be unable to reassign the MAC address because DECnet will not allow it.

If the paths go down, that may also affect which MAC address is being used.

For example, if LAA is configured first on path 4 and then DECnet is enabled over that same path, the following sequence of events may take place:

1 You reassign the MAC address on path 4 through LAA by entering:

```
SETDefault !4 -PATH MacAddress
```

2 If you or the configuration file then enables DECnet routing over path 4 by entering:

```
SETDefault !4 -DECnet CONTrol = ROute
```

the MAC address configured in the previous step is overwritten.

3 If path 4 goes down and comes back up, it still has the DECnet-configured address.

4 If you then disable DECnet by entering:

```
SETDefault !4 -DECnet CONTrol = NoRoute
```

the MAC address of path 4 defaults to the address burned on the adapter's PROM.

5 If path 4 goes down again and comes back up, the MAC address used is the address reassigned using LAA.

If DECnet is enabled first on path 4 and you then attempt to reassign the MAC address using LAA, the following sequence of events takes place:

1 You or the configuration file enables DECnet routing on the path by entering:

```
SETDefault !4 -DECnet CONTrol = ROute
```

2 You attempt to reassign the MAC address through LAA by entering:

```
SETDefault !4 -PATH MacAddress
```

Since DECnet is enabled, you are prevented from doing so, and you receive a warning message. The path continues to use the address configured through DECnet.

3 If path 4 goes down and then comes back up, the path still uses the DECnet-configured MAC address.

4 If you then disable DECnet by entering:

```
SETDefault !4 -DECnet CONTrol = NoRoute
```

the path now uses the MAC address burned in on the adapter's PROM.

5 If path 4 then goes down again and comes back up, the path continues to use the MAC address burned in on the adapter PROM.

29 CONFIGURING NETVIEW SERVICE POINT

This chapter describes how to configure a SSCP-PU session to a VTAM host. This feature supports SSCP-PU sessions to a VTAM host through PU4, PU5, and DLUr devices.

This software version does not provide logical unit (LU) support. If the NETBuilder bridge/router receives an ACTLU request over an SSCP-PU session, the bridge/router indirectly sends a negative response that indicates there are no active LUs in the router.

Configuring NetView Service Point

To configure NetView Service Point for SSCP-PU session support on the bridge/router, follow these steps:

1 Configure the local node name and node ID using:

```
SETDefault -SNA LocalNodeName <netid.cpname> <node_id>
```

The local node ID is the ID block (ID BLK) followed by the ID number (ID NUM). Although the local node can communicate with multiple SSCPs, you can only have one node ID for the local node. The ID BLK and ID NUM is an eight-digit value and, and the values must match those configured on the VTAM host. The default ID BLK is 05D. Because the netid value must match the VTAM host configuration, obtain netid from your systems programmer.

For example, to configure the local node name US3COMHQ.NB2SF020 and node ID 01724001 (the ID BLK is 017 and the ID NUM is 24001), enter:

`SETDefault -SNA LocalNodeName US3COMHQ.NB2SF020 01724001`

2 Define the SNA port definition using:

```
SETDefault !<port> -SNA PortDef = <DLC type>
  (LLC2|FR|PPP|DLSW|SDLC|UNdef) [ActLimit=<limit(1-16)>]
  [DatMode=(Half|Full)] [ROle=(Neg|Pri|Sec)]
```

With this command, you set the data link control (DLC) type and other attributes for the port. Specify LLC2 for token ring, Ethernet, FDDI, or Boundary Access Node (BAN) links. Specify PPP for PPP links. Specify FR for Frame Relay links for Boundary Network Node (BNN), DLSW for DLSw links, or SDLC for SDLC links.

If you specify DLSW as the DLC type, or if you specify LLC2 for BAN links only, you must specify the port number as !0.

For more information about the PortDef parameter, refer to Chapter 54 in *Reference for NETBuilder Family Software.*

3 If you set the port DLC type in step 2 to LLC2, FR, or DLSw, go to step a. If you set the port DLC type to SDLC in step 2, go to step b.

a Define the SSCP link station to a port using:

```
ADD !<port> -SNA SscpLinkSta <pu name> <dest media addr>
  [sap=<num>] [LinkName=name] [AutoStart=(Yes|No)]
  [Xid3=(Yes|No)]
```

Define a local PU that will use the link to communicate with the SSCP. If the DLC type specified with the PortDef parameter is LLC2 or DLSw, the destination address is a MAC address and SAP. If the DLC type is Frame Relay, the address will be a DLCI. If you specified !0 in the previous step for BAN links, specify !0 for the SscpLinkSta parameter. If you set the AutoStart value to No, then you must start the link station or initiate the link from the host side.

For more information about the PortDef parameter, refer to Chapter 54 in *Reference for NETBuilder Family Software*.

b Define the SSCP link station to a port over an SDLC line using:

```
ADD !<port> -SNA SDlcLinkSta <pu name> <station addr>(Hex 1-FE)
  [LinkName=name] [AutoStart =(Yes|No)] [Xid3=(Yes|No)]
  [SendWindow=<num>] [ContactTimer=<num>] [NoRspTimer=<num>]
  [NoRsptimRetry=<num>]
```

If the port DLC type set in step 2 is PPP, you do not need to specify a media address.

4 Repeat the previous step for each SSCP or SDLC link station added.

You can add up to 16 SSCP link stations or 16 SDLC link stations to a port.

5 If you configured multiple SSCP-PU sessions to different hosts, define the default PU using:

```
SETDefault -SNA DefaultPU <pu name>
```

The DefaultPU parameter is required when applications are added that support the sending of unsolicited ALERTS. The PU name must match one of the PU names defined with either the SscpLinkSta or SdlcLinkSta parameters.

6 Enable the SNA Service by entering:

```
SETDefault -SNA CONTrol = Enable
```

After this command has been enabled, the bridge/router can communicate with the VTAM host.

If you use DLSw, you need another route at the other end. Also note that the IDBLK and IDNUM must be the same on both hosts.

Figure 29-1 shows a sample configuration, and Table 29-1 lists the commands required to configure both the bridge/router and the VTAM hosts so that the SSCP-PU sessions can take place.

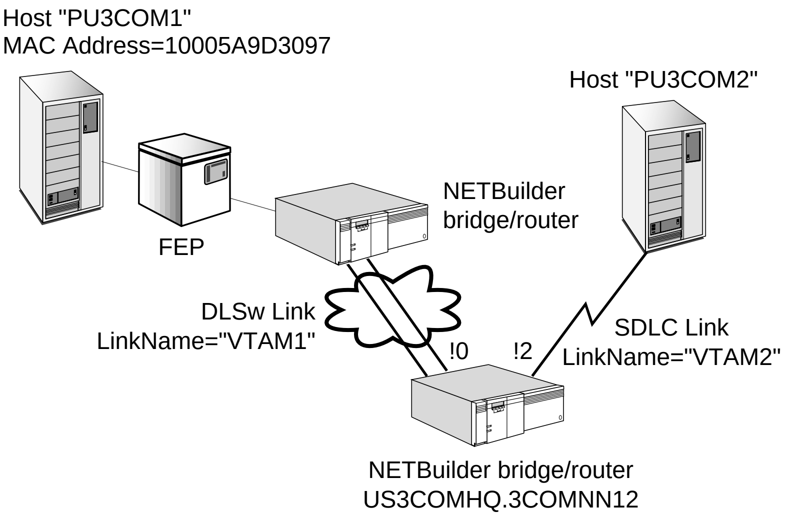


Figure 29-1 SSCP-PU Session Configuration

Table 29-1 SSCP-PU Session Configuration Commands

NETBuilder Bridge/Router	VTAM Host
SETDefault -SNA LocalNodeName US3COMHQ.3COMNB12 01724001	Configuration on VTAM1
SETDefault !0 -SNA PortDef = DLSW	US3COMHQ.PU3COM1
SETDefault !2 -SNA PortDef = SDLC DatMode=Full ROle=Neg	XID: IDBLK: 017 IDNUM: 24001
ADD !0 -SNA SscpLinkSta PU3COM1 10005A9D3097 LinkName=VTAM1 Xid3=Yes	
ADD !2 -SNA SdlcLinkSta PU3COM2 FE LinkName=VTAM2 Xid3=Yes	Configuration on VTAM2
SETDefault -SNA CONTrol = Enable	US3COMHQ.PU3COM2
	XID: IDBLK: 017 IDNUM: 24001

Activating and Deactivating SSCP Link Stations

You can dynamically activate and deactivate a link to a specific SSCP link station using:

```
SET -SNA LinkStaCONT = <linkname> Activate|Deactivate
```

For example, to deactivate the link LINK0004, enter:

```
SET -SNA LinkStaCONT = LINK0004 Deactivate
```

To reactivate that link, enter:

```
SET -SNA LinkStaCONT = LINK0004 Activate
```

To determine the link name (if assigned by the system), enter:

```
SHow -SNA LinkStaCONT
```

Activating and Deactivating All SSCP-PU Sessions

You can dynamically activate and deactivate the SNA Service, which affects all SSCP-PU sessions, using:

```
SET -SNA CONTrol = Enable | Disable
```

For example, to dynamically deactivate the SNA Service, which brings down all SSCP-PU sessions, enter:

```
SET -SNA CONTrol = Disable
```

To reactivate the SNA Service, enter:

```
SET -SNA CONTrol = Enable
```

After you reactivate the SNA Service, SSCP-PU sessions automatically come up only if AutoStart is set to Yes on the link stations.

30 Configuring Binary Synchronous Communications Connectivity

This chapter describes how to configure the bridge/router to provide Binary Synchronous Communications (BSC, also known as BISYNC) connectivity over DLSw networks. Using BSC pass-through, you can enable a secondary BSC control unit (CU) at a remote site to access a primary BSC device/host at a central site using a DLSw connection across the WAN.

BSC pass-through is supported only on selected models of the SuperStack II NETBuilder bridge/router and the OfficeConnect NETBuilder bridge/router. Also, BSC pass-through is supported on leased lines only.

This is a pass-through implementation only, which means that the bridge/routers only pass the BSC traffic through the larger network. The BSC host device and the BSC devices must be configured appropriately for the application.

The bridge/router supports the following BSC protocols:

- BSC 3270 protocol
- BSC 3780/2780 protocol (point-to-point only)

The BSC protocols supported can be used on the following IBM platforms:

- 3X74 controllers
- 3780/2780 RJE
- 3745 front end processors

This BSC implementation only supports EBCDIC versions of BSC.

BSC does not support local switching, which means BSC traffic cannot be received on one port and then transmitted out another port as pure BSC traffic. The BSC traffic must be transmitted out the second port over a DLSw connection.

Configuring BSC Pass-Through

To configure BSC pass-through for a single CU accessing a host, you need to configure BSC for the central site and remote site bridge/routers. The following procedure describes how to configure both.

For information on the BSC Service parameters used in these procedures, refer to Chapter 15 in *Reference for NETBuilder Family Software*.

Prerequisites

Before beginning these procedures, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/routers according to Chapter 1.

- Configure the DLSw connection over the WAN between the central site bridge/router and remote site bridge/router using the procedures described in Chapter 24 and Chapter 25.

Figure 30-1 shows a configuration in which a single CU at a remote site (secondary) is accessing a host (primary) at a central site using BSC that is transmitted across the WAN through a DLSw connection.

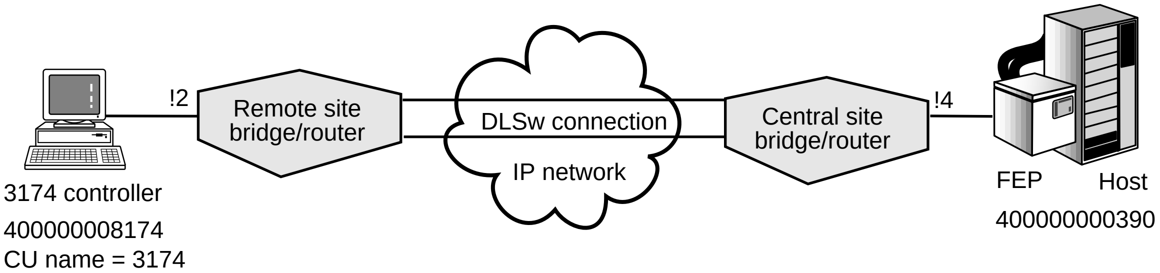


Figure 30-1 BSC Single Secondary CU Configuration

Remote Site Configuration

To configure the remote site bridge/router, follow these steps:

1 For the port connected to the BSC device, configure the port owner to BSC using:

```
SETDefault !<port> -PORT OWNer = BSC
```

2 Set internal clocking on the path using:

```
SETDefault !<path> -PATH CLock = Internal
```

Because the BSC device is a DTE, the bridge/router must act as a DCE and provide internal clocking, or alternatively, you can use modem eliminators to provide clocking. If you change the bridge/router from acting as a DTE to acting as a DCE, you must use a different cable. For more information, refer to the *WAN Cabling and Connectivity Guide*. You can find this guide on the 3Com Corporation World Wide Web site by entering: http://www.3com.com/.

3 Set the baud rate on the path using:

```
SETDefault !<path> -PATH BAud = <kbps> (0.110–16000)
```

Only the following baud rates are supported for BSC:

1.2	1.8	2.4
3.6	4.8	7.2
9.6	19.2	38.4

The baud rates set for both the local and remote site bridge/routers must match or be close so that the BSC host will receive responses in a timely manner. For more information, refer to “Baud Rate and Line Speed Considerations” on page 30-5.

4 Set the path line type for leased using:

```
SETDefault !<path> -PATH LineType = Leased
```

BSC pass-through is supported on leased lines only.

5 Toggle the path using:

```
SETDefault !<path> -PATH CONTrol = Enable
```

Toggling the path enables the clocking and baud rate settings to take effect.

Other PATH Service parameters such as DUplex, ENCoding, and TxIdle do not need to be configured, and their values are ignored by BSC.

For information about parameters in the PATH Service, refer to Chapter 42 in *Reference for NETBuilder Family Software*.

6 Set the port as a BSC primary device using:

```
SETDefault !<port> -BSC Role = Primary
```

The port on the remote site bridge/router must be set to Primary because the BSC device(s) are always secondary devices.

For example, to configure port 2 as the BSC primary, enter:

```
SETDefault !2 -BSC Role = Primary
```

CAUTION: *Do not configure both the remote site and central site as primary. If both sides of the BSC link are configured to primary, neither side will initiate the DLSw circuit, and no BSC traffic will be sent.*

7 Enable BSC on the port using:

```
SETDefault !<port> -BSC CONTrol = Enable
```

8 Define the BSC CU that represents the BSC controller and enable it using:

```
ADD !<port> -BSC BscCU <cu name> <cu addr> <local mac> <remote
  mac> [Lsap=<value>] [Rsap=<value>] [ENable]
```

The name can be up to 8 characters long, must be unique on the bridge/router, and it cannot be the name "ALL." The name is not case-sensitive. The CU name is used to define the CU information on the bridge/router, and is also used to disable and enable the CU for modifying the CU definition (refer to "Modifying Existing BSC CU Definitions" on page 30-5).

The CU address must be between 0 and 31. The MAC addresses must be entered in noncanonical format and must be in the valid LAA range (refer to Chapter 28).

For example, to define the CU named "3174" on port 2 with a CU address of 10 and local MAC address of 400000003174 and a remote MAC address of 400000000390, and use the default SAP values, enter:

```
ADD !2 -BSC BscCU 3174 10 400000003174 400000000390 ENable
```

The specified CU at the remote site is ready for the BSC connection, which can take place after BSC is configured on the central site bridge/router.

Central Site Configuration

To configure BSC on the central site bridge/router, follow these steps:

1 Configure the port attached to the front-end processor (FEP) for BSC using:

```
SETDefault !<port> -PORT OWNer = BSC
```

2 Set external clocking on the path using:

```
SETDefault !<path> -PATH CLock = External
```

Set the path to External clocking because most FEPs provide clocking. Consult your systems programmer to verify that the FEP provides clocking on the line. If the FEP does not provide clocking, set the -PATH CLock parameter to Internal and use the appropriate cables for the bridge/router and the FEP; if you use internal clocking, you must also set the baud rate using the SETDefault !<path> -PATH BAud command.

3 Set the path line type for leased using:

```
SETDefault !<path> -PATH LineType = Leased
```

BSC pass-through is supported on leased lines only.

4 Toggle the path using:

```
SETDefault !<path> -PATH CONTrol = Enable
```

Toggling the path enables the clocking and connector parameters to take effect.

5 Set the port as a BSC secondary device using:

```
SETDefault !<port> -BSC Role = Secondary
```

The port on the central site bridge/router must be set to Secondary because the FEP or host is always a primary device.

For example, to set port 4 as the BSC secondary, enter:

`SETDefault !4 -BSC Role = Secondary`

CAUTION: *Do not configure both the central site and remote site as secondary. If both sides of the BSC link are configured to secondary, neither side will initiate the DLSw circuit, and no BSC traffic will be sent.*

6 Enable BSC on the port using:

```
SETDefault !<port> -BSC CONTrol = Enable
```

7 Define the BSC CU that represents the CU at the remote site and enable it using:

```
ADD !<port> -BSC BscCU <cu name> <cu addr> <local mac> <remote
  mac> [Lsap=<value>] [Rsap=<value>] [ENable]
```

The MAC addresses must be entered in noncanonical format and must be in the valid LAA range (refer to Chapter 28).

The local and remote MAC addresses should be the reverse of the local and remote MAC addresses entered in step 7 on page 30-3 in the remote site bridge/router procedure. If you define the local and remote SAP values, you should also enter the reverse SAP values that you configured on the remote site.

For example, to define and enable a CU named “3174” on port 4 with a CU address of 10 and local MAC address of 400000000390 and a remote MAC address of 400000003174, enter:

`ADD !4 -BSC BscCU 3174 10 400000000390 400000003174 ENable`

The BscCU definition entered here refers to the device at the remote site, even though it is added on the host port.

The remote site BSC can access the central site host (assuming that the DLSw connection across the WAN is correctly configured).

8 Repeat step 7 for each CU you will connect to at the central site.

Baud Rate and Line Speed Considerations

Because BSC is a time-sensitive protocol, you should be careful when configuring the DLSw connection baud rate so that BSC traffic can be effectively transmitted across the network. Note the following considerations when configuring BSC:

- The baud rate configured for the central site BSC link should match the baud rate for the remote site BSC link or be close to it. If there is a wide variance between the baud rates on both bridge/routers, BSC transmission errors and time-outs can occur.
- The baud rate configured for the DLSw connection across the WAN must be higher than the baud rate configured for the BSC links. To prevent BSC session time-outs, follow these guidelines:
  - If you are running only BSC traffic across the DLSw connection, the baud rate for the DLSw connection across the WAN must be higher than the baud rate for the corresponding BSC links.
  - If you are running BSC traffic with other IBM traffic types, such as SDLC and LLC2, over the same DLSw connection, you must increase the link speed of the DLSw connection and use DLSw prioritization to prioritize the BSC traffic higher than the non-BSC traffic. For more information about DLSw prioritization, refer to Chapter 24.
  - If you are running BSC traffic with non-IBM traffic types such as IPX over the same DLSw connection, you must increase the link speed of the DLSw connection and you must use either protocol reservation or data prioritization to prioritize the BSC traffic higher than the other traffic types. You must set BSC traffic to a higher priority so that the BSC traffic will maintain enough speed to ensure proper request/response between the BSC devices. For more information about protocol reservation, refer to Chapter 38. For more information about data prioritization, refer to Chapter 41.

Modifying Existing BSC CU Definitions

After you have added and defined a BSC CU, you can modify the CU definition. You must first disable the CU using:

```
SETDefault !<CU name> -BSC CUCONTrol = Disable
```

To modify an existing BSC CU definition, use:

```
ADD !<port> -BSC BscCU <cu name> <cu addr> <local mac> <remote
  mac> [Lsap=<value>] [Rsap=<value>] [ENable]
```

You can enable the CU using the ENable option when you change the definition. If you do not want to enable the CU at that time, do not specify ENable. You can reenable the CU at another time without entering the definition using:

```
SETDefault !<CU name> -BSC CUCONTrol = Enable
```

You can delete a single defined CU or all defined CUs on a port using:

```
DELete !<port> -BSC BscCU <cu name> | ALL
```

You can only delete CUs that have been disabled with the CUCONTrol parameter.

BSC Configuration Examples

This section provides two BSC configuration examples, one for configuring a single CU on a remote site, and one for configuring multiple CUs at a remote site.

Example 1: CU At Single Remote Site

Figure 30-2 is a BSC configuration example for a single CU at the remote site. Table 30-1 lists the commands necessary to configure BSC on both the remote site and central site bridge/routers.

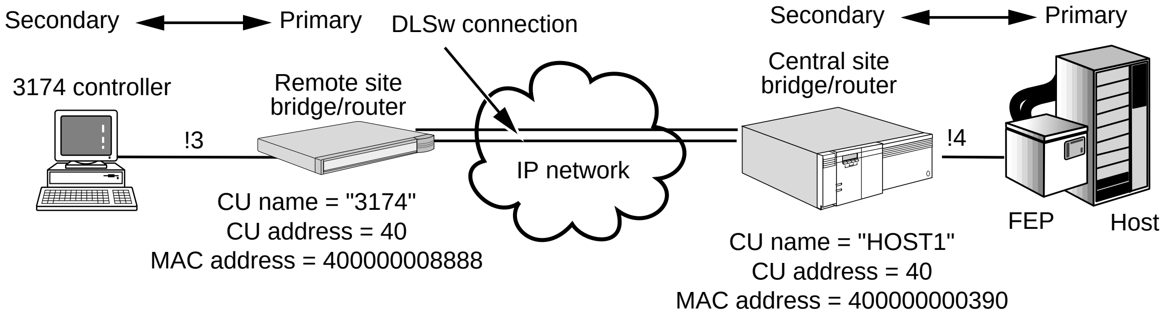


Figure 30-2 BSC Configuration Example (Single CUs)

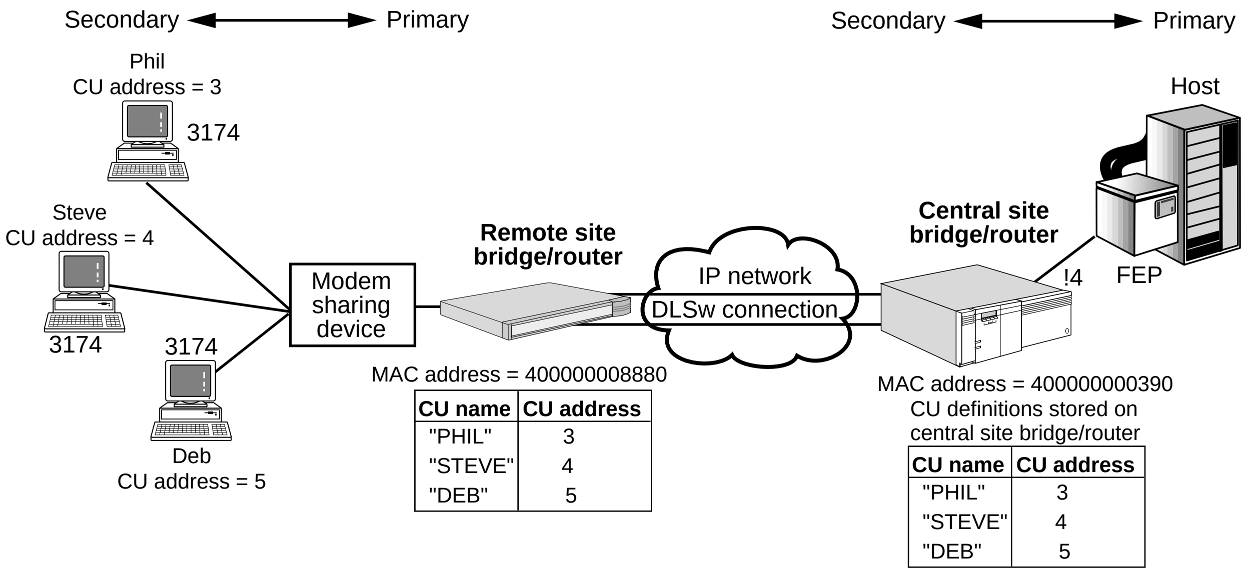
Table 30-1 BSC Configuration Example Commands (Single CU)

Commands Entered at Remote Site	Commands Entered at Central Site
SETDefault !3 -PORT OWNer = BSC	SETDefault !4 -PORT OWNer = BSC
SETDefault !3 -PATH CLock = Internal	SETDefault !4 -PATH CLock = External
SETDefault !3 -PATH BAud = 9.6	SETDefault !4 -PATH BAud = 9.6
SETDefault !3 -PATH LineType = Leased	SETDefault !4 -PATH LineType = Leased
SETDefault !3 -PATH CONTrol = Enable	SETDefault !4 -PATH CONTrol = Enable
SETDefault !3 -BSC Role = Primary	SETDefault !4 -BSC Role = Secondary
SETDefault !3 -BSC CONTrol = Enable	SETDefault !4 -BSC CONTrol = Enable
ADD !3 -BSC BscCU 3174 40 400000008888 400000000390 ENable	ADD !4 -BSC BscCU 3174 40 400000000390 400000008888 ENable

Certain commands shown in the table use the port ID mapped to the path ID. On most bridge/router models, the port and path numbers are mapped one-to-one. On bridge/router models with ISDN interfaces, the default path number mapped to the port number is one number different; for example path 4 is mapped to port 3.

Example 2: Multiple CUs On One Port at a Remote Site

Figure 30-3 is a BSC configuration example for multiple CUs on one remote site port. Table 30-2 lists the commands necessary to configure BSC on both the remote site and central site bridge/routers.


Secondary
Primary
Secondary
Primary
Phil
CU address = 3
3174
Steve
CU address = 4
3174
3174
Deb
CU address = 5
Modem sharing device
Remote site bridge/router
MAC address = 400000008880
IP network
DLSw connection
Central site bridge/router
!4
FEP
Host
MAC address = 400000000390
CU definitions stored on central site bridge/router

CU name	CU address
"PHIL"	3
"STEVE"	4
"DEB"	5

CU name	CU address
"PHIL"	3
"STEVE"	4
"DEB"	5

Figure 30-3 BSC Configuration Example (Multiple CUs on One Port)

Table 30-2 BSC Configuration Example Commands (Multiple CUs on One Port)

Commands Entered at Remote Site	Commands Entered at Central Site
SETDefault !3 -PORT OWNer = BSC	SETDefault !4 -PORT OWNer = BSC
SETDefault !3 -PATH CLock = Internal	SETDefault !4 -PATH CLock = External
SETDefault !3 -PATH BAud = 9.6	SETDefault !4 -PATH BAud = 9.6
SETDefault !3 -PATH LineType = Leased	SETDefault !4 -PATH LineType = Leased
SETDefault !3 -PATH CONTrol = Enable	SETDefault !4 -PATH CONTrol = Enable
SETDefault !3 -BSC Role = Primary	SETDefault !4 -BSC Role = Secondary
SETDefault !3 -BSC CONTrol = Enable	SETDefault !4 -BSC CONTrol = Enable
ADD !3 -BSC BscCU PHIL 3 400000008880 400000000390 ENable	ADD !4 -BSC BscCU PHIL 3 400000000390 400000008880 ENable
ADD !3 -BSC BscCU STEVE 4 400000008880 400000000390 ENable	ADD !4 -BSC BscCU STEVE 4 400000000390 400000008880 ENable
ADD !3 -BSC BscCU DEB 5 400000008880 400000000390 ENable	ADD !4 -BSC BscCU DEB 5 400000000390 400000008880 ENable

Example 3: CUs at Multiple Remote Sites

Figure 30-4 is a BSC configuration example for a virtual multidrop environment (multiple remote sites, each with one CU). Table 30-3 lists the commands necessary to configure BSC on both the remote site and central site bridge/routers.

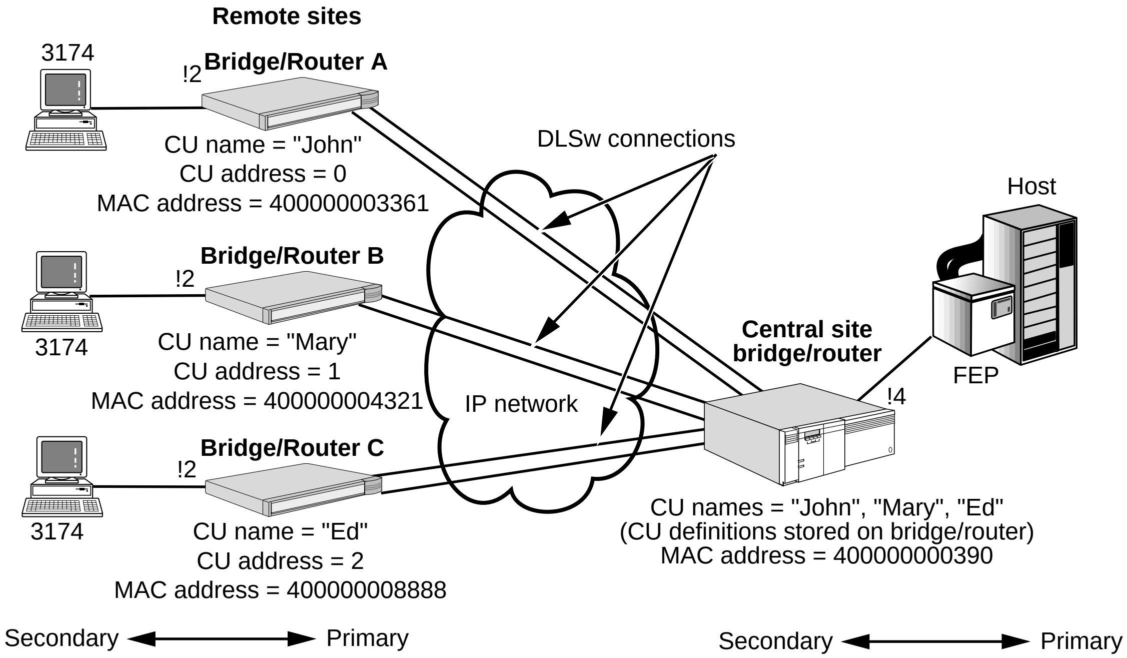


Figure 30-4 BSC Virtual Multidrop Configuration Example

Table 30-3 BSC Configuration Example Commands (Multiple CUs)

Commands Entered at Remote Sites	Commands Entered at Central Site
Remote Site A:	
SETDefault !2 -PORT OWNer = BSC	SETDefault !4 -PORT OWNer = BSC
SETDefault !2 -PATH CLock = Internal	SETDefault !4 -PATH CLock = External
SETDefault !2 -PATH BAud = 9.6	SETDefault !4 -PATH LineType = Leased
SETDefault !2 -PATH LineType = Leased	SETDefault !4 -PATH CONTrol = Enable
SETDefault !2 -PATH CONTrol = Enable	SETDefault !4 -BSC Role = Secondary
SETDefault !2 -BSC Role = Primary	SETDefault !4 -BSC CONTrol = Enable
SETDefault !2 -BSC CONTrol = Enable	ADD !4 -BSC BscCU JOHN 0 400000000390 400000003361 ENable
ADD !2 -BSC BscCU JOHN 0 400000003361 400000000390 ENable	ADD !4 -BSC BscCU MARY 1 400000000390 400000004321 ENable
Remote Site B:	ADD !4 -BSC BscCU ED 2 400000000390 400000008888 ENable
SETDefault !2 -PORT OWNer = BSC	
SETDefault !2 -PATH CLock = Internal	
SETDefault !2 -PATH BAud = 9.6	

(continued)

Table 30-3 BSC Configuration Example Commands (Multiple CUs) (continued)

Commands Entered at Remote Sites	Commands Entered at Central Site
SETDefault !2 -PATH LineType = Leased	
SETDefault !2 -PATH CONTrol = Enable	
SETDefault !2 -BSC Role = Primary	
SETDefault !2 -BSC CONTrol = Enable	
ADD !2 -BSC BscCU MARY 1 400000004321 400000000390 ENable	
Remote Site C:	
SETDefault !2 -PORT OWNer = BSC	
SETDefault !2 -PATH CLock = Internal	
SETDefault !2 -PATH BAud = 9.6	
SETDefault !2 -PATH LineType = Leased	
SETDefault !2 -PATH CONTrol = Enable	
SETDefault !2 -BSC Role = Primary	
SETDefault !2 -BSC CONTrol = Enable	
ADD !2 -BSC BscCU ED 2 400000008888 400000000390 ENable	

31 CONFIGURING POLLED ASYNCH CONNECTIVITY

This chapter describes how to configure the bridge/router to provide polled asynchronous communications (referred to as *asynch* in the remainder of this chapter) so that remote asynch devices can communicate with an asynch polling host. The bridge/router offers transparent transmission of asynch "pass-through" across the WAN.

Polled asynch connectivity is supported only on selected models of the SuperStack II NETBuilder bridge/router and the OfficeConnect NETBuilder bridge/router. Polled asynch is only supported on RS232 ports.

Some parameters provided can be used to configure the tunnel framing appropriate for the asynchronous protocol and devices being used. These protocols are vendor-specific. For more information about the specific asynchronous protocol being used, check your vendor's documentation.

Configuring Asynch Tunnels on Both Central and Remote Sites

To configure polled asynch connectivity, you need to configure asynch for the local and the remote bridge/router. The following procedure describes how to configure both bridge/routers.

For information on the ATUN Service parameters used in these procedures, refer to Chapter 9 in *Reference for NETBuilder Family Software*.

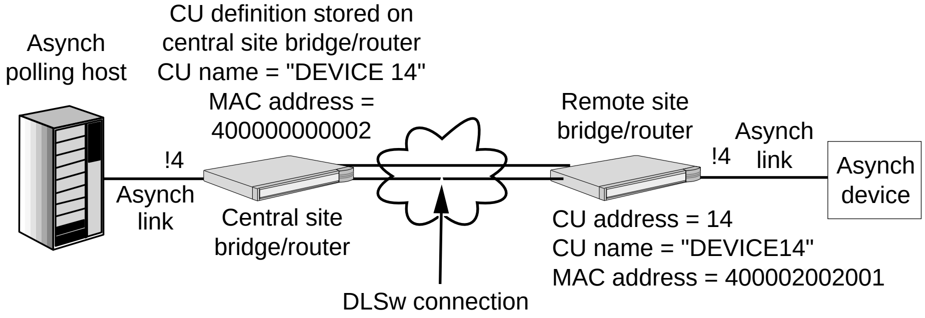
Prerequisites

Before beginning these procedures, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the ports and paths of your bridge/routers according to Chapter 1.
- Configure the DLSw connection over the WAN between the central site bridge/router and remote site bridge/routers using the procedures described in Chapter 24 and Chapter 25.

Figure 31-1 shows a configuration in which a single asynch device CU at a remote site is communicating with an asynch polling host at a central site with a DLSw connection across the WAN. Note that you can have multiple CUs at a central site, but only one CU on a remote site port.

This example provides a procedure for a simple configuration. For more specific configuration examples, refer to "Asynch Tunneling Configuration Examples" on page 31-7.



Asynch polling host
CU definition stored on central site bridge/router
CU name = "DEVICE 14"
MAC address = 400000000002
!4
Asynch link
Central site bridge/router
Remote site bridge/router
!4
Asynch link
Asynch device
CU address = 14
CU name = "DEVICE14"
MAC address = 400002002001
DLSw connection

Figure 31-1 Asynch Configuration with a Single CU

The configuration procedure is organized by the following major tasks (some parameters may be optional):

- General asynch port and path configuration:

  Set the -Port OWNer parameter to ATUN, and configure the -PATH BAud, DataBits, StopBits, and PARity parameters, and re-enable the -PATH CONtrol parameter

- Asynch port configuration:

  Configure the -ATUN FrameSize, FrameChars, IdleTimer, FrameGap, PortCONTrol, CUAddress, and AddrLOCation parameters

- Asynch CU configuration:

  Configure the -ATUN LocalMac, LocalSap, RemoteMac, and RemoteSap parameters, and enable the -ATUN CUCONTrol parameter.

The procedures are performed on both the central site and remote site bridge/routers.

General Asynch Port and Path Configuration

The first major task is to configure asynch support on port and paths. To configure asynch port and path support, perform the following steps on both the central and remote site bridge/routers:

1 For the port connected to the asynch device, configure the port owner to ATUN by entering:

```
SETDefault !<port> -PORT OWNer = ATUN
```

2 Set the baud rate on the path using:

```
SETDefault !<path> -PATH BAud = <kbps> (0.110–16000)
```

Only the following baud rates are supported for asynch:

0.110	0.135	0.150	0.200	0.300
0.600	1.2	1.8	2.4	3.6
4.8	7.2	9.6	19.2	38.4

When entering a baud rate less than 1.0, you must enter the leading 0 before the decimal point (for example, 0.110, not .110). For baud rates with a trailing zero (for example, 0.150), you do not have to enter the trailing 0.

For example, to set the baud rate to 0.600 on path 4, enter:

```
SETDefault !4 -PATH BAud = 0.6
```

The baud rate configured on the bridge/router must be consistent with the baud rate configured on the asynch device.

3 Configure the transmission characteristics of the asynch path by performing the following sub-steps:

a Configure the number of data bits in each character transmitted or received on the asynch path using:

```
SETDefault !<path> -PATH DataBits = 5 | 6 | 7 | 8
```

b Configure the number of stop bits appended to each character transmitted on the asynch path using:

```
SETDefault !<path -PATH StopBits = 1 | 1.5 | 2
```

c Configure how you want the parity bits appended to each transmitted or received character using:

```
SETDefault !<path> -PATH PARity = Even | Odd | Mark | Space |
  None
```

Using this parameter, you specify whether the parity bit is appended to make the total parity even or odd. Or, you can specify whether the parity bit appended is always 1 (Mark), or 0 (Space). This parameter applies to both transmitted and received characters. To configure different values for transmitted and received characters, use the -PATH RxParity and -PATH TxParity features. For more information about these parameters, refer to Chapter 42 in *Reference for NETBuilder Family Software*.

4 Toggle the path using:

```
SETDefault !<path> -PATH CONTrol = Enable
```

Toggling the path enables the baud rate and the other -PATH Service parameters to take effect.

Proceed to the next section.

Asynch Port Configuration

The next steps determine how the bridge/router groups received data into frames for transmission, and how these frames are routed to asynch tunnels. The correct setting of these -ATUN Service parameters will depend on the specific asynch protocol you are using. For more information about parameters in the ATUN Service, refer to Chapter 9 in *Reference for NETBuilder Family Software*.

To configure asynch ports, follow these steps:

1 Configure how the incoming character stream is broken into frames by performing the following steps:

a Configure the maximum number of bytes to be collected before forwarding using:

```
SETDefault !<port> -ATUN FrameSize = <bytes> (1-1024)
```

This parameter may be used to reduce latency for an application with a fixed frame size by saving the wait for IdleTimer expiration.

b Configure the special characters that will indicate the end of a frame using:

```
SETDefault !<port> -ATUN FrameChars <char>...
```

c Configure the length of idle time interval that will cause accumulated data to be forwarded as a frame using:

```
SETDefault !<port> -ATUN IdleTimer = <milliseconds> (0-5000)
```

d Configure the minimum amount of idle time to leave between frames transmitted by the bridge/router using:

```
SETDefault !<port> -ATUN FrameGap = <milliseconds> (0-1000)
```

If the FrameGap parameter is configured, the bridge/router separates the frames before sending them out the port.

2 To configure how the asynch port will be used, perform the following steps:

a On the central site, configure the asynch port using:

```
SETDefault !<port> -ATUN PortCONTrol = ([Enabled | Disabled],
  [CentralSite | RemoteSite], [Address | NoAddress], [BCAddr |
  NoBCaddr], [ForcePoll| NoForcePoll], [TestEcho | NoTestEcho])
```

Specify CentralSite and Enabled to enable the asynch port. When you configure the asynch port for the central site, you can set the port to provide addressing using the Address value.

For example, to enable asynch on port 4 and set it for central site operation and to enable addressing on the port, enter:

```
SETDefault !4 -ATUN PortCONTrol = Enabled, CentralSite, Address
```

Although addressing is not required on the central site, it is recommended where possible. Note that the addressing parameters may need to be different at opposite ends of the tunnel.

If you choose to use addressing, you can choose whether to use the address specified with the -ATUN BroadCastAddr parameter by specifying the BCAddr value.

b On the remote site, configure the asynch port using:

```
SETDefault !<port> -ATUN PortCONTrol = ([Enabled | Disabled],
  [CentralSite | RemoteSite], [Address | NoAddress], [BCAddr |
  NoBCaddr], [ForcePoll| NoForcePoll], [TestEcho | NoTestEcho])
```

Specify RemoteSite and Enabled to enable the asynch port. For example, to enable asynch port 4 and set it for remote site operation and for no addressing (recommended for remote sites), enter:

```
SETDefault !4 -ATUN PortCONTrol = Enabled, RemoteSite,
  NoAddress
```

The NoAddress setting specifies that each frame is sent on every asynch tunnel. A remote site port allows only a single CU (tunnel), so addressing is generally not necessary. By not configuring specific addresses on the remote site, the remote site configuration is simpler than the central site configuration.

For more information about the PortCONTrol parameter, refer to Chapter 9 in *Reference for NETBuilder Family Software*.

3 If you specified the Address value in the previous step (normally for central sites only), configure addressing by performing the following sub-steps:

a Configure the address location using:

```
SETDefault !<port> -ATUN AddrLOCation = <offset> (0-1024)
```

This parameter specifies which data byte of received frames should be considered an address byte. The value is specified as an offset from the first byte of the frame.

For example, to set the offset on port 4 to 1, enter:

```
SETDefault !4 -ATUN AddrLOCation = 1
```

b If you specified the BCAddr value in the PortCONTrol parameter, configure the broadcast address using:

```
SETDefault !<port> -ATUN BroadCastAddr = <value> (0-255)
```

The address you enter is a special value for the address byte and indicates an "all stations" destination. All frames whose address byte (as specified with AddrLOCation) matches the value of the broadcast address are forwarded to all active asynch tunnels on the port.

For example, to configure the broadcast address for port 4 to 255, enter:

```
SETDefault !4 -ATUN BroadCastAddr = 255
```

Proceed to the next section.

Asynch CU Configuration

The following steps define CUs on the bridge/router to provide the definition of the CUs, which determine the tunnel connection(s) to and from remote sites. To configure asynch CUs, follow these steps:

1 Define the name of each CU and assign each one to the port using:

```
ADD !<port> -ATUN PortCU <CU name>...
```

Each CU name can be up to 8 characters long and must be unique on the bridge/router. Use the CU name(s) to configure the remaining parameters in the central site procedure.

The difference between defining CUs on a central site and a remote site is:

- On a central site port, you can define multiple CUs, each representing a tunnel to a remote site.
- On a remote site port, you can define a single CU, which provides a tunnel endpoint to the central site.

For example, to define the CU name "DEVICE14" on port 4, enter:

```
SETDefault !4 -ATUN PortCU DEVICE14
```

You can define multiple CUs on a central site using one command. For example, if there are two other remote site CUs named DEVICE15 and DEVICE16, you can define all three on port 4 by entering:

```
ADD !4 -ATUN PortCU DEVICE14 DEVICE15 DEVICE16
```

2 Define the CU address using:

```
SETDefault !<CU name> -ATUN CUADDRess = <value>(0-255)
  [-<value>(0-255)]
```

For example, to assign a CU address of 1 to a CU named HOST1, enter:

```
SETDefault !HOST1 -ATUN CUADDRess = 1
```

If you have multiple physical devices, multidropped at a remote site, only a single tunnel is configured to a remote port using a single CU definition. In this case, you can use an address range to route frames for all the devices into the same tunnel. For example, if you have four CUs with CU addresses of 15, 16, 17, and 18, you can specify a CU address range of 15-18 and assign it to a CU name of DEVICE15 by entering:

```
SETDefault !DEVICE15 -ATUN CUADDRess = 15-18
```

When addressing is used on the port (as set with the PortCONTrol parameter), an address byte is extracted from each frame (set with the AddrLOCation parameter). The frame is then directed to the CU whose address range includes this value.

The CUADDress values for all enabled CUs on a port cannot overlap; an addressed frame is mapped to a single CU only.

3 Configure the peer MAC addresses for the tunnel endpoints by performing the following steps on both the central site and remote site bridge/routers:

a On the central site bridge/router, configure the MAC address of the CU on the central site that will be used as the source address for initiating a DLSw circuit or LLC2 session using:

```
SETDefault !<CU name> -ATUN LocalMac = <address>
```

By default, the address is configured in noncanonical format and must be in the valid LAA range (refer to Chapter 28). You can configure the address in canonical format by entering the prefix “mac” or “cmac” before the address. The MAC address entered for the LocalMac parameter must be unique; the same MAC address cannot be used as the value for the LocalMac parameter on the same asynch network.

For example, to configure the local MAC address for the CU named DEVICE14, enter:

```
SETDefault !DEVICE14 -ATUN LocalMac = 400000000002
```

The MAC address you enter as the LocalMac is the same address you will enter in step d as the RemoteMac address on the remote site bridge/router.

b On the central site bridge/router, configure the MAC address of the CU on the remote site that will be used as the peer address for the DLSw circuit or LLC2 session using:

```
SETDefault !<CU name> -ATUN RemoteMac = <address>
```

By default, the address is configured in noncanonical format The same restrictions for LocalMac described in the previous step also apply to the RemoteMac parameter.

For example, to configure the MAC address of the remote site CU, enter:

```
SETDefault !DEVICE14 -ATUN RemoteMac = 400002002001
```

c On the remote site bridge/router, configure the MAC address of the CU on the remote site that will be used as the source address for initiating a DLSw circuit or LLC2 session using:

```
SETDefault !<CU name> -ATUN LocalMac = <address>
```

Use the same MAC address that you configured in step b. For example, to configure the local MAC address as 400002002001 for the CU named DEVICE14, enter:

```
SETDefault !DEVICE14 -ATUN LocalMac = 400002002001
```

d. On the remote site bridge/router, configure the MAC address of the CU on the central site that will be used as the peer address for the DLSw circuit or LLC2 session using:

```
SETDefault !<CU name> -ATUN RemoteMac = <address>
```

Use the same MAC address you configured in step a. For example, to configure the remote MAC address as 400000000002 for the CU named DEVICE14, enter:

```
SETDefault !DEVICE14 -ATUN RemoteMac = 400000000002
```

4. Optionally, configure the peer SAP values for both tunnel peers by performing the following substeps on both the central site and remote site bridge/routers:

a. On the central site bridge/router, configure the local SAP as the source SAP for initiating a DLSw circuit for tunneling asynch data using:

```
SETDefault !<CU name> -ATUN LocalSap = <sap> (hex 04-ec[by 4])
```

For example, to configure the local SAP as 04 for DEVICE14, enter:

```
SETDefault !DEVICE14 -ATUN LocalSap = 04
```

b. On the central site bridge/router, configure the remote SAP as the destination SAP for initiating a DLSw circuit for tunneling asynch data using:

```
SETDefault !<CU name> -ATUN RemoteSap = <sap> (hex 04-ec[by 4])
```

For example, to configure the remote SAP as 04 for DEVICE14, enter:

```
SETDefault !DEVICE14 -ATUN RemoteSap = 04
```

c. Repeat steps a and b on the remote site bridge/router, reversing the SAP values entered at the central site. For example, if the SAP values are different, enter the LocalSap value entered on the central site as the RemoteSap value on the remote site, and vice-versa.

5. Activate the asynch connection to the CU using:

```
SETDefault !<CU name> -ATUN CUCONTrol = (Enabled | Disabled)
```

After you enter this parameter, the asynch connection will be ready to accept or initiate sessions.

For example, to activate an asynch connection to the CU named DEVICE14, enter:

```
SETDefault !DEVICE14 -ATUN CUCONTrol = Enabled
```

Repeat this step on both the central site and remote site bridge/routers.

6. Repeat steps 2 through 5 for each CU name defined in step 1.

Asynch Tunneling Configuration Examples

This section provides two asynch tunneling configuration examples. The first example shows a central site communicating with three remote sites, each with a single asynch device. The second example shows a central site communicating with two remote sites, each with different and more complex asynch configurations.

The specific settings of the -PATH Service parameters depend on the devices being used. Similarly the -ATUN framing and addressing parameters must be appropriate for the devices and the protocols in use. These examples describe the characteristics of a hypothetical polling protocol.

Example 1: Single Asynch Devices at the Remote Sites

Figure 31-2 shows a configuration with a central site communicating with three remote sites, each with a single asynch device. Table 31-1 lists the commands to configure the asynch tunneling at the central site and for the three remote sites.

For this example, the asynch devices are operating at 9600 baud, using 8 databits and even parity, and needs one stop bit. The host sends fixed-size (4-byte) poll frames with an address in the third byte of every frame (no broadcast), and the devices send variable-sized frames (80-byte maximum) in response to a poll. Both the asynch host and the asynch devices send characters back-to-back within a frame, but delay at least 30 milliseconds between sending frames; they also recognize a received frame by seeing a delay of at least 10 milliseconds.

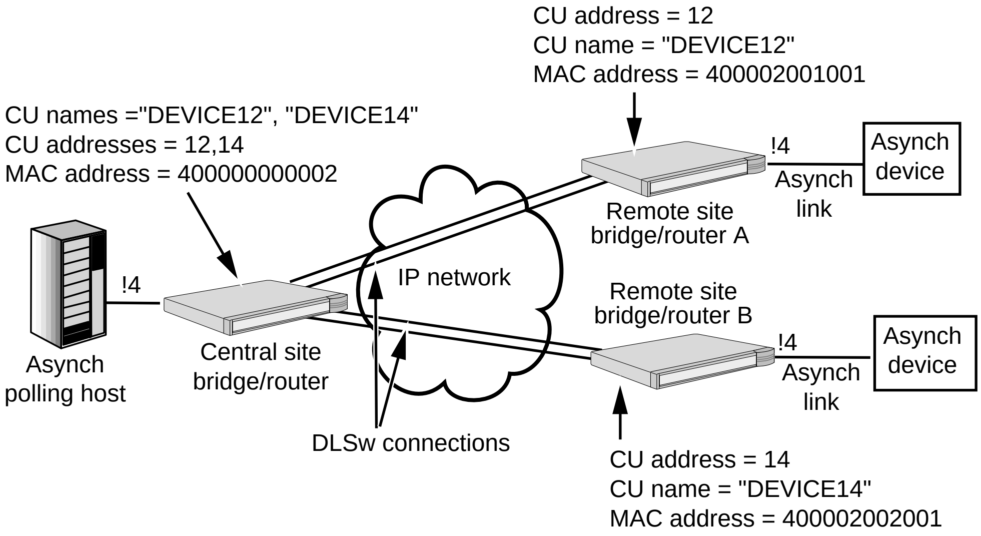

CU address = 12
CU name = "DEVICE12"
MAC address = 400002001001
CU names ="DEVICE12", "DEVICE14"
CU addresses = 12,14
MAC address = 400000000002
!4
Asynch device
Asynch link
Remote site bridge/router A
IP network
!4
Asynch polling host
Central site bridge/router
Remote site bridge/router B
!4
Asynch device
Asynch link
DLSw connections
CU address = 14
CU name = "DEVICE14"
MAC address = 400002002001

Figure 31-2 Asynch Configuration Example (Single CU at Remote Sites)

Table 31-1 Asynch Configuration Example Commands (Single CU at Each Remote Site)

Commands Entered at Central Site:	Commands Entered at Remote Sites:
	Remote Site A:
SETDefault !4 -PORT OWNer = ATUN	SETDefault !4 -ATUN PORT OWNer = ATUN
SETDefault !4 -PATH BAud = 9.6	SETDefault !4 -ATUN PATH BAud = 9.6
SETDefault !4 -PATH DataBits = 8	SETDefault !4 -PATH DataBits = 8
SETDefault !4 -PATH PARity = Even	SETDefault !4 -PATH PARity = Even
SETDefault !4 -PATH StopBits = 1	SETDefault !4 -PATH StopBits = 1
SETDefault !4 -PATH CONTrol = Enable	SETDefault !4 -ATUN PATH CONTrol = Enable
SETDefault !4 -ATUN FrameSize = 4	SETDefault !4 -ATUN FrameSize = 80
SETDefault !4 -ATUN IdleTimer = 10	SETDefault !4 -ATUN IdleTimer = 10
SETDefault !4 -ATUN FrameGap = 10	SETDefault !4 -ATUN FrameGap = 10
SETDefault !4 -ATUN PortCONTrol = (Enabled, CentralSite, Address, NoBCAddr, NoForcePoll, NoTestEcho)	SETDefault !4 -ATUN PortCONTrol = (Enabled, RemoteSite, NoAddress, NoBCAddr, NoForcePoll, NoTestEcho)
SETDefault !4 -ATUN AddrLOCation = 2	ADD !4 -ATUN PortCU = DEVICE12
ADD !4 -ATUN PortCU = DEVICE12 DEVICE14	SETDefault !DEVICE12 -ATUN CUADDRess = 12
SETDefault !DEVICE12 -ATUN CUADDRess = 12	SETDefault !DEVICE12 -ATUN LocalMac = 400002001001

(continued)

Table 31-1 Asynch Configuration Example Commands (Single CU at Each Remote Site) (continued)

Commands Entered at Central Site:	Commands Entered at Remote Sites:
SETDefault !DEVICE12 -ATUN LocalMac = 400000000002	SETDefault !DEVICE12 -ATUN RemoteMac = 400000000002
SETDefault !DEVICE12 -ATUN RemoteMac = 400002001001	SETDefault !DEVICE12 -ATUN CUCONTrol = Enabled
SETDefault !DEVICE12 -ATUN CUCONTrol = Enabled	Remote Site B:
SETDefault !DEVICE14 -ATUN CUADDRess = 14	SETDefault !4 -ATUN PORT OWNer = ATUN
SETDefault !DEVICE14 -ATUN LocalMac = 400000000002	SETDefault !4 -ATUN PATH BAud = 9.6
SETDefault !DEVICE14 -ATUN RemoteMac = 400002002001	SETDefault !4 -PATH DataBits = 8
SETDefault !DEVICE14 -ATUN CUCONTrol = Enabled	SETDefault !4 -PATH PARity = Even
	SETDefault !4 -PATH StopBits = 1
	SETDefault !4 -ATUN PATH CONTrol = Enable
	SETDefault !4 -ATUN FrameSize = 80
	SETDefault !4 -ATUN IdleTimer = 10
	SETDefault !4 -ATUN FrameGap = 10
	SETDefault !4 -ATUN PortCONTrol = (Enabled, RemoteSite, NoAddress, NoForcePoll, NoBCAddr)
	ADD !4 -ATUN PortCU = DEVICE14
	SETDefault !DEVICE14 -ATUN CUADDRess = 14
	SETDefault !DEVICE14 -ATUN LocalMac = 400002002001
	SETDefault !DEVICE14 -ATUN RemoteMac = 400000000002
	SETDefault !DEVICE14 -ATUN CUCONTrol = Enabled

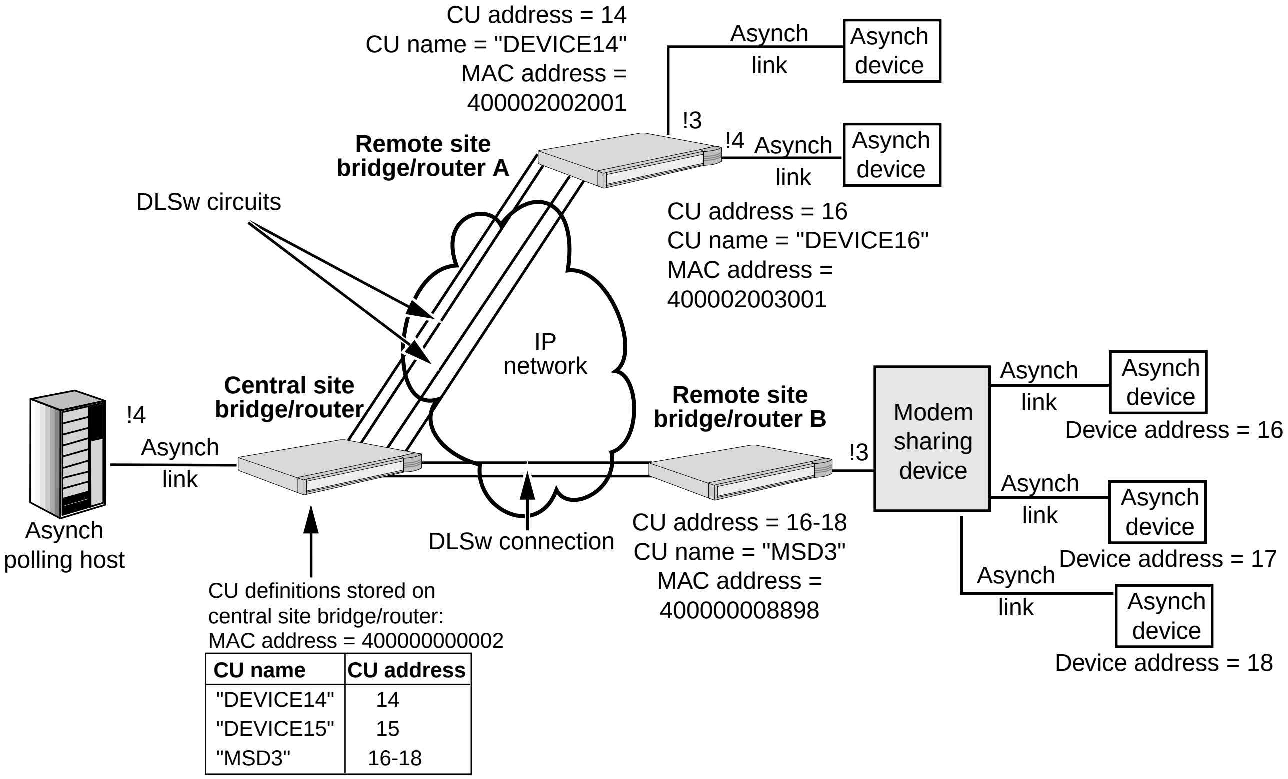
Example 2: Multiple Asynch Devices at Remote Sites

Figure 31-3 shows a configuration with a central site communicating with two remote sites, each with a different configuration. The bridge/router at remote site A has multiple attached asynch devices, each one over a separate port. The bridge/router at remote site B is connected to a modem sharing device, which is connected to multiple asynch devices; Since only one tunnel is allowed to that port (a remote site port only allows a single CU definition), an address range is used in the CU definition at the central site. The CU range allows all three devices to map to a single tunnel.

Table 31-2 lists the commands to configure the asynch tunneling at both the central site and for the three remote sites.

For this example, the asynch host is operating at 9600 baud, using 7 databits. The asynch host is transmitting even parity and receiving odd parity and expecting two stop bits. The asynch devices are similar but transmit odd and receive even parity. The host sends variable length frames, each with an address in the first byte, and the special address 255 is meant to go to all devices (broadcast). The devices send variable-sized frames in response that always terminate with an ASCII Carriage Return (decimal 13).

Both the asynch host and the asynch devices send characters back-to-back within a frame, but delay at least 30 milliseconds between frames. When receiving frames, the asynch host and asynch devices do not depend on the inter-frame gap to recognize a frame.



CU name	CU address
"DEVICE14"	14
"DEVICE15"	15
"MSD3"	16-18

Figure 31-3 Asynch Configuration Example (Multiple CUs at Remote Sites)

Table 31-2 Asynch Configuration Example Commands (Multiple CUs at Remote Sites)

Commands Entered at Central Site:	Commands Entered at Remote Sites:
	Remote Site A:
SETDefault !4 -PORT OWNer = ATUN	SETDefault !3 -PORT OWNer = ATUN
SETDefault !4 -PATH BAud = 9.6	SETDefault !3 -PATH BAud = 9.6
SETDefault !4 -PATH DataBits = 7	SETDefault !3 -PATH DataBits = 7
SETDefault !4 -PATH RxParity = Even	SETDefault !3 -PATH RxParity = Odd
SETDefault !4 -PATH TxParity = Odd	SETDefault !3 -PATH TxParity = Even
SETDefault !4 -PATH StopBits = 2	SETDefault !3 -PATH StopBits = 2
SETDefault !4 -PATH CONTrol = Enable	SETDefault !3 -PATH CONTrol = Enable
SETDefault !4 -ATUN IdleTimer = 20	ADD !3 -ATUN FrameChars 13
SETDefault !4 -ATUN FrameGap = 0	SETDefault !3 -ATUN IdleTimer = 20
SETDefault !4 -ATUN PortCONTrol = (Enabled, CentralSite, Address, BCAddr, NoForcePoll, NoTestEcho)	SETDefault !3 -ATUN FrameGap = 0
SETDefault !4 -ATUN AddrLOCation = 0	SETDefault !3 -ATUN PortCONTrol = (Enabled, RemoteSite, NoAddress, NoBCAddr, NoForcePoll, NoTestEcho)
SETDefault !4 -ATUN BroadCastAddr = 255	SETDefault !4 -PORT OWNer = ATUN
ADD !4 -ATUN PortCU = DEVICE14 DEVICE15 MSD3	SETDefault !4 -PATH BAud = 9.6
SETDefault !DEVICE14 -ATUN CUADDRess = 14	SETDefault !4 -PATH DataBits = 7
SETDefault !DEVICE14 -ATUN LocalMac = 400000000002	SETDefault !4 -PATH RxParity = Odd
SETDefault !DEVICE14 -ATUN RemoteMac = 400002002001	SETDefault !4 -PATH TxParity = Even

(continued)

Table 31-2 Asynch Configuration Example Commands (Multiple CUs at Remote Sites) (continued)

Commands Entered at Central Site:	Commands Entered at Remote Sites:
SETDefault !DEVICE14 -ATUN CUCONTrol = Enabled	SETDefault !4 -PATH StopBits = 2
SETDefault !DEVICE15 -ATUN CUADDRess = 16	SETDefault !4 -PATH CONTrol = Enable
SETDefault !DEVICE15 -ATUN LocalMac = 400000000002	ADD !4 -ATUN FrameChars 13
SETDefault !DEVICE15 -ATUN RemoteMac = 400002003001	SETDefault !4 -ATUN IdleTimer = 20
SETDefault !DEVICE15 -ATUN CUCONTrol = Enabled	SETDefault !4 -ATUN FrameGap = 0
SETDefault !MSD3 -ATUN CUADDRess = 16-18	SETDefault !4 -ATUN PortCONTrol = (Enabled, RemoteSite, NoAddress, NoBCAddr, NoForcePoll, NoTestEcho)
SETDefault !MSD3 -ATUN LocalMac = 400000000002	ADD !3 -ATUN PortCU = DEVICE14
SETDefault !MSD3 -ATUN RemoteMac = 400000008898	SETDefault !DEVICE14 -ATUN LocalMac = 400002002001
SETDefault !MSD3 -ATUN CUCONTrol = Enabled	SETDefault !DEVICE14 -ATUN LocalSap = 04
	SETDefault !DEVICE14 -ATUN RemoteMac = 400000000002
	SETDefault !DEVICE14 -ATUN CUCONTrol = Enabled
	ADD !4 -ATUN PortCU = DEVICE16
	SETDefault !DEVICE15 -ATUN LocalMac = 400002003001
	SETDefault !DEVICE15 -ATUN LocalSap = 04
	SETDefault !DEVICE15 -ATUN RemoteMac = 400000000002
	SETDefault !DEVICE15 -ATUN CUCONTrol = Enabled
	Remote Site B:
	SETDefault !4 -ATUN PORT OWNer = ATUN
	SETDefault !4 -ATUN PATH BAud = 9.6
	SETDefault !4 -ATUN PATH DataBits = 7
	SETDefault !4 -PATH RxParity = Odd
	SETDefault !4 -PATH TxParity = Even
	SETDefault !4 -PATH StopBits = 2
	SETDefault !4 -PATH CONTrol = Enable
	SETDefault !4 -ATUN FrameChars 13
	SETDefault !4 -ATUN IdleTimer = 20
	SETDefault !4 -ATUN FrameGap = 0
	SETDefault !4 -ATUN PortCONTrol = (Enabled, RemoteSite, NoAddress, NoBCAddr, NoForcePoll, NoTestEcho)
	ADD !3 -ATUN PortCU = MSD3
	SETDefault !MSD3 -ATUN LocalMac = 400000008898
	SETDefault !MSD3 -ATUN RemoteMac = 400000000002
	SETDefault !MSD3 -ATUN CUCONTrol = Enabled

32 CONFIGURING BOUNDARY ROUTING SYSTEM ARCHITECTURE

This chapter describes how to implement Boundary Routing system architecture, how Boundary Routing works, and where it can be used.

The information in this chapter applies to Boundary Routing system architecture in both non-IBM and IBM environments except where specifically called out. A *non-IBM environment* is an environment where Systems Network Architecture (SNA) or NetBIOS are not used, for example, an Internet Protocol (IP) or Internetwork Packet Exchange (IPX) environment. An *IBM environment* is an SNA or NetBIOS environment.

For conceptual information, refer to "How Boundary Routing System Architecture Works" on page 32-26. For information on using Integrated Services Digital Network (ISDN) systems in a Boundary Routing topology, refer to Chapter 37.

After you configure Boundary Routing, you can set up auto startup phase 2 on the central node. For information on auto startup, refer to Chapter 33.

Configuring Basic Boundary Routing

This section describes how to configure a wide area port on the central node for Boundary Routing over the Point-to-Point Protocol (PPP), Frame Relay, or X.25. For information on which NETBuilder platforms can be used as a central node, refer to "Where Can Boundary Routing Be Used?" on page 32-26.

Prerequisites

Before performing one of the following procedures, make sure that you have configured the wide area port and path. For information on configuring wide area ports and paths, refer to Chapter 1.

For the procedure for configuring PPP, refer to the next section. For Frame Relay configuration procedures, refer to "Configuring for Frame Relay" on page 32-6. For X.25 configuration procedures, refer to "Configuring for X.25" on page 32-11.

Configuring for PPP

To configure the Boundary Routing port for PPP, follow these steps:

1 Verify that the owner of the wide area port is PPP by entering:

```
SHow -PORT CONFiguration
```

If the port owner is not PPP, reconfigure the owner using:

```
SETDefault !<port> -PORT OWNer = PPP
```

2 Enable Boundary Routing using:

```
SETDefault !<port> -BCN CONTrol = Enabled
```

3 If you are configuring a NETBuilder II bridge/router, verify that the configured and actual media types of the remote LAN network match use:

```
SHow !<port> -BCN RemoteLanType
```

If the configured and actual media types do not match, reconfigure the media type. For example, set the media type to token ring using:

```
SETDefault !<port> -BCN RemoteLanType = TokenRing
```

4 If you are configuring a NETBuilder II bridge/router and the port you are configuring is connected to a token ring peripheral node, set the MAC address format for Address Resolution Protocol (ARP) to noncanonical using:

```
SETDefault !<port> -PORT ProtMacAddrFmt = NonCanonARP
```

5 If you are planning to use the IPX Protocol in your Boundary Routing topology, determine if you want to use smart filtering.

Do not use smart filtering unless you have a stable WAN link.

You can use smart filtering to eliminate NetWare Routing Information Protocol (NRIP) and Service Advertising Protocol (SAP) broadcasts, and protocol island traffic. For conceptual information, refer to “Reduced WAN Usage Costs” on page 32-38.

By default, smart filtering is disabled on all ports. You can enable smart filtering using:

```
SETDefault !<port> -BCN CONTrol = SmartFiltering
```

For smart filtering to operate on a SuperStack II NETBuilder boundary router, you must also run Boundary Routing software version 7.0 or later.

After enabling the port later in this procedure, a short delay occurs before smart filtering begins filtering packets.

If you plan to configure NRIP/SAP updates on the central node to be incremental (-NRIP CONTrol = NoPEriodic and -SAP CONTrol = NoPEriodic), then smart filtering is not needed. You cannot obtain additional bandwidth savings. The use of nonperiodic NRIP and SAP updates assumes that servers on leaf networks can operate in a nonperiodic environment.

6 If you are configuring Boundary Routing in an IBM environment, follow these steps:

a Enable the Boundary Routing of IBM traffic using:

```
SETDefault !<port> -BCN CONTrol = IbmTraffic
```

For information on how Boundary Routing in an IBM environment works, refer to “How Boundary Routing System Architecture Works” on page 32-26.

b By default, the central node is configured to perform Boundary Routing of SNA traffic only. If you want the central node to perform Boundary Routing of NetBIOS traffic also, enter:

```
SETDefault -DLSw CONTrol = EnableNetBios
```

c Enable LLC2 on all LAN ports of the central node using:

```
SETDefault !<port> -LLC2 CONTrol = Enable
```

If your topology includes clients on leaf networks that must exchange data, use this same syntax on the wide area ports of your central node that interface these leaf networks. For conceptual information on this topic, refer to “Peer Data Exchange” on page 32-41.

d Configure the logical link control, type 2 (LLC2) data link interface on all LAN ports of the central node, if necessary.

Refer to Chapter 21 for configuration information. In most cases, the default settings of the -LLC2 parameters are sufficient and minimal (if any) configuration will be necessary.

e Verify that the appropriate type of bridging is enabled on all LAN ports of the central node.

The token ring port must be configured for transparent bridging when configuring a boundary router. This is the default setting, and must not be disabled.

On LAN ports that use Ethernet as the media type, verify that transparent bridging is enabled using:

```
SHow !<port> -BRidge TransparentBRidge
```

If the value of this parameter is not set to TransparentBRidge, use:

```
SETDefault !<port> -BRidge TransparentBRidge = TransparentBRidge
```

On LAN ports that use token ring as the media type, verify that source route bridging is enabled and that a unique ring number has been assigned by entering:

```
SHow -SR CONFiguration
```

If source route bridging has not been enabled, use:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

If a ring number has not been assigned, use:

```
  SETDefault !<port> -SR RingNumber = [<number> (1-4095) |
0x<number>
  (1-FFF)]
```

f If desired, enable the end system source routing of LLC2 packets, and optionally, of IP packets using:

```
SETDefault !<port> -SR RouteDiscovery = (LLC2, [IP])
```

When enabling end system source routing, you must specify the route discovery of LLC2 end system packets so that LLC2 sessions are locally terminated at both central and peripheral nodes. For more information on local termination, refer to “Local Termination” on page 25-34. You can optionally specify the route discovery of IP end system packets so that you can determine the connectivity of network devices using the PING command.

g Reset the default virtual ring number (92) for your tunnel, if desired, using:

```
SETDefault -LLC2 TUNnelVRing = <Number>(1 - 4095)
```

7 Enable the port.

By default, all ports are enabled; however, you must re-enable the Boundary Routing port for the -PORT parameters you configured in earlier steps to take effect. To re-enable a port, use:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

8 Determine if you want to use data compression over the Boundary Routing path.

If you want to use data compression, additional configuration steps are required. For information, refer to Chapter 39. The procedure for configuring data compression on a link connecting a Boundary Routing peripheral node to a central node is exactly the same as on a link connecting two access routers.

9 If you want to administer IP addresses for the peripheral nodes (for Telnet and Simple Network Management Protocol (SNMP) management) from the central node, you must decide whether you want your peripheral nodes to acquire their IP addresses from a Reverse Address Resolution Protocol (RARP) or BOOTP server.

With the central node configured as a RARP server and the peripheral node configured as a RARP client (by default), the peripheral node can obtain its IP address by sending a RARP request to the central node. If you decide to use the RARP server option, follow steps a, b, and c.

By default, the peripheral node is configured as a client to a BOOTP server. The BOOTP server must exist on a network attached to the central node. The BOOTP server can be a 3Com product or a product supplied by another vendor. By configuring UDP Broadcast Helper on the central node, the central node propagates BOOTP requests from the peripheral node to the BOOTP server and obtains the peripheral node's IP address. If you decide to use the BOOTP server option, follow steps d and e.

a If you decide to acquire the peripheral node's IP address from a RARP server, enable the RARP server by entering:

```
SETDefault -ARP RarpCONTrol = RarpServer
```

b To acquire the peripheral node's IP address from a RARP server, assign an IP address to the peripheral node.

Two methods exist to assign an IP address to the peripheral node: port-to-IP address mapping and the IP Address Translation Table.

If you do not know the media access control (MAC) address of the peripheral node, use the port-to-IP address mapping. Configure the -IP RemoteAddress parameter to map an IP address to the virtual port over which the central node receives the RARP request from the peripheral node. For example, if the central node has virtual port V3 enabled for Boundary Routing, you can map an IP address to it by entering:

```
SETDefault !V3 -IP RemoteAddress = 129.213.1.1
```

To assign an IP address through the IP Address Translation Table, you need to obtain the MAC address of the peripheral node, which can be found on a label on the back of the SuperStack II NETBuilder platform.

Use the -IP ADDRess parameter to add the IP address and MAC address of the peripheral node to the IP Address Translation Table. For example, to add the IP address of 129.213.1.1 and the MAC address of %080002A00890 to the peripheral node, enter:

```
ADD -IP ADDRess 129.213.1.1 %080002A00890
```

c To acquire the IP address of the peripheral node from a RARP server, check the setting of the -IP ICMPReply parameter by entering:

```
SHow -IP CONFiguration
```

The -IP ICMPReply parameter should be set to Mask. The peripheral node also requests the subnet mask from the central node using an ICMP Address Mask request. The central node responds with the subnet mask configured for the interface on which the request is received. If you do not set the ICMPReply parameter to Mask, the central node will not send the subnet mask to the peripheral node, causing the IP address to be incomplete.

If the setting of this parameter is not Mask, enter:

```
SETDefault -IP ICMPReply = Mask
```

d If you decide to acquire the IP address of the peripheral node from a BOOTP server, configure User Datagram Protocol (UDP) Broadcast Helper or check that UDP Broadcast Helper is configured.

For information on how to configure UDP Broadcast Helper, refer to Chapter 20. To check UDP Broadcast Helper is configured, enter:

```
SHow -UDPHELP CONFiguration
```

In the display that appears, make sure that the -UDPHELP CONTrol parameter is set to Enable and that either BPSERVER or UDP port number 67 appears on the Active Ports list. If one or both steps of the configuration have not been completed, you must follow the appropriate steps to make sure that UDP Broadcast Helper is completely configured. For instructions on completing each step, refer to “Relaying BOOTP and DHCP Traffic” on page 20-4.

e To acquire the IP address of the peripheral node from a BOOTP server, assign an IP address to the peripheral node.

Two methods exist to assign an IP address to the peripheral node: port-to-IP address mapping and MAC address-to-IP address mapping. The type of BOOTP server you are using will determine which method you can use. If your BOOTP server is a 3Com product, you can use either method. If your BOOTP server is a product supplied by another vendor, you must use the MAC address-to-IP address mapping.

You must complete this step on your BOOTP server. For instructions on assigning an IP address using either of these methods, refer to the documentation that accompanies your BOOTP server.

If you are acquiring the IP address of the peripheral node using a BOOTP server provided by a non-3Com vendor, you must add the MAC address and IP address of the peripheral node to the database of the BOOTP server. Refer to the documentation that accompanies your BOOTP server to determine how to do this.

10 After you configure the wide area port on the central node for the Boundary Routing feature, configure the central node for bridging and/or routing for each protocol used in the Boundary Routing topology.

For more information, refer to the bridging and routing chapters.

11 Determine whether the peripheral nodes require any configuration changes.

The configuration procedure can be performed at the peripheral node through an attached console or remotely through the central node using the Telnet Protocol. For additional information, refer to the documentation that accompanies the peripheral nodes.

To verify the configuration, refer to “Verifying the Configuration” on page 32-15.

You can also configure a back-up link over PPP to provide disaster recovery or bandwidth-on-demand. For conceptual information, refer to “Dial-up Backup Line for Disaster Recovery or Bandwidth-on-Demand” on page 32-44. If you need to provide a redundant link or route for mission-critical applications, refer to “Configuring Network Resiliency” on page 32-23.

After you configure the Boundary Routing software, you can set up auto startup phase 2 on the central node. For information on auto startup, refer to Chapter 33.

Configuring for Frame Relay

Before beginning the following procedure, make sure that you have completed the steps in “Prerequisites” on page 32-1.

To configure the wide area and virtual ports on the central node for Boundary Routing over Frame Relay, follow these steps. For information on virtual ports, refer to Chapter 1.

1 Verify that the owner of the wide area port is Frame Relay by entering:

```
SHow -PORT CONFiguration
```

If the port owner is not Frame Relay, reconfigure the owner using:

```
SETDefault !<port> -PORT OWNer = FrameRelay
```

2 If your Frame Relay switch supports a Local Management Interface (LMI) Protocol, verify that LMI is enabled on the wide area port using:

```
SHow [!<port>] -FR CONFiguration
```

The software supports multiple types of LMI. Refer to the description of the -FR CONTrol parameter in Chapter 25 in *Reference for NETBuilder Family Software* for information on the types of LMI supported.

Determine if the type of LMI specified for the -FR CONTrol parameter and the type supported by your switch match. If they do not match, you must reconfigure the LMI type using the -FR CONTrol parameter. For more information on this parameter, refer to Chapter 25 in *Reference for NETBuilder Family Software*.

If the switch does not support any LMI Protocol, configure the -FR CONTrol parameter using:

```
SETDefault !<port> -FR CONTrol = NoLmi
```

Specification ANSI T1.617 describes the LMI Protocol. An appendix in this specification includes Annex-D, which relates to the construction of LMI packets. NTT-LMI is the LMI Protocol supported by NTT Frame Relay switches.

3 Create a virtual port for each remote network that is attached to the Frame Relay cloud using:

```
ADD !<port> -PORT VirtualPort {<path> {<FR_DLCI>}}
```

For example, if you have a remote network on path 4 that uses Frame Relay DLCI 35, add virtual port V1 by entering:

```
ADD !V1 -PORT VirtualPort 4@35
```

4 Enable the Boundary Routing software for each virtual port associated with the path using:

```
SETDefault !<port> -BCN CONTrol = Enabled
```

For example, to enable Boundary Routing on virtual port V1, enter:

```
SETDefault !V1 -BCN CONTrol = Enabled
```

Make sure you enable the -BCN CONTrol parameter on a virtual port, not on a parent port. For information on parent ports, refer to Chapter 1.

5 If you are configuring a NETBuilder II bridge/router, verify that the configured and actual media types of the remote LAN network match using:

```
SHow !<port> -BCN RemoteLanType
```

If the configured and actual media types do not match, reconfigure the media type. For example, set the media type to token ring using:

```
SETDefault !<port> -BCN RemoteLanType = TokenRing
```

6 If you are configuring a NETBuilder II bridge/router and the port you are configuring is connected to a token ring peripheral node, set the MAC address format for ARP to noncanonical using:

```
SETDefault !<port> -PORT ProtMacAddrFmt = NonCanonARP
```

7 If you are planning to use the IPX Protocol in your Boundary Routing topology, determine if you want to use smart filtering.

Do not use smart filtering unless you have a stable WAN link that uses the LMI Protocol.

You can use smart filtering to eliminate NRIP and SAP rebroadcasts, and protocol island traffic. For conceptual information, refer to “Reduced WAN Usage Costs” on page 32-38.

By default, smart filtering is disabled on all ports. You can enable smart filtering by using:

```
SETDefault !<port> -BCN CONTrol = SmartFiltering
```

Make sure you enable or disable smart filtering on the appropriate virtual ports to suit your needs.

After you enable each virtual port later in this procedure, a short delay occurs before smart filtering begins filtering packets.

If you plan to configure NRIP/SAP updates on the central node to be incremental (-NRIP CONTrol = NoPEriodic and -SAP CONTrol = NoPEriodic), then smart filtering is not needed. You cannot obtain additional bandwidth savings. The use of nonperiodic NRIP and SAP updates assumes that servers on leaf networks operate in a nonperiodic environment.

For smart filtering to operate on a SuperStack II NETBuilder boundary router, you must use Boundary Routing software version 7.0 or later.

8 If you are configuring Boundary Routing in an IBM environment, follow these steps:

a Enable the Boundary Routing of IBM traffic using:

```
SETDefault !<port> -BCN CONTrol = IbmTraffic
```

For information on how Boundary Routing in an IBM environment works, refer to "How Boundary Routing System Architecture Works" on page 32-26.

b By default, the central node is configured to perform Boundary Routing of SNA traffic only. If you also want the central node to perform Boundary Routing of NetBIOS traffic, enter:

```
SETDefault -DLSw CONTrol = EnableNetBios
```

c Enable LLC2 on all LAN ports of the central node using:

```
SETDefault !<port> -LLC2 CONTrol = Enable
```

If your topology includes clients on leaf networks that must exchange data, use the same syntax specified above on the virtual ports that interface these leaf networks. For conceptual information on this topic, refer to "Peer Data Exchange" on page 32-41.

d Configure the LLC2 data link interface on all central node LAN ports, if necessary.

Refer to Chapter 21 for configuration information. In most cases, the default settings of the -LLC2 parameters are sufficient and minimal (if any) configuration will be necessary.

e Verify that the appropriate type of bridging is enabled on all LAN ports of the central node.

On LAN ports that use Ethernet as the media type, verify that transparent bridging is enabled using:

```
SHow !<port> -BRidge TransparentBRidge
```

If the value of this parameter is not set to TransparentBRidge, use:

```
SETDefault !<port> -BRidge TransparentBRidge = TransparentBRidge
```

On LAN ports that use token ring as the media type, verify that source route bridging is enabled and that a unique ring number has been assigned by entering:

```
SHow -SR CONFiguration
```

If source route bridging has not been enabled, use:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

If a ring number has not been assigned, use:

```
SETDefault !<port> -SR RingNumber = [<number>(1-4095) |
  0x<number>(1-FFF)]
```

f If desired, enable the end system source routing of LLC2 packets, and optionally, of IP packets using:

```
SETDefault !<port> -SR RouteDiscovery = (LLC2, [IP])
```

When enabling end system source routing, you must specify the route discovery of LLC2 end system packets so that LLC2 sessions are locally terminated at both central and peripheral nodes. For more information on local termination, refer to "Local Termination" on page 25-34. You can optionally specify the route discovery of IP end system packets so that you can determine the connectivity of network devices using the PING command.

g Reset the default virtual ring number (92) for your tunnel, if desired, using:

```
SETDefault -LLC2 TUNnelVRing = <Number>(1 - 4095)
```

9 Enable each virtual port.

By default, all virtual ports are enabled; however, you must re-enable each virtual port for the -PORT parameters you configured in earlier steps to take effect. To re-enable a virtual port, use:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

10 Determine if you want to use data compression over the Boundary Routing path.

If you want to use data compression, additional configuration steps are required. For information, refer to Chapter 39. The procedure for configuring data compression on a link connecting a Boundary Routing peripheral node to a central node is exactly the same as on a link connecting two access routers.

11 If you want to administer IP addresses for the peripheral nodes (for Telnet and SNMP management) from the central node, you must decide whether you want your peripheral nodes to acquire their IP addresses from a RARP or BOOTP server.

With the central node configured as a RARP server and the peripheral node configured as a RARP client (by default), the peripheral node can obtain its IP address by sending a RARP request to the central node. If you decide to use the RARP server option, follow steps a, b, and c.

By default, the peripheral node is configured as a client to a BOOTP server. The BOOTP server must exist on a network attached to the central node. The BOOTP server can be a 3Com product or a product supplied by another vendor. By configuring UDP Broadcast Helper on the central node, the central node propagates BOOTP requests from the peripheral node to the BOOTP server and obtains the IP address of the peripheral node. If you decide to use the BOOTP server option, follow steps d and e.

a If you decide to acquire the IP address of the peripheral node from a RARP server, enable the RARP server by entering:

```
SETDefault -ARP RarpCONTrol = RarpServer
```

b To acquire the peripheral node’s IP address from a RARP server, assign an IP address to the peripheral node.

Two methods exist to assign an IP address to the peripheral node: port-to-IP address mapping and the IP Address Translation Table.

If you do not know the MAC address of the peripheral node, use the port-to-IP address mapping. Configure the -IP RemoteAddress parameter to map an IP address to the virtual port over which the central node receives the RARP request from the peripheral node. For example, if the central node has virtual port V3 enabled for Boundary Routing, you can map an IP address to it by entering:

```
SETDefault !V3 -IP RemoteAddress = 129.213.1.1
```

To assign an IP address through the IP Address Translation Table, you need to obtain the MAC address of the peripheral node, which can be found on a label on the back of the SuperStack II bridge/router.

Use the -IP ADDRess parameter to add the IP address and MAC address of the peripheral node to the IP Address Translation Table. For example, to add the IP address of 129.213.1.1 and the MAC address of %080002A00890 to the peripheral node, enter:

```
ADD -IP ADDRess 129.213.1.1 %080002A00890
```

c To acquire the IP address of the peripheral node from a RARP server, check the setting of the -IP ICMPReply parameter by entering:

```
SHow -IP CONFiguration
```

The -IP ICMPReply parameter should be set to Mask. The peripheral node also requests the subnet mask from the central node using an ICMP Address Mask request. The central node responds with the subnet mask configured for the interface on which the request is received. If you do not set the ICMPReply parameter to Mask, the central node will not send the subnet mask to the peripheral node, causing the IP address to be incomplete.

If the setting of this parameter is not Mask, enter:

```
SETDefault -IP ICMPReply = Mask
```

d If you decide to acquire the IP address of the peripheral node from a BOOTP server, configure UDP Broadcast Helper or make certain that UDP Broadcast Helper is configured.

For information on how to configure UDP Broadcast Helper, refer to Chapter 20. To make certain UDP Broadcast Helper is configured, enter:

```
SHow -UDPHELP CONFiguration
```

In the display that appears, make sure that the -UDPHELP CONTrol parameter is set to Enable and that either BPSERVER or UDP port number 67 appears on the Active Ports list. If one or both steps of the configuration have not been completed, you must follow the appropriate steps to make sure that UDP Broadcast Helper is completely configured. For instructions on completing each step, refer to “Relaying BOOTP and DHCP Traffic” on page 20-4.

e To acquire the IP address of the peripheral node from a BOOTP server, assign an IP address to the peripheral node.

Two methods exist to assign an IP address to the peripheral node: port-to-IP address mapping and MAC address-to-IP address mapping. The type of BOOTP server you are using will determine which method you can use. If your BOOTP server is a 3Com product, you can use either method. If your BOOTP server is a product supplied by another vendor, you must use the MAC address-to-IP address mapping.

You must complete this step on your BOOTP server. For instructions on assigning an IP address using either of these methods, refer to the documentation that accompanies your BOOTP server.

If you are acquiring the IP address of the peripheral node using a BOOTP server provided by a non-3Com vendor, you must add the MAC address and IP address of the peripheral node to the database of the BOOTP server. Refer to the documentation that accompanies your BOOTP server to determine how to do this.

12 After configuring the wide area and virtual ports on the central node for the Boundary Routing feature, configure the virtual ports of the central node for bridging and/or routing for each protocol used in the Boundary Routing topology.

For more information on bridging and routing over Frame Relay, refer to Chapter 42.

13 Determine whether the peripheral nodes require any configuration changes.

The configuration procedure can be performed at the peripheral node through an attached console or remotely through the central node using the Telnet Protocol. For additional information, refer to the documentation that accompanies the peripheral nodes.

To verify the configuration, refer to "Verifying the Configuration" on page 32-15.

If you need to provide a redundant link or route for mission-critical applications, refer to "Configuring Network Resiliency" on page 32-23.

After you configure Boundary Routing, you can also set up auto startup phase 2 on the central node. For information on auto startup, refer to Chapter 33.

Configuring for X.25

Before beginning the following steps, make sure that you have completed the steps in "Prerequisites" on page 32-1.

If you are configuring or are already performing Boundary Routing over X.25 and you re-enable the X.25 virtual port on your central node by entering SETDefault !Vn -PORT CONTrol = Enabled, you must also re-enable the X.25 path of the peripheral node. To re-enable the path, enter:

```
SETDefault !Vn -PATH CONTrol = Enabled
```

If you do not enter this command, the X.25 path of the peripheral node will remain up, but the peripheral node will not know that the X.25 virtual port of the central node has gone down. The peripheral node will continue to transmit packets to the central node, but the central node will not respond.

To configure wide area and virtual ports on the central node for Boundary Routing over X.25, follow these steps. For information on virtual ports, refer to Chapter 1.

1 Set the owner of the wide area port to X.25 using:

```
SETDefault !<port> -PORT OWNer = X25
```

2 Configure each wide area port for communication with an X.25 PDN by assigning a DTE address using:

```
SETDefault !<port> -X25 X25Address = <0–99999999999999>(1–15
  digits)
```

For example, to assign a DTE address of 31102859060 to port 3, enter:

```
SETDefault !3 -X25 X25Address = 31102859060
```

3 Create a virtual port for each remote network that is attached to the X.25 cloud using:

```
ADD !<port> -PORT VirtualPort {<path> {<X.25 DTE>}}
```

For example, if you have a remote network on path 4 that uses X.25 DTE 31107551234, add virtual port V1 by entering:

```
ADD !V1 -PORT VirtualPort 4#31107551234
```

4 Enable the Boundary Routing feature on each virtual port associated with the path using:

```
SETDefault !<port> -BCN CONTrol = Enabled
```

For example, to enable Boundary Routing on virtual port V1, enter:

```
SETDefault !V1 -BCN CONTrol = Enabled
```

Make sure you enable the -BCN CONTrol parameter on a virtual port, not on a parent port. For information on parent ports, refer to Chapter 1.

5 If you are configuring a NETBuilder II bridge/router, verify that the configured and actual media types of the remote LAN network match using:

```
SHow !<port> -BCN RemoteLanType
```

If the configured and actual media types do not match, reconfigure the media type. For example, set the media type to token ring using:

```
SETDefault !<port> -BCN RemoteLanType = TokenRing
```

6 If you are configuring a NETBuilder II bridge/router and the port you are configuring is connected to a token ring peripheral node, set the MAC address format for ARP to noncanonical using:

```
SETDefault !<port> -PORT ProtMacAddrFmt = NonCanonARP
```

7 If you are planning to use the IPX Protocol in your Boundary Routing topology, determine if you want to use smart filtering.

You can use smart filtering to eliminate IPX NRIP and SAP rebroadcasts, and protocol island traffic. For conceptual information, refer to “Reduced WAN Usage Costs” on page 32-38.

By default, smart filtering is disabled on all ports. You can enable smart filtering using:

```
SETDefault !<port> -BCN CONTrol = SmartFiltering
```

Make sure you enable or disable smart filtering on the appropriate virtual ports to suit your needs.

After you enable each virtual port later in this procedure, a short delay occurs before smart filtering begins filtering packets.

If you plan to configure NRIP/SAP updates on the central node to be incremental (-NRIP CONTrol = NoPEriodic and -SAP CONTrol = NoPEriodic), then smart filtering is not needed. You can obtain no further bandwidth savings. The use of nonperiodic NRIP and SAP updates assumes that servers on leaf networks can operate in a nonperiodic environment.

For smart filtering to operate on a SuperStack II NETBuilder boundary router, you must use Boundary Routing software version 7.0 or later.

8 If you are configuring the Boundary Routing feature in an IBM environment, follow these steps:

a Enable the Boundary Routing of IBM traffic using:

```
SETDefault !<port> -BCN CONTrol = IbmTraffic
```

For information on how Boundary Routing in an IBM environment works, refer to "How Boundary Routing System Architecture Works" on page 32-26.

b By default, the central node is configured to perform Boundary Routing of SNA traffic only. If you also want the central node to perform Boundary Routing of NetBIOS traffic, enter:

```
SETDefault -DLSw CONTrol = EnableNetBios
```

c Enable LLC2 on all LAN ports of the central node using:

```
SETDefault !<port> -LLC2 CONTrol = Enable
```

If your topology includes clients on leaf networks that must exchange data, use the same syntax specified above on the virtual ports that interface these leaf networks. For conceptual information on this topic, refer to "Peer Data Exchange" on page 32-41.

d Configure the LLC2 data link interface on all central node LAN ports, if necessary.

Refer to Chapter 21 for configuration information. In most cases, the default settings of the -LLC2 parameters should be sufficient and little, if any, configuration should be necessary.

e Verify that the appropriate type of bridging is enabled on all LAN ports of the central node.

On LAN ports that use Ethernet as the media type, verify that transparent bridging is enabled using:

```
SHow !<port> -BRidge TransparentBRidge
```

If the value of this parameter is not set to TransparentBRidge, use:

```
SETDefault !<port> -BRidge TransparentBRidge = TransparentBRidge
```

On LAN ports that use token ring as the media type, verify that source route bridging is enabled and that a unique ring number has been assigned by entering:

```
SHow -SR CONFiguration
```

If source route bridging has not been enabled, use:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

If a ring number has not been assigned, use:

```
SETDefault !<port> -SR RingNumber = [<number>(1-4095)|
 0x<number>(1-FFF)]
```

f If desired, enable the end system source routing of LLC2 packets, and optionally, of IP packets using:

```
SETDefault !<port> -SR RouteDiscovery = (LLC2, [IP])
```

When enabling end system source routing, you must specify the route discovery of LLC2 end system packets so that LLC2 sessions are locally terminated at both central and peripheral nodes. For more information on local termination, refer to "Local Termination" on page 25-34. You can optionally specify the route discovery of IP end system packets so that you can determine the connectivity of network devices using the PING command.

g Reset the default virtual ring number (92) for your tunnel, if desired, using:

```
SETDefault -LLC2 TUNnelVRing = <Number>(1 - 4095)
```

9 Verify that the protocol identifier to be included in an outgoing X.25 call request is set appropriately using:

```
SHow !<port> -BCN X25ProtID
```

If the setting is inappropriate, specify a new protocol identifier using:

```
SETDefault !<port> -BCN X25ProtID = <protocol id> (octet)
```

The valid range includes 1 through 0xFF.

10 If you want to assign a higher priority to boundary-routed packets than to other types of traffic, prioritize traffic on the Boundary Routing port.

To assign a priority to boundary-routed packets, configure X.25 user profiles using the -PROFile X25ProfileType parameter. Refer to "ProfileType" on page 41-6 and to "X.25 Profiles Configuration Examples" on page 45-6 for more information.

The default values of the X.25 parameters adhere to the default values of the X.25 standard. However, depending on the requirements of the X.25 switch your central node is connected to, it may be necessary to adjust values of parameters such as X25PacketSiZe, X25ThruputClass, and X25WindowSiZe. For more information on the X25ProfileType parameter and information on adjusting X.25 parameters to suit your installation, refer to Chapter 45 in *Reference for NETBuilder Family Software*.

11 Enable each virtual port.

By default, all virtual ports are enabled; however, you must re-enable each virtual port for the -PORT parameters you configured in earlier steps to take effect. To re-enable a virtual port, use:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

12 Determine if you want to use data compression over the Boundary Routing path.

If you want to use data compression, additional configuration steps are required. For information, refer to Chapter 39. The procedure for configuring data compression on a link connecting a Boundary Routing peripheral node to a central node is exactly the same as on a link connecting two access routers.

13 If you want to administer IP addresses for the peripheral nodes (for Telnet and SNMP management) from the central node, use port-to-IP address mapping or the RARP IP Address Translation Table.

a Enable the RARP server so that the central node can respond to RARP queries from the peripheral node by entering:

```
SETDefault -ARP RarpCONTrol = RarpServer
```

With the central node configured as the RARP server and the peripheral node configured as the RARP client (by default), the peripheral node can obtain its IP address by sending a RARP request to the central node.

b Assign an IP address to the peripheral node.

If you do not know the MAC address of the peripheral node, you can use the port-to-IP address mapping. Configure the -IP RemoteAddress parameter to map an IP address to the port over which the central node receives the RARP request from the peripheral node. For example, if the virtual port V3 is enabled for the Boundary Routing feature, enter:

```
SETDefault !V3 -IP RemoteAddress = 129.213.1.1
```

To assign an IP address through the RARP IP Address Translation Table, you need the MAC address of the peripheral node. The MAC address of the peripheral node can be found on a label on the back of the SuperStack II bridge/router.

Use the -IP ADDRess parameter to add the IP address of the peripheral node and MAC address to the IP Address Translation Table. For example, to add the peripheral node IP address of 129.213.1.1 and the MAC address of %080002A00890, enter:

```
ADD -IP ADDRess 129.213.1.1 %080002A00890
```

c Check the setting of the ICMPReply parameter by entering:

```
SHow -IP CONFiguration
```

The -IP ICMPReply parameter should be set to Mask. The peripheral node also requests the subnet mask from the central node using an ICMP Address Mask request. The central node responds with the subnet mask configured for the interface on which the request is received. If you do not set the ICMPReply parameter to Mask, the central node will not send the subnet mask to the peripheral node, causing the IP address to be incomplete.

If the setting of this parameter is not Mask, enter:

```
SETDefault -IP ICMPReply = Mask
```

14 After you configure the wide area and virtual ports of the central node for the Boundary Routing feature, configure the virtual ports central node for bridging and/or routing for each protocol used in the Boundary Routing topology.

For more information on bridging and routing over X.25, refer to Chapter 45.

15 Determine whether the peripheral nodes require any configuration changes.

The configuration procedure can be performed at the peripheral node through an attached console or remotely through the central node using the Telnet Protocol. For additional information, refer to the document that accompanies the peripheral nodes.

To verify the configuration, refer to "Verifying the Configuration" next.

If you need to provide a redundant route for mission-critical applications, refer to "Configuring Network Resiliency" on page 32-23.

Verifying the Configuration

To verify the initial configuration of your Boundary Routing ports or troubleshoot problems related to Boundary Routing over PPP, Frame Relay, or X.25, follow these steps:

1 Check the state of the ports by entering:

```
SHow -PORT CONFiguration
```

In the Current Port Parameters display, verify the following items:

- Under the Owner column, the owner of the wide area port is set correctly.
- Under the Ctrl column, the port or virtual port is enabled.
- Under the State column, the state is Up.

2 If the port state is not Up, check the state of the paths by entering:

```
SHow -PATH CONFiguration
```

In the Current Path Parameters display, verify the following items:

- Under the Ctrl column, the wide area path is enabled.
- Under the State column, the state is Up.
- Under the Conn column, the connector type is appropriately set.
- Under the Clock column, the clock source is correct.

3 Verify that Boundary Routing is enabled on each port you configured for Boundary Routing by entering:

```
SHow -BCN CONTrol
```

4 If you have configured the Boundary Routing of IBM traffic, determine the status of the Boundary Routing port using:

```
SHow !<port> -BCN IbmStatus
```

In the IBM Status display, verify the following:

- Under the Port State column, the state is UP.
- Under the Status column, the status is ACTIVE.
- Under the State column, the state is RUNNING.

The status will be INACTIVE and the state will be STARTING for a short time while the port is activating.

The status will be INACTIVE and the state will be DISABLED if Boundary Routing has been improperly configured on the port. Refer to "Configuring Basic Boundary Routing" on page 32-1 to determine which step was improperly completed and redo it.

The state will be INACTIVE and the state will be REMOTE - UNKNOWN if the peripheral node is running a version of software that is incompatible with the software running on the central node or a problem exists with the WAN. Check the version of software that is running on the peripheral node. To determine the version of software that should be running, refer to the release notes. If software incompatibility is not the problem, check the cabling of the peripheral node and, if necessary, go on to the following step to further check the WAN.

5 If you have configured the Boundary Routing feature over PPP, verify the PPP configuration and status by entering:

```
SHow -PPP STATUS
```

The Link Control Protocol (LCP) state and the Network Control Protocol (NCP) state display. In the LCP and NCP State display, verify the following items:

- In the PPP Link Control Protocol Status section of the display, verify under the LCP column that each path configured for Boundary Routing is in the OPEN state. The OPEN state indicates that both ends of the serial line connection are up and ready to bridge or route.
- In the PPP Network Control Protocol Status section of the display, verify in the BRIDGE column that the wide area port configured for Boundary Routing is in the OPEN state. The Network Control Protocol Status for all protocols other than bridging should be in the DISABLED state.

6 If you are operating the Boundary Routing feature over Frame Relay, verify the data link connection identifier (DLCI) status for all active Frame Relay ports by entering:

```
SHow -FR DLciStat
```

Verify that the status of the link is active. If a DLCI is not in the list, the corresponding virtual port is down.

7 If you are operating the Boundary Routing feature over X.25, verify the status of the virtual circuits by entering:

```
SHow -X25 STATUS
```

Verify that the state of the virtual circuits is up and running. Also verify that the DTE addresses and the protocols running on the virtual circuits are as you configured them.

8 If you are operating the Boundary Routing feature over an Synchronous Data Link Control (SDLC) line, verify the status of the central unit (CU) by entering:

```
SHow -SDLC CUStatus
```

Troubleshooting the Configuration

If you are unable to make connections to the leaf network after configuring the central node, perform the following troubleshooting procedure. If your configuration continues to operate improperly, contact your network supplier or 3Com for assistance.

To troubleshoot the Boundary Routing configuration, follow these steps:

1 Check that all cables on the central site network and the leaf network are properly connected.

For installation instructions, refer to the installation guides that shipped with the central and peripheral nodes.

2 Verify that the software configuration of the central node is correct.

For details, refer to "Verifying the Configuration" on page 32-15.

3 If you are managing the peripheral node from an SNMP agent, verify that the central node is configured to respond to incoming SNMP requests by entering:

```
SHow -SNMP CONTrol
```

By default, the value of this parameter is set to Manage. If the value of this parameter is not Manage, enter:

```
SETDefault -SNMP CONTrol = Manage
```

4 In TCP/IP environments, make sure that you have correctly configured the default gateway.

Because the central node is the bridge/router in a Boundary Routing environment, use its MAC address (instead of the MAC address of the peripheral node) as the default gateway address when configuring clients on the leaf network that need access to hosts on the central site network.

5 If you see console messages that indicate smart filtering operations have stopped on a port, you can obtain information about the cause of the failure using:

```
SHow !<port> -PORT DIAGnostics
```

To troubleshoot the smart filtering configuration, follow these steps:

a Verify that the link is up and stable.

If the link is prone to dropping packets, smart filtering operations will cease.

b Verify that the Boundary Routing feature is configured correctly and operating.

For port, virtual port, and path configuration steps, refer to Chapter 1. For Boundary Routing port configuration steps, refer to "Configuring Basic Boundary Routing" on page 32-1.

c Make sure the peripheral node supports smart filtering.

If your peripheral node is running pre-7.0 Boundary Routing software, you must upgrade to software version 7.0 or later.

d Make sure the size and configuration of your network is suitable for Boundary Routing and smart filtering operations.

For "out of memory" errors, you should try to decrease memory consumption. For example, you can use IPX policies to limit the view of the network. Optionally, you can increase memory or upgrade your peripheral node.

e Restart smart filtering operations using:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

6 If you are using the Boundary Routing feature in an IBM environment, verify that smart polling and data link switching (DLSw), if applicable, are functioning by following these steps:

a Check the status of smart polling by entering:

```
SHow -SYS STATistics -LLC2
```

Verify that smart polling is functioning by comparing the number of RR frames received and transmitted by Boundary Routing and LAN ports. The number of RR frames received and transmitted by the Boundary Routing port should be substantially less than those on the LAN port.

Determine if Test and Xid frames are being received and transmitted on the correct ports.

For information on the LLC2 statistics display, refer to Appendix H.

b Check the status of DLSw by entering:

```
SHow -DLSw Display
```

A display appears only if circuits are active or if an attempt to make a connection is being made.

c Check the status of SDLC CUs by entering:

```
SHow -SDLC CUStatus
```

7 If you have configured the central node to perform the Boundary Routing operation of NetBIOS traffic and it does not appear to be performing this function, re-enable each port or virtual port on the central node using:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

Customizing Boundary Routing

This section describes procedures you can use to customize your Boundary Routing configuration.

Configuring Dial-Related Enhancements

You can configure the following dial-related enhancements:

- Disaster recovery and bandwidth-on-demand in a PPP environment
- Disaster recovery in a Frame Relay environment
- Dial-on-demand in a PPP environment.

Beginning with software version 9.1, the bridge/router began using the concept of *bandwidth management*, a process that applies static bandwidth, dynamic bandwidth, or a combination of these to provide the ISDN and serial ports using PPP with the bandwidth they need to meet current requirements. Unlike versions of software previous to 9.1, bandwidth management does not view links as primary or secondary lines. It instead dynamically allocates or de-allocates unrestricted, available resources as needed to manage link traffic.

For information on configuring modems, refer to the *WAN Cabling and Connectivity Guide*. You can find this guide on the 3Com World Wide Web site by entering:

http://www.3com.com/

For conceptual information on configuring disaster recovery over Frame Relay, refer to Chapter 42.

Configuring Dual PVCs in a Boundary Routing Environment

This section describes how to configure a secondary permanent virtual circuit (PVC) dedicated to IBM traffic over a Frame Relay link in a Boundary Routing environment. The information in this section applies only to platforms that support the configuration of virtual ports. IBM traffic refers to both IBM System Network Architecture (SNA) and NetBIOS frames.

Dual PVCs are used in environments where IBM traffic is running with non-IBM traffic at a leaf node, and the IBM traffic is forwarded to a central site using Boundary Routing.

To implement dual PVCs, you configure two PVCs over a single Frame Relay Boundary Routing physical port (one PVC is dedicated to IBM traffic and the other PVC is dedicated to non-IBM traffic). Both PVCs are sent to a common bridge/router.

Configuration on the leaf nodes is completed by enabling the ports using the SETDefault -PORT CONTrol = Enable command after all other parameters have been set. In this particular configuration, enabling the ports triggers the transfer of the configuration information to the leaf nodes.

All configuration is performed on the central node.

For conceptual information on how dual PVCs work, refer to "Dual PVCs for IBM Traffic" on page 32-46.

Configuring Dual PVCs on the Central Node

The default condition for Frame Relay PVCs is that a single PVC is used to transmit all traffic types. To configure a separate PVC that will transmit only IBM traffic, you configure a virtual port *pair* on the Boundary Routing Frame Relay physical port on the central node. One of these virtual ports will then be configured to transmit non-IBM traffic. Doing this also indicates the virtual port to which IBM traffic should be redirected.

Figure 32-1 shows dual PVCs configured from a central node to two leaf nodes in an Ethernet environment. Figure 32-2 shows dual PVCs configured from a central node to two leaf nodes in a Token Ring environment.

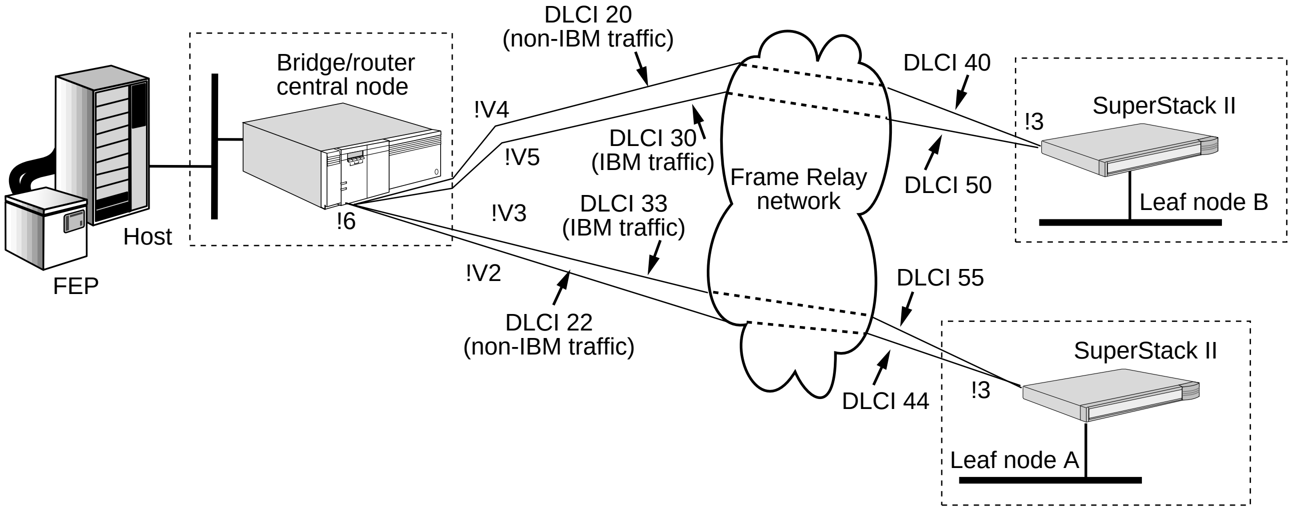


Figure 32-1 Dual PVCs for IBM Traffic in a Boundary Routing Ethernet Environment

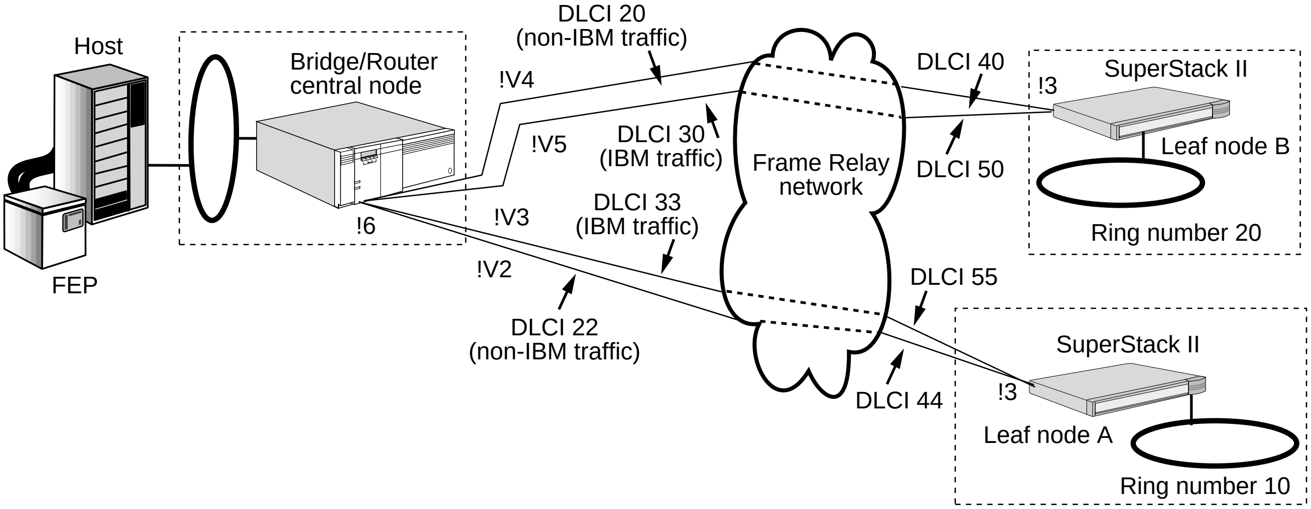


Figure 32-2 Dual PVCs for IBM Traffic in a Token Ring Environment

Prerequisites. Before beginning these procedures, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your wide area interfaces according to procedures in Chapter 1.
- Acquire services from a Frame Relay service provider according to the *WAN Cabling and Connectivity Guide*. You can find this guide on the 3Com World Wide Web site by entering:

  `http://www.3com.com/`

- Make sure ports that will be used for *non-IBM traffic* have the default DLCI value of 0. To check these values, use:

  ```
  SHow !<Vport> -BCN LclNonIbmDlci
  ```

- Make sure virtual ports that will be used for *IBM traffic* have the -SR SrcRouBridge parameter configured to the default value SrcRouBridge.

Refer to *Reference for NETBuilder Family Software* for more information about this parameter and the default value of the DLCIs.

Basic Configuration for Both Ethernet and Token Ring. To perform the basic configuration of dual PVCs on a central node, follow these steps:

1. Using Figure 32-1 or Figure 32-2 as an example, configure four virtual ports on Frame Relay physical port 6 by entering:

   ```
   ADD !V2 -PORT VirtualPort 6@22
   ADD !V3 -PORT VirtualPort 6@33
   ADD !V4 -PORT VirtualPort 6@20
   ADD !V5 -PORT VirtualPort 6@30
   ```

   These commands configure two virtual ports to leaf node A (DLCI 22 and 33) and two to leaf node B (DLCI 20 and 30). The leaf nodes learn the DLCIs from the Frame Relay switch. Virtual ports 2 and 4 are for the PVCs carrying non-IBM traffic and virtual ports 3 and 5 are for the PVCs carrying IBM traffic.

2. Enable Boundary Routing of IBM traffic on virtual ports 3 and 5 by entering:

   ```
   SETDefault !V3 -BCN CONTrol=IbmTraffic
   SETDefault !V5 -BCN CONTrol=IbmTraffic
   ```

3. Define the PVC pairs by entering:

   ```
   SETDefault !V3 -BCN LclNonIbmDlci=22
   SETDefault !V5 -BCN LclNonIbmDlci=20
   ```

   In Figure 32-1 and Figure 32-2, DLCIs 30 and 33 are the default IBM ports.The previous commands indicated to the central node that virtual ports 2 and 3 are one PVC pair, and virtual ports 4 and 5 are another PVC pair. They also instructed the central node to redirect IBM traffic to virtual ports 3 and 5, and to use virtual ports 2 (DLCI 22) and 4 (DLCI 20) for non-IBM traffic to the leaf nodes.

4. Instruct leaf node A to use DLCI 44 and leaf node B to use DLCI 40 for non-IBM traffic by entering:

   ```
   SETDefault !V3 -BCN RemNonIbmDlci=44
   SETDefault !V5 -BCN RemNonIbmDlci=40
   ```

5 Choose one of the following steps:

a If you are configuring dual PVCs in an Ethernet environment, proceed to “Enabling the Ports and Sending Leaf Nodes the Configuration Information” on page 32-23.

b If you are configuring dual PVCs in a Token Ring environment, continue with the procedure in the next section.

Additional Configuration Required for Token Ring Environments. The following procedure enables source route bridging, defines IBM MAC addresses in non-canonical format, and configures Token Ring numbers on the virtual port.

The Token Ring port must be configured for transparent bridging when configuring a boundary router. This is the default setting, and must not be disabled.

Make sure you have completed steps 1 through 4 in “Basic Configuration for Both Ethernet and Token Ring” then continue with the following steps:

1 To configure Token Ring source route bridging on the ports illustrated in Figure 32-2, enter:

```
SETDefault !V2 -BCN RemoteLanType = TokenRing
SETDefault !V3 -BCN RemoteLanType = TokenRing
SETDefault !V4 -BCN RemoteLanType = TokenRing
SETDefault !V5 -BCN RemoteLanType = TokenRing
```

2 To specify IBM MAC addresses in non-canonical format, enter:

```
SETDefault !V2 -PORT PortMacAddrFmt = NonCanARP
SETDefault !V3 -PORT PortMacAddrFmt = NonCanARP
SETDefault !V4 -PORT PortMacAddrFmt = NonCanARP
SETDefault !V5 -PORT PortMacAddrFmt = NonCanARP
```

3 You must also configure the source route ring number and route discovery information on the IBM virtual ports. Assuming that the ring numbers for leaf node A and leaf node B are 10 and 20 respectively, enter:

```
SETDefault !V2 -SR RingNumber = None
SETDefault !V3 -SR RingNumber = 10
SETDefault !V4 -SR RingNumber = None
SETDefault !V5 -SR RingNumber = 20
SETDefault !V2 -SR RouteDiscovery = None
SETDefault !V3 -SR RouteDiscovery = LLC2
SETDefault !V4 -SR RouteDiscovery = None
SETDefault !V5 -SR RouteDiscovery = LLC2
```

The ring number and route discovery configuration to be used at the leaf node must be explicitly configured on the virtual ports running both IBM and non-IBM traffic.

Though ring numbers are mapped to the virtual ports, the ring numbers are actually used for the leaf node token ring LANs.

Proceed to the next section to complete this procedure.

Enabling the Ports and Sending Leaf Nodes the Configuration Information. This procedure enables the ports and transmits the needed configuration information to the leaf nodes. This is the last step for configuring dual PVCs in both Ethernet and Token Ring environments.

Enable the virtual ports, by entering:

```
SETDefault !V2 -PORT CONTrol = Enable
SETDefault !V3 -PORT CONTrol = Enable
SETDefault !V4 -PORT CONTrol = Enable
SETDefault !V5 -PORT CONTrol = Enable
```

Enabling these virtual ports triggers the transfer of the configuration information from the central node to the leaf nodes.

Verifying the Dual PVC Configuration

To verify that dual PVCs are correctly configured from the leaf node and the central node, follow these steps:

1. Verify that dual PVCs are configured from the leaf node using:

   ```
   SHow [!<port> | !*] -FR DLciStat
   ```

   This parameter displays the DLCI status and statistics for all active Frame Relay ports.

2. Verify that traffic is being transmitted through the ports using:

   ```
   SHow -SYS STATistics -LLC2
   ```

3. Verify that dual PVCs are configured from the central node using:

   ```
   SHow [!<port> | !*] -BCN IbmStatus
   ```

   When the -BCN CONTrol parameter has been set to IbmTraffic, this parameter displays the status of Boundary Routing ports over which IBM traffic is running.

Configuring Network Resiliency

In a Boundary Routing topology, you can protect the operation of mission-critical applications if a failure occurs.

If you are configuring disaster recovery over Frame Relay on the peripheral node of a Boundary Routing environment, do not configure network resiliency.

If you are configuring network resiliency on a boundary router leaf network, make sure the PORT OWNer parameter and the PATH LineType parameter are set to values other than Auto. The port or path will not come up until you change those parameters to values other than Auto.

You can implement network resiliency in two different ways: you can configure a backup or *redundant link* between a central and peripheral node or a backup or *redundant route* to an alternate central node as shown in Figure 32-3.

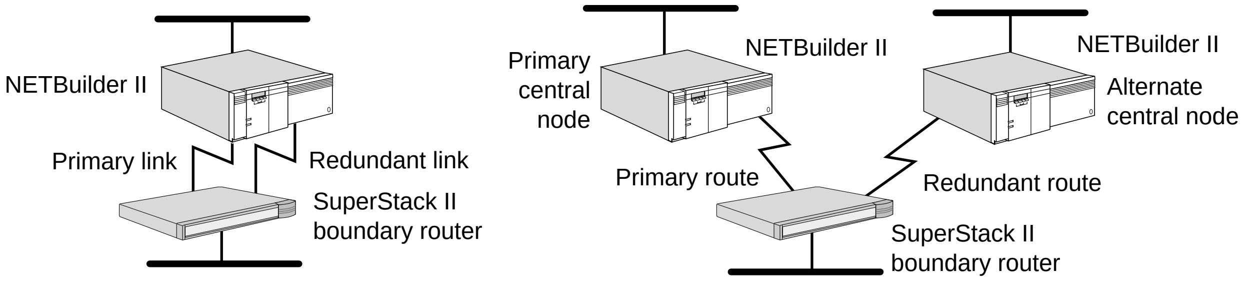


Figure 32-3 Different Types of Network Resiliency

A redundant link provides a backup link if the primary link fails. A redundant route provides a backup route to an alternate central node if the primary route or the primary central node fails. The bandwidth management feature introduced in software version 9.1 views line resources as unrestricted, available resources, or resources configured for a specific function, such as disaster recovery only, instead of as primary and secondary lines. Refer to Chapter 37 for more information.

When power is turned on, the auto startup feature brings up one active path only. It does not bring up a second path for network resiliency. You need to configure the second path for network resiliency using the procedure in the following section.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Determine how you want to implement network resiliency.

  You must determine if you want to configure a redundant link between your central and peripheral nodes or if you want to configure a redundant route to an alternate central node. You must also determine which wide area networking protocol you will be using over the redundant link or route.
- Refer to "Network Resiliency" on page 32-47 to familiarize yourself with network resiliency and any configuration steps particular to your network resiliency implementation.
- Make sure you have fully configured the Boundary Routing port that you are planning the redundancy for (the primary link or route) according to instructions in "Configuring Basic Boundary Routing" on page 32-1.

Procedure

Use the following procedure to configure the redundant link or route in your Boundary Routing topology. The steps apply to both types of network resiliency configurations unless specifically called out. Complete the step on the central node unless specifically instructed to complete the step on both the central and peripheral nodes or the peripheral node only. Refer to *Reference for NETBuilder Family Software* for general information on parameters used in the following procedure.

Some of the steps in the following procedure require that you perform configuration on the peripheral node (boundary router). You can access the Boundary Routing software through the System Configuration menu, an interface that prompts you to complete the tasks. If you cannot complete the steps outlined in this procedure using the System Configuration menu, you can exit the interface and access a command line user interface by selecting Quit from the System Configuration menu. To return to the menu, enter the InStall command. For more information on the Boundary Routing user interface, refer to the documentation that shipped with your boundary router.

To configure network resiliency, follow these steps:

1 Configure the wide area port and path according to instructions in Chapter 1.

If you are configuring a redundant link that will be running PPP, make certain that you assign the primary and secondary paths to one port using the -PORT PAths parameter.

2 If you are using Frame Relay or X.25, create a virtual port for each leaf network that is attached to the Frame Relay or X.25 cloud according to instructions in Chapter 1.

3 Configure the wide area or virtual port for the Boundary Routing feature according to “Configuring Basic Boundary Routing” on page 32-1.

4 Complete the following steps at both ends of the redundant link or route. Refer to “Network Resiliency” on page 32-47 for information on parameter settings for specific network resiliency configurations.

a Assign an owner to the wide area port using:

```
SETDefault !<port> -PORT OWNer = PPP | FrameRelay | X25
```

b Set the line type on the wide area path using:

```
SETDefault !<path> -PATH LineType = Leased | Dialup | Auto
```

c Set the attributes of the wide area path using:

```
SETDefault !<path> -PATH DialCONTrol = (UnReSTricted |
DisasterRecovery | NoDisasterRecovery, [Answer |NoAnswer],
[Originate | NoOriginate])
```

5 If you are configuring a redundant PPP link as a backup to a primary PPP link, follow these steps:

a Enable bandwidth-on-demand on both ends of the wide area link using:

```
SETDefault !<port> -PORT DialInitState = (DialonDemand)
```

a Enable disaster recovery on both ends of the wide area link using:

```
SETDefault !<port> -PORT DialCONTrol = (DisasterRcvry)
```

b If in an IBM Boundary Routing topology, set the value of the -LLC2 RetryCount to 20 on the Boundary Routing port using:

```
SETDefault !<port> -LLC2 RetryCount = 20
```

Increasing the value of this parameter from its default setting to 20 ensures that the retry timer will not time out and bring the circuit down, if the primary link goes down and the secondary line comes up.

6 If you are configuring a redundant PPP link as a backup to a primary Frame Relay link, configure the same address or network number (for example, an IP address or an IPX network number) on both wide area ports of the central node.

For more information on how to do this, refer to “Frame Relay Environment” on page 32-49 and “IBM Environment” on page 32-50.

7 If you are configuring a redundant route to an alternate central node, follow these steps:

a Enable automatic dialing at the peripheral node end of the redundant route only, using:

```
SETDefault !<port> -PORT AutoDial = Enabled
```

Setting the value of this parameter to Enabled allows the peripheral node to automatically dial the alternate central node if the primary route or primary central node fails.

b Configure the same address or network number (for example, an IP address or an IPX network number) on the wide area ports of both central nodes.

Refer to the appropriate bridging or routing chapter, for example, the chapter on IPX routing, for information on configuring an address or network number. For more information on why you need to complete this step, refer to "Primary and Alternate Central Node Configuration" on page 32-58.

c If you plan to bridge or route AppleTalk, IP, or IPX in your Boundary Routing topology, enable the central MAC address on the wide area ports of both central nodes, using:

```
SETDefault !<port> -BCN CONTrol = CentralMac
```

You may also need to configure the wide area port on the alternate central node with the same bridging or routing attributes as the primary central node. To make this determination, refer to "Using the Central MAC Address" on page 32-59. If you determine that you need to do this, refer to the appropriate bridging or routing chapter, for example, the chapter on IPX routing, for information on configuring these attributes.

d Disable the wide area port on the alternate central node using:

```
SETDefault !<port> -PORT CONTrol = Disabled
```

Disabling this port prevents it from coming up before the primary central node port. It also can control which central node is primary and which is alternate instead of the software negotiation making the decision.

If the primary route or primary central node fails, you can enable the wide area port on the alternate central node using:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

e If you are configuring a redundant PPP route as a backup to a primary X.25 route, you must make certain that if the X.25 route goes down at the peripheral node, the virtual port associated with this route is disabled at the central node before the redundant PPP route activates.

For more information on how to do this, refer to "X.25 Environment" on page 32-55 and "IBM Environment" on page 32-56.

How Boundary Routing System Architecture Works

Boundary Routing system architecture can interconnect networks using wide area links. Boundary Routing is ideally suited for environments that require a large number of small remote office networks (leaf networks) to be connected to a central office (central site network).

Where Can Boundary Routing Be Used?

To implement Boundary Routing system architecture in a network topology, a central node (which provides the routing function) and peripheral nodes must be present.

The central node must be a NETBuilder II bridge/router, or a model 227, 327, 427, 527 SuperStack II NETBuilder bridge/router, or a model 147 OfficeConnect NETBuilder bridge/router. The central node can be configured as a bridge or a router.

Table 32-1 provides information on the platforms that can be used as a central node.

Table 32-1 Central Node Information

Platform	Number of Peripheral Nodes Supported	Protocols Supported
NETBuilder II bridge/router	Up to 40 or 75 peripheral nodes (software packages SW/NBII-CP and SW/NBII-FF support 75 peripheral nodes). No restrictions as to the number of peripheral nodes the NETBuilder II bridge/router supports over PPP, Frame Relay, or X.25.	Bridging and all routing protocols
Model 227* SuperStack II bridge/router	Up to 10 peripheral nodes over Frame Relay. Up to 3 peripheral nodes using PPP over 56/64 KB leased or dial-up lines. Must use a SuperStack II NETBuilder boundary router that supports Ethernet as the LAN media type as a peripheral node.	Bridging and all routing protocols except APPN
Model 327* SuperStack II bridge/router	Up to 10 peripheral nodes over Frame Relay. Up to 3 peripheral nodes over PPP. Must use a SuperStack II NETBuilder boundary router that supports token ring as the LAN media as a peripheral node.	Source route bridging and all routing protocols except APPN. Does not support transparent bridging.
Model 427* SuperStack II bridge/router and Model 147 OfficeConnect bridge/router	Up to 10 peripheral nodes over Frame Relay. Up to 3 peripheral nodes using PPP over 56/64 KB leased or dial-up lines. Must use a SuperStack II NETBuilder boundary router that supports Ethernet as the LAN media type as a peripheral node.	Bridging and all routing protocols except APPN
Model 527* SuperStack II bridge/router	Up to 10 peripheral nodes over Frame Relay. Up to 3 peripheral nodes over PPP. Must use a SuperStack II NETBuilder boundary router that supports token ring as the LAN media as a peripheral node.	Source route bridging and all routing protocols except APPN. Does not support transparent bridging.

* These platforms do not support the IBM network management application LAN Net Manager (LNM).

The peripheral node can be any 3Com platform that runs the Boundary Routing system architecture software. For example, the peripheral node can be one of the following devices:

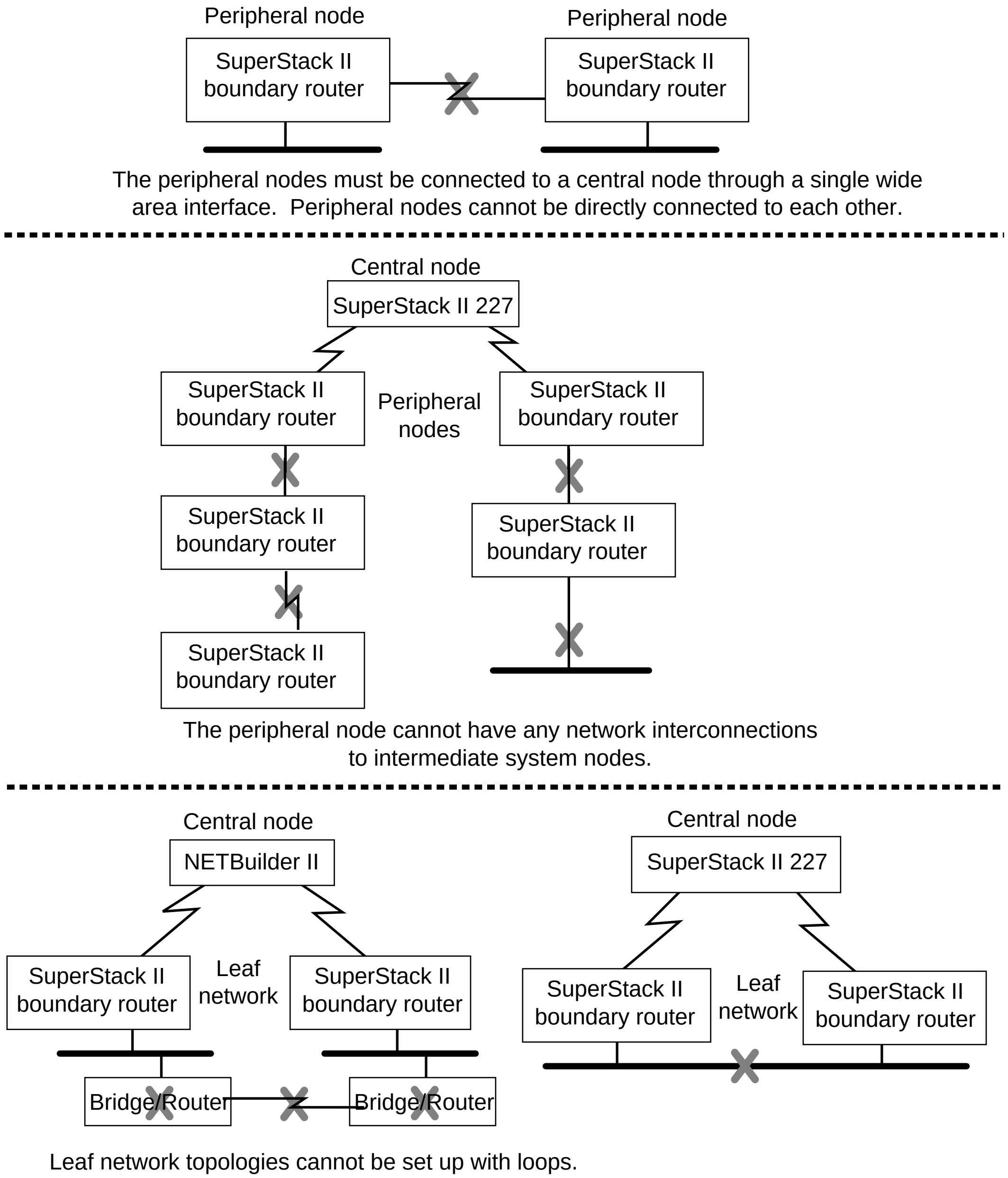
- SuperStack II and OfficeConnect NETBuilder boundary router
- LinkBuilder Ether Connect System (ECS) Remote Control Module (runs Boundary Routing over PPP only)

In Boundary Routing network topologies, the following rules apply:

- The remote office networks must be leaf networks. A single (one and only one) active network interconnection from the leaf network to the central node is permitted.
- The peripheral node and the central node must be connected over a point-to-point serial link or a virtual circuit.
- The central node is the bridge/router in a Boundary Routing environment. Any client configuration on the leaf network that requires addressing the router needs to use the address information pertaining to the central node, not the peripheral node.

A backup link can be configured for bandwidth-on-demand, dial-on-demand, or network resiliency. A backup route to an alternate central node can be configured for network resiliency.

Figure 32-4 shows illegal topologies in which Boundary Routing cannot operate.


Peripheral node
SuperStack II boundary router
Peripheral node
SuperStack II boundary router
Central node
SuperStack II 227
SuperStack II boundary router
Peripheral nodes
SuperStack II boundary router
SuperStack II boundary router
SuperStack II boundary router
SuperStack II boundary router
Central node
NETBuilder II
SuperStack II boundary router
Leaf network
SuperStack II boundary router
Bridge/Router
Bridge/Router
Central node
SuperStack II 227
SuperStack II boundary router
Leaf network
SuperStack II boundary router

The peripheral nodes must be connected to a central node through a single wide area interface. Peripheral nodes cannot be directly connected to each other.

The peripheral node cannot have any network interconnections to intermediate system nodes.

Leaf network topologies cannot be set up with loops.

Figure 32-4 Illegal Boundary Routing Topologies

Typical Boundary Routing Environment

This section provides examples of the following types of Boundary Routing environments:

- Non-IBM
  - Using a NETBuilder II bridge/router as a central node
  - Using a model 227 or 427 SuperStack II NETBuilder bridge/router as a central node
- IBM
  - Using a NETBuilder II bridge/router as a central node

- Using a NETBuilder II bridge/router as a regional central node
- Using a model 327 or 527 SuperStack II NETBuilder bridge/router as a central node
- APPN

BSC traffic is not supported in Boundary Routing environments.

Non-IBM Environment Using a NETBuilder II Bridge/Router

Figure 32-5 shows a typical non-IBM environment in which Boundary Routing system architecture is used. A NETBuilder II bridge/router is used as the central node.

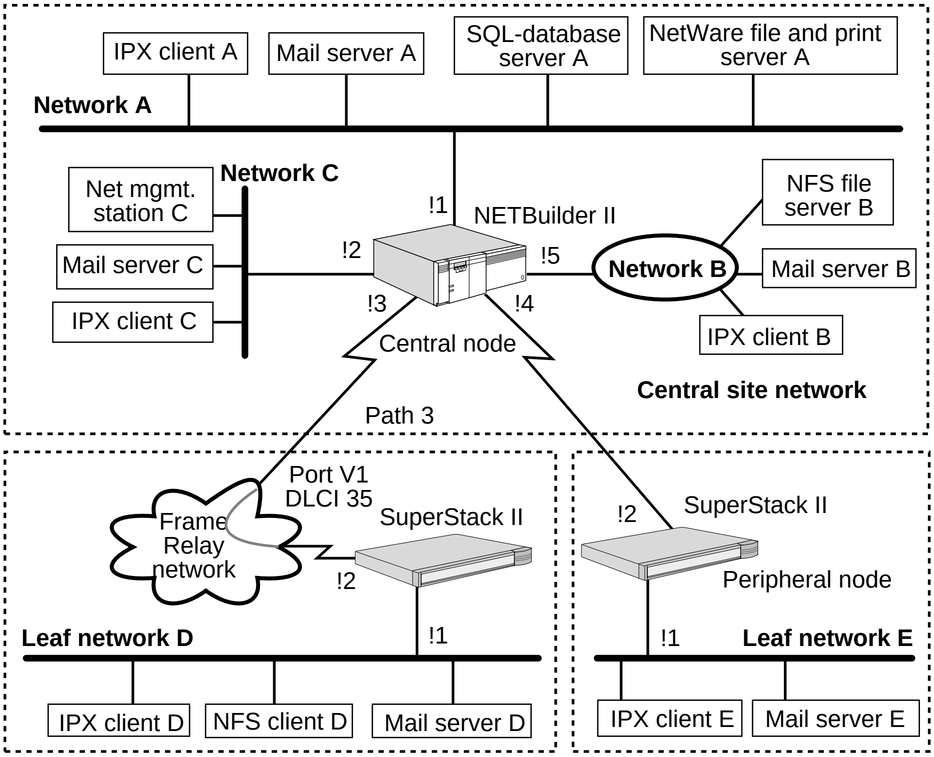


Figure 32-5 Typical Non-IBM Boundary Routing Environment Using NETBuilder II

In this figure, the central site network comprises networks A, B, and C, and two leaf networks D and E. The leaf networks are connected to the central site network using Boundary Routing system architecture.

Network A contains a mail server for electronic mail exchange, an Structured Query Language (SQL) database server, and a NetWare file and print server. All hosts on this network use IPX as the underlying network protocol.

Network B is a multiprotocol token ring network containing an Network File System (NFS) file server and a mail server for electronic mail exchange. The file server uses IP as the underlying network protocol, and the mail server uses IPX.

Network C is also a multiprotocol network containing a mail server and a network management station. The mail server uses IPX as the underlying protocol; the network management station provides SNMP and Telnet, both of which use IP as the underlying protocol. The network management station manages the central node and the peripheral nodes.

Leaf network D, which uses a virtual port and is connected across a Frame Relay network, requires access to the NetWare file and print server, the SQL database server, and the NFS file server. Leaf network D also exchanges electronic mail with the central site network and leaf network E. NetWare, SQL, and electronic mail are run over IPX. NFS is run over IP.

Leaf network E requires access to the NetWare file and print server, and exchanges electronic mail with the central site network and leaf network D. NetWare and electronic mail are run over IPX.

If you use Boundary Routing system architecture to interconnect these networks, the following configuration steps are required:

- On the central site:
  - Configure the WAN links.
  - Enable Boundary Routing on port 4 and virtual port V1.
  - Configure RARP to assign IP addresses to the peripheral nodes.
  - Assign IP network addresses for networks B, C, and D.
  - Assign IPX network addresses for networks A, B, C, D, and E.
  - Configure ports 1, 2, 4, and 5 and virtual port V1 of the central node to route IPX.
  - Configure ports 2 and 5 and virtual port V1 of the central node to route IP.
  - Configure the remote LAN type on port 4 and virtual port V1.
- On the peripheral nodes:

  In most cases, no configuration is necessary on the peripheral node. Refer to the documentation that accompanies your peripheral node to determine if configuration is necessary.

When using Boundary Routing system architecture to achieve connectivity, fewer configuration steps are required at each leaf network, resulting in administrative savings and troubleshooting costs if the number of leaf networks is large.

Non-IBM Environment Using a SuperStack II Bridge/Router model 227 or 427

Figure 32-6 and Figure 32-7 show typical non-IBM environments in which Boundary Routing system architecture is used. In Figure 32-6 a model 227 SuperStack II bridge/router is the central node, while in Figure 32-7 model 427 SuperStack II bridge/router is the central node.

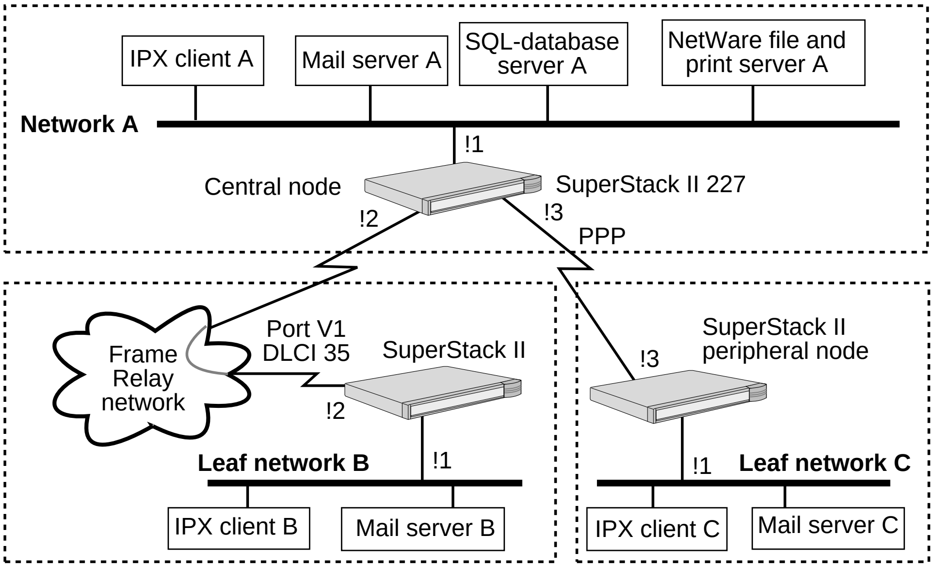


Figure 32-6 Typical Non-IBM Boundary Routing Environment Using Model 227 SuperStack II Bridge/Router

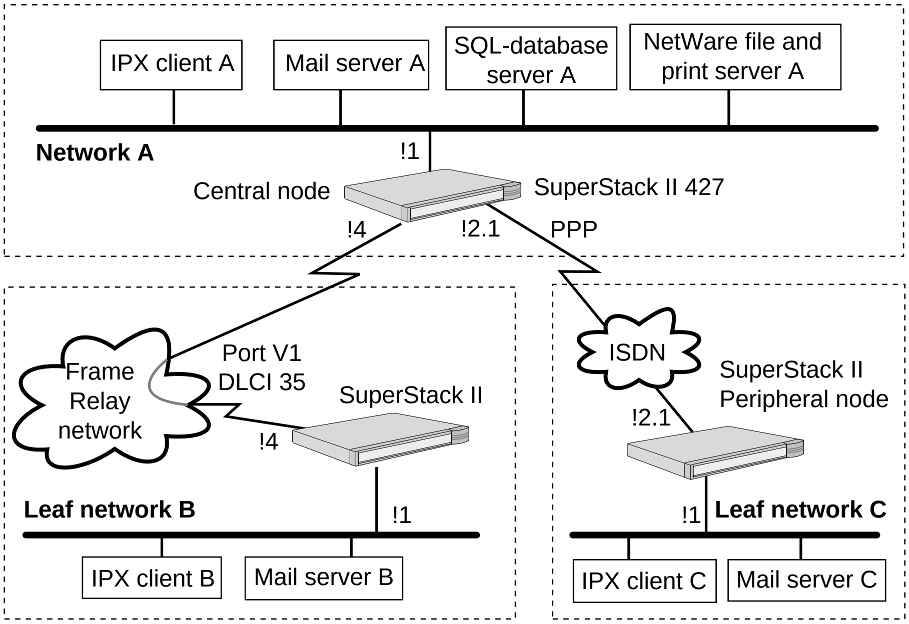


Figure 32-7 Typical Non-IBM Boundary Routing Environment Using Model 427 SuperStack II Bridge/Router

For information on the number of peripheral nodes that the model 227 and 427 SuperStack II bridge/routers support, refer to Table 32-1.

In these figures, the central site network is network A and the leaf networks are B and C. The leaf networks are connected to the central site network using Boundary Routing system architecture. IPX is the underlying network protocol in these topologies.

Network A contains a mail server for electronic mail exchange, an SQL database server, and a NetWare file and print server.

Leaf network B, which uses a virtual port and is connected across a Frame Relay network, requires access to the NetWare file and print server and the SQL database server. Leaf network B also exchanges electronic mail with the central site network and leaf network C.

In Figure 32-6, network C is connected across a PPP network while in Figure 32-7, it is connected across an ISDN network. Leaf network C requires access to the NetWare file and print server, and exchanges electronic mail with the central site network and leaf network B.

If you use Boundary Routing system architecture to interconnect these networks, the following configuration steps are required:

- On the central site:
  - Configure the WAN links, including the ISDN interface on model 427 bridge/router.
  - Enable the Boundary Routing feature on ports V1 and 3 in Figure 32-6 and ports V1 and 2.1 in Figure 32-7.
  - Assign IPX network addresses for networks A, B, and C.
  - Configure ports 1, 3, and V1 in Figure 32-6 and ports 1, 2.1, V1 in Figure 32-7 of the central node to route IPX.
  - Configure the remote LAN type on ports V1 and 3 in Figure 32-6 and ports V1 and 2.1 in Figure 32-7.
- On the peripheral nodes:

  In most cases, no configuration is necessary on the peripheral node. Refer to the documentation that accompanies your peripheral node to determine if configuration is necessary.

When using Boundary Routing system architecture to achieve connectivity, fewer configuration steps are required at each leaf network, resulting in administrative savings and troubleshooting costs if the number of leaf networks is large.

IBM Environment Using a NETBuilder II Bridge/Router as a Central Node

Although an SNA example is used, the information in this section applies to both SNA and NetBIOS topologies except where specifically noted.

Figure 32-8 shows an SNA Boundary Routing topology with a NETBuilder II bridge/router as a central node.

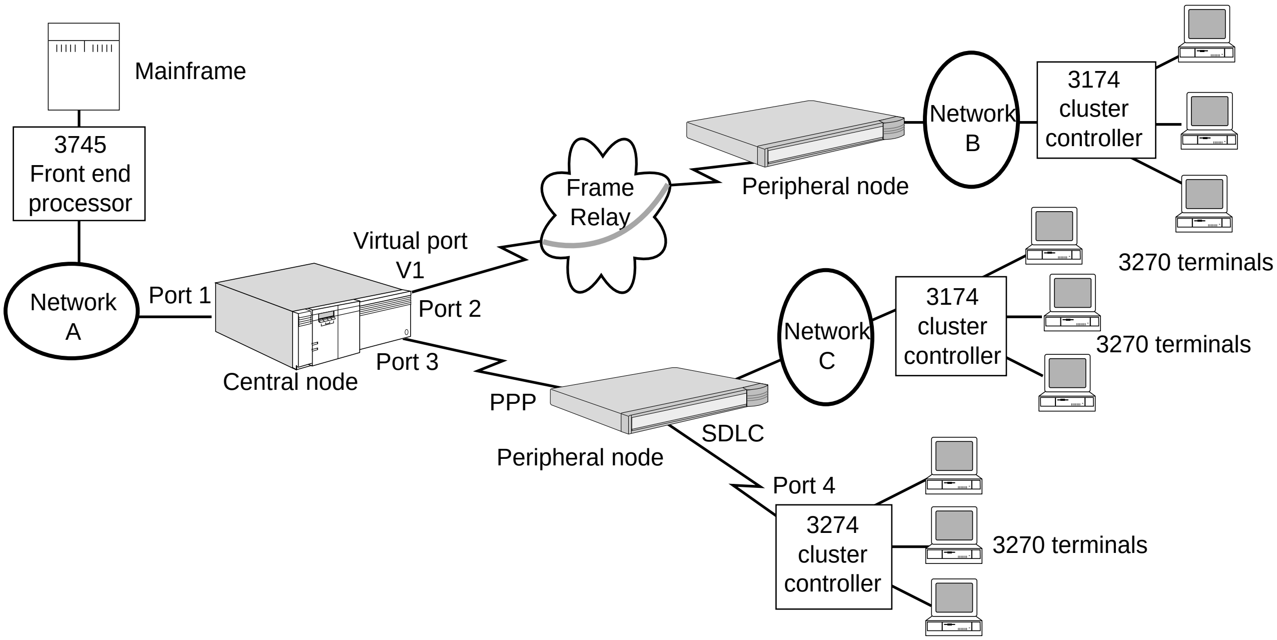


Figure 32-8 SNA Boundary Routing Topology: NETBuilder II As Central Node

In this figure, the central site network is network A and the leaf networks are networks B and C. The leaf networks are connected to the central site network using Boundary Routing system architecture.

Network A contains an IBM 3745 front-end processor (FEP) and a mainframe computer. Networks B and C contain remote 3174 cluster controllers and 3270 terminals, which must periodically access applications on the mainframe computer using SNA LLC2 sessions or SDLC sessions.

If you use Boundary Routing system architecture to interconnect these networks, the following configuration steps are required:

- On the central site:
  - For NetBIOS topologies, enable Boundary Routing of NetBIOS traffic.
  - Configure the LLC2 data link interface on port 1.
  - Enable source route bridging on port 1.
  - Assign unique ring number on port 1.
  - Configure the WAN links.
  - Enable Boundary Routing on port 3 and virtual port V1.
  - Configure the remote LAN type on port 3 and virtual port V1.
  - Set the address format for ARP to noncanonical for port 3 and virtual port V1.
  - Enable Boundary Routing of IBM traffic on port 3 and virtual port V1.
- On the peripheral nodes:
  - In most cases, no configuration is necessary on the peripheral node. Refer to the documentation that accompanies your peripheral node to determine if configuration is necessary.

- If you are configuring SDLC as a client protocol over the boundary router link, you must configure the SDLC port and path attributes, as well as the CU information, on the peripheral node. For more information, refer to Chapter 22.

  For sending SDLC traffic over a boundary router link, no additional configuration is required at the central node.

When using Boundary Routing system architecture to achieve connectivity, fewer configuration steps are required at each leaf network, resulting in administrative savings and troubleshooting costs if the number of leaf networks is large.

IBM Environment Using a NETBuilder II Bridge/Router as a Regional Central Node

Although an SNA example is used, the information in this section applies to both SNA and NetBIOS topologies except where specifically noted.

Figure 32-9 shows Boundary Routing system architecture in an SNA environment with a NETBuilder II bridge/router as a regional central node.

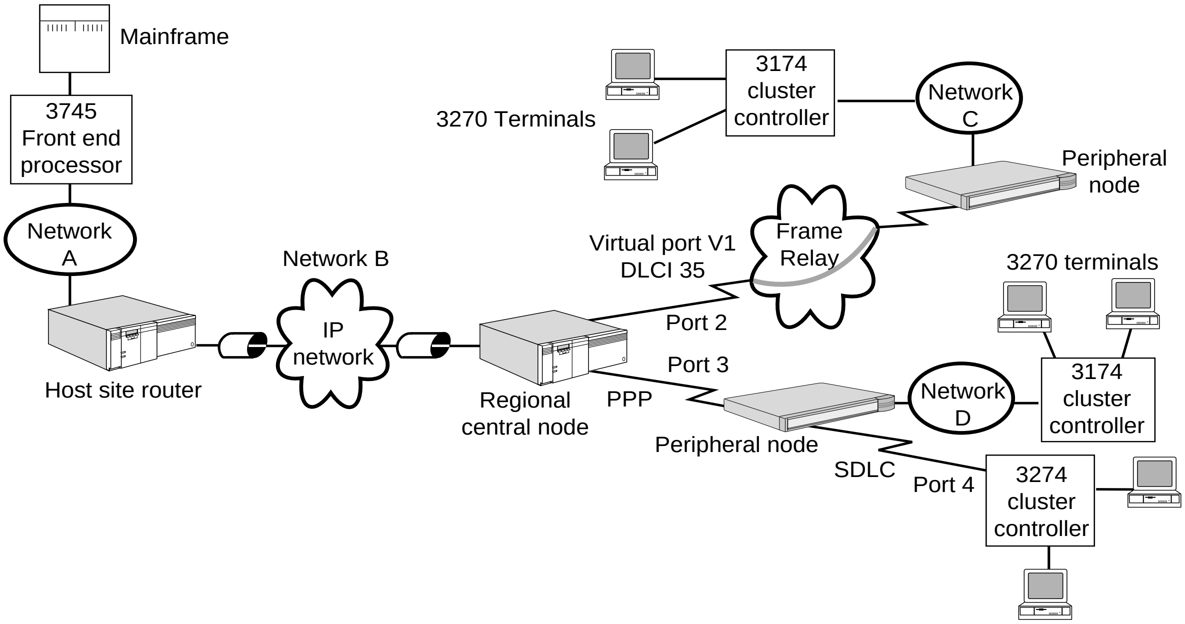


Figure 32-9 SNA Boundary Routing Topology: NETBuilder II As Regional Central Node

In this figure, the central site networks are networks A and B and the leaf networks are networks C and D. The leaf networks are connected to the central site network using Boundary Routing system architecture.

Network A contains an IBM 3745 FEP and a mainframe computer. Networks C and D contain remote 3174 cluster controllers and 3270 terminals, which must periodically access applications on the mainframe computer using SNA LLC2 sessions or SDLC sessions.

This topology differs from the traditional Boundary Routing topology because packets that passed between the mainframe and the terminals and vice versa must additionally traverse network B, which is an IP internetwork. DLSw enabled on both the regional central node and the host site router allows the SNA traffic to traverse the IP internetwork. For more information on DLSw, refer to Chapter 24.

If you use Boundary Routing system architecture to interconnect these networks, the following configuration steps are required:

- On the host site:
  - For NetBIOS topologies, enable Boundary Routing of NetBIOS traffic.
  - Configure the LLC2 data link interface on LAN port.
  - Enable source route bridging on LAN port.
  - Assign unique ring number on LAN port.
  - Configure DLSw on WAN port.
- On the regional central site:
  - Configure the WAN links.
  - Enable Boundary Routing on port 3 and virtual port V1.
  - Configure the remote LAN type on port 3 and virtual port V1.
  - Set the address format for ARP to noncanonical for port 3 and virtual port V1.
  - Enable Boundary Routing of IBM traffic on port 3 and virtual port V1.
  - Configure DLSw on the port that interfaces network B, the IP internetwork.
- On the peripheral nodes:
  - In most cases, no configuration is necessary on the peripheral node. Refer to the documentation that accompanies your peripheral node to determine if configuration is necessary.
  - If you are configuring SDLC as a client protocol over the boundary router link, you must configure the SDLC port and path attributes and the CU information on the peripheral node. For more information, refer to Chapter 22.

    For sending SDLC traffic over a boundary router link, no additional configuration is required at the central node.

When using Boundary Routing system architecture to achieve connectivity, fewer configuration steps are required at each leaf network, resulting in administrative savings and troubleshooting costs if the number of leaf networks is large.

IBM Environment Using a SuperStack II NETBuilder Bridge/Router Model 327 or 527 As a Central Node

Figure 32-10 and Figure 32-11 show SNA Boundary Routing topologies with model 327 and 527 SuperStack II NETBuilder bridge/routers as central nodes, respectively.

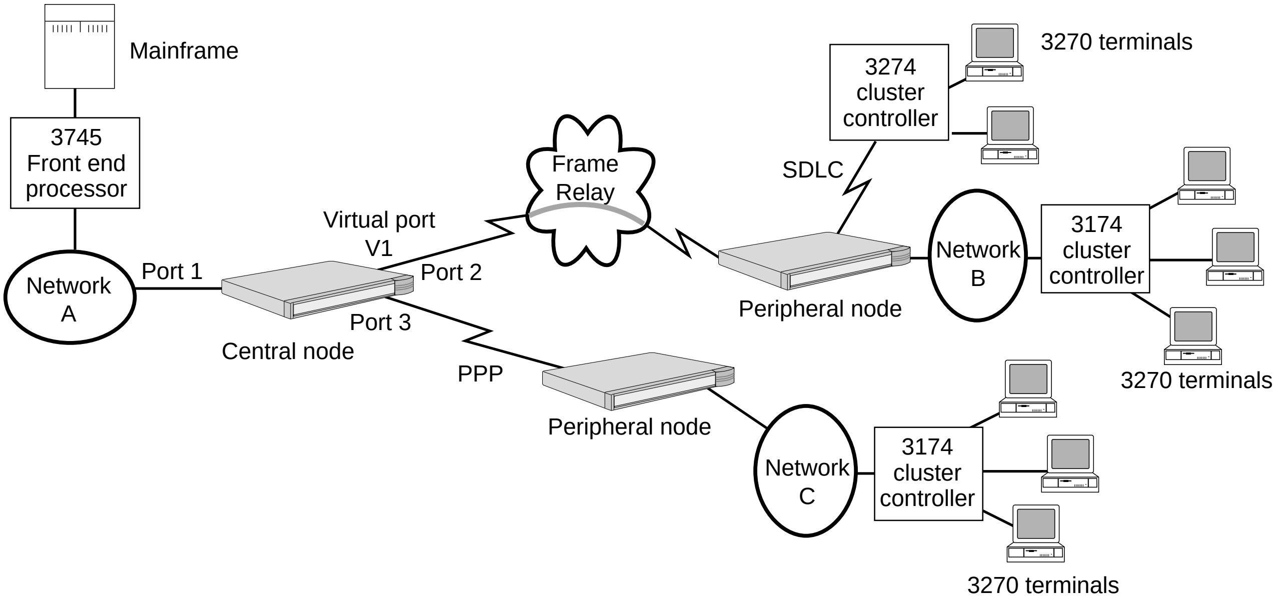


Figure 32-10 SNA Boundary Routing Topology: Model 327 SuperStack II Bridge/Router as Central Node

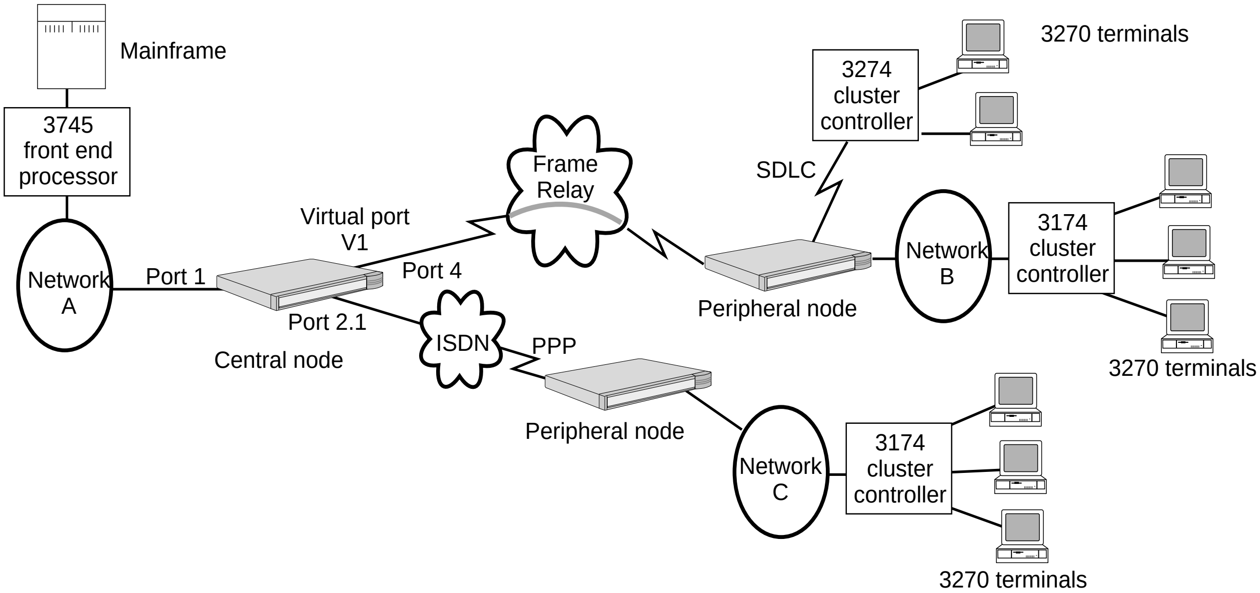


Figure 32-11 SNA Boundary Routing Topology: Model 527 SuperStack II Bridge/Router as Central Node

In both figures, the central site network is network A and the leaf networks are networks B and C. The leaf networks are connected to the central site network using Boundary Routing system architecture.

Network A contains an IBM 3745 FEP and a mainframe computer. Networks B and C contain remote 3174 cluster controllers and 3270 terminals, which must periodically access applications on the mainframe computer using SNA LLC2 sessions. The boundary router that is connected to network B also provides a connection to an SDLC device, in this case, a remote 3274 cluster controller with 3270 terminals attached. The terminals must periodically access applications on the mainframe computer using SDLC sessions.

If you use Boundary Routing system architecture to interconnect these networks, the following configuration steps are required:

- On the central site:
    - Configure the LLC2 data link interface on port 1.
    - Enable source route bridging on port 1.
    - Assign unique ring number on port 1.
    - Configure the WAN links, including the ISDN interface on model 527.
    - Enable Boundary Routing on ports V1 and 3 in Figure 32-10 and on ports V1 and 2.1 in Figure 32-11.
    - Configure the remote LAN type on ports V1 and 3 in Figure 32-10 and on ports V1 and 2.1 in Figure 32-11.
    - Set the address format for ARP to noncanonical on ports V1 and 3 in Figure 32-10 and on ports V1 and 2.1 in Figure 32-11.
    - Enable Boundary Routing of IBM traffic on ports V1 and 3 in Figure 32-10 and on ports V1 and 2.1 in Figure 32-11.
- On the peripheral nodes:
    - In most cases, no configuration is necessary on the peripheral node. Refer to the documentation that accompanies your peripheral node to determine if configuration is necessary.
    - If you are configuring SDLC as a client protocol over the boundary router link, you must configure the SDLC port and path attributes and CU information on the peripheral node. For more information, refer to Chapter 22.

      For sending SDLC traffic over a boundary router link, no additional configuration is required at the central node.

When using Boundary Routing system architecture to achieve connectivity, fewer configuration steps are required at each leaf network, resulting in administrative savings and troubleshooting costs if the number of leaf networks is large.

APPN Topology

For more information on Boundary Routing in an APPN environment, refer to "Configuring APPN for Boundary Routing" on page 10-29.

SDLC Over Boundary Router Links

You can attach an SNA/SDLC system to a peripheral node, and send SDLC traffic over a boundary router link. To do this, no additional configuration of WAN ports at the central node is required. All SDLC configuration required for routing SDLC traffic over a boundary router link is performed at the peripheral node.

If you plan to configure SDLC on a SuperStack II bridge/router, refer to the appropriate SuperStack II Ethernet or Token Ring guide.

Boundary Routing Features

The Boundary Routing software provides the following advantages when connecting remote office networks:

- Simplifies network administration through configuration at the central node.
- Reduces WAN usage costs through smart filtering, dial-on-demand, payload or data compression, and data exchange with specific peers in an IBM Boundary Routing topology.
- Provides higher reliability through local termination and the automatic prioritization of IBM traffic in an IBM Boundary Routing topology.
- Provides continuous operation with a dial-up backup line for disaster recovery and bandwidth-on-demand, and a mechanism for constructing resilient networks.

Simplified Network Administration

In remote office network environments, Boundary Routing system architecture can be used to construct a manageable network topology and simplify network administration. The topology is manageable because routing is used to switch packets between the leaf networks and the central site network. This allows for greater flexibility in network segmentation and better control over the traffic. Administration is simplified because, unlike traditional routing where the administrative burden is on both ends of the interconnection, most of the administration is performed at the central site network. A few, simple configurations may be needed at the leaf networks. The leaf networks often may require no configuration at all.

Reduced WAN Usage Costs

The following features reduce WAN usage costs.

Smart Filtering. Smart filtering reduces the cost associated with WAN lines by minimizing the number of packets that must be sent over the WAN link, particularly overhead traffic such as topology-maintenance messages. This feature is called smart filtering because the filtering decisions are automatically made by the central node in the Boundary Routing system based on the configuration at the central site and the traffic flow from the remote device. The filtering actions are then taken by the peripheral node.

You can use smart filtering in an IPX environment and an extension of smart filtering in an IBM environment called *smart polling*.

You can use smart filtering to do the following if you are using the IPX Protocol in your Boundary Routing topology:

- Eliminate non-IBM traffic belonging to protocol islands that are confined to a leaf network from the WAN link when the central node is strictly routing on the Boundary Routing port (bridging has been disabled).

  As shown in Figure 32-12, the VINES, AppleTalk, and LAT clouds represent protocol islands and have *no* connection needs with other leaf networks or the central node. Protocol islands consist of network topologies that are always confined to a single leaf network and have no interconnection needs with other leaf networks or the central node. Smart filtering prevents traffic generated by these protocol islands from being forwarded over the WAN link because the central node instructs the peripheral node to filter.

If bridging is enabled on the Boundary Routing port, all traffic, including protocol island traffic, is forwarded.

- Eliminate periodic rebroadcasts of IPX NRIP and SAP updates from the central node and NetWare servers on the leaf networks on the WAN link without requiring static configurations of routes or services at either end.

To enable smart filtering, set the value of the -BCN CONTrol parameter to SmartFiltering.

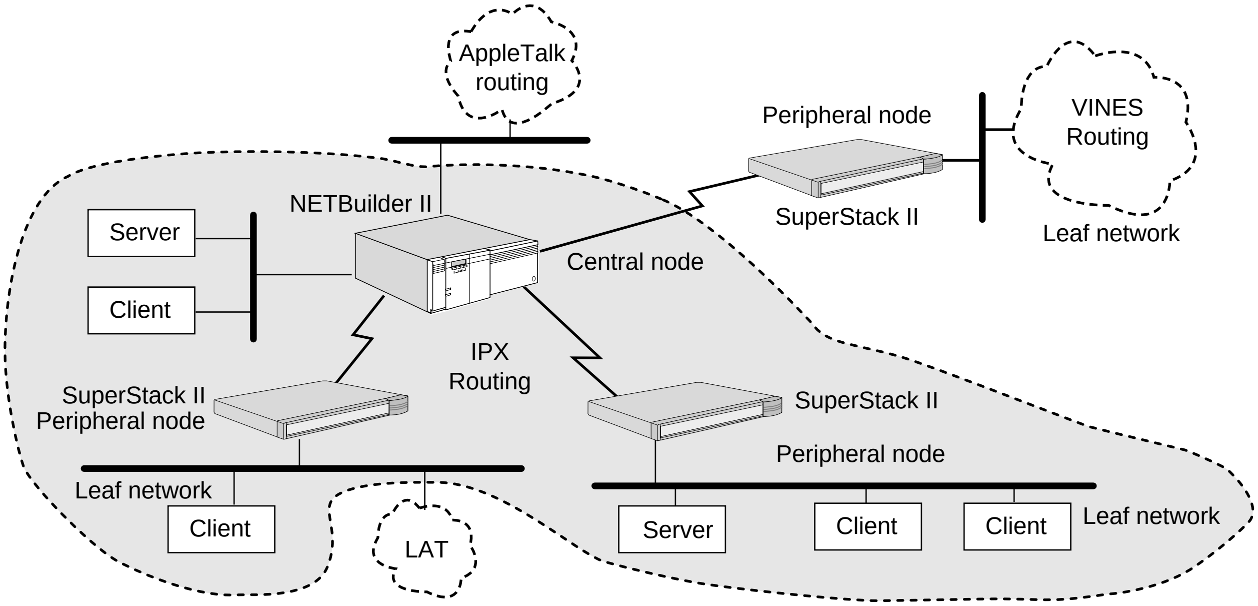


Figure 32-12 Protocol Islands in Boundary Routing Environments

The extension of smart filtering for IBM topologies called smart polling is available in software version 8.2 and later.

SNA and NetBIOS use the data link protocol LLC2. After a user at a terminal initiates an LLC2 session with an SNA or NetBIOS host, polling packets are exchanged continually between the central node and the peripheral node during the session to indicate that the LLC2 session is still alive. If multiple LLC2 sessions between the host and clients are running, the number of polling packets exchanged by the central and peripheral nodes becomes significant.

Smart polling reduces the number of polling packets exchanged between the central and peripheral nodes. For example, suppose that four LLC2 sessions are running simultaneously between a mainframe computer and four different terminals on the same leaf network. Instead of the central and peripheral nodes exchanging polling packets for each session, the central and peripheral nodes assume that a poll reply for one session indicates that the other three sessions are still alive.

Smart polling is effective when one or multiple LLC2 sessions are running simultaneously between a host and terminals on the same leaf network. In fact, the more sessions that are running simultaneously, the greater the reduction of the number of polling packets sent over the WAN link.

To activate smart polling, set the value of the -BCN CONTrol parameter to IbmTraffic.

Smart Filtering for Boundary Routing over X.25. If smart filtering is operating on a peripheral node, and the link between the peripheral node and the X.25 packet-switched network is inoperable, then the central node and the peripheral node can become unsynchronized. The result is that the NetWare servers at the remote site are not refreshed with information about other NetWare servers located at the central site. This condition can be corrected by re-enabling the affected virtual port on the central site router.

Smart Filtering and SAP. The smart filtering feature for ports using Boundary Routing software cannot be used when the size of the SAP table in your network exceeds 400 services.

Before enabling smart filtering, check the size of your SAP information table by entering:

```
SHow -IPX AllServers
```

If the number displayed is greater than 400, do not attempt to enable smart filtering on any of the ports that use Boundary Routing unless you use the Advertise Policy parameter to control the list of SAP entries.

If there are only NetWare clients at the remote site and no NetWare servers, another way to reduce SAP traffic over ports using Boundary Routing is to turn SAP talk off by using the CONTrol parameter in the SAP Service on the WAN ports at the central site. Leave the SAP talk on LAN ports. To turn SAP talk off, use:

```
SETDefault !<port> -SAP CONTrol = NoTalk
```

If the remote site has a server, you can run NLSP on the Boundary Routing port between the router and the server. NLSP is supported on version 8.0 and higher with Netware 3.12, 4.01, and 4.1.

Smart Filtering in a Boundary Routing Topology. If you have enabled the smart filtering feature in your Boundary Routing topology and have subsequently added or deleted IPX services on a server on a currently active leaf network without restarting the server, you must disable then re-enable smart filtering on the central node. Re-enabling smart filtering on the central node enables it to update the services learned from the remote leaf network.

To disable smart filtering, use:

```
SETDefault !<port> -BCN CONTrol = NoSmartFiltering
```

To enable smart filtering, use:

```
SETDefault !<port> -BCN CONTrol = SmartFiltering
```

Disabling Smart Filtering. If you want to disable the smart filtering feature and have enabled the smart filtering feature in your Boundary Routing topology, you should re-enable the port after the smart filtering feature is disabled. To re-enable the port after smart filtering is disabled, use:

```
SETDefault !<port> -PORT CONTrol = Disabled
SETDefault !<port> -PORT CONTrol = Enabled
```

Dial-On-Demand. To further reduce phone line costs when communicating over a WAN link in a Boundary Routing environment, you can configure the WAN link to be a dial-on-demand (DOD) line. When a demand occurs (user data needs to be transmitted), DOD automatically makes the call to establish the connection. The call is then terminated and reestablished automatically without any intervention depending upon whether or not there is data to be sent across the line. Connections that are no longer in use are temporarily terminated until a new demand occurs. For conceptual information on DOD, refer to “Dial-on-Demand” on page 37-4.

When routing IPX over a DOD line in a Boundary Routing environment, you can use the IPX spoofing feature in software version 8.0 and later to control the number of NetWare Communication Protocols (NCP) KeepAliveRequest packets (also known as WatchDog packets) from central node servers to the peripheral node clients. Spoofing helps manage the amount of traffic over the DOD line without violating the integrity of NCP connection maintenance. For conceptual information on spoofing, refer to “Summary of Bandwidth Manager Commands and Parameters” on page 37-28. For procedural steps, refer to “IPX with Incremental Broadcasts over a DOD Link” on page 37-26.

Data Compression. You can use data compression in all types of Boundary Routing topologies, but in particular, using the data compression feature in an SNA Boundary Routing topology causes SNA packets to be dramatically reduced in size. Data compression reduces the cost associated with the WAN lines by compressing the size of SNA packets, which increases the rate at which the now-smaller SNA packets traverse the line. Data compression causes the WAN line to be used more efficiently, that is, the faster SNA traffic traverses the WAN line, the more bandwidth is available to route or forward more SNA packets.

For more information on data compression, refer to Chapter 39.

Peer Data Exchange. You can configure specific clients or *peers* on leaf networks in an IBM Boundary Routing topology to exchange data. For example, in the NetBIOS topology shown in Figure 32-13, imagine that LAN requesters B and C need to exchange data with each other, but they do not need to exchange data with LAN requesters A and D.

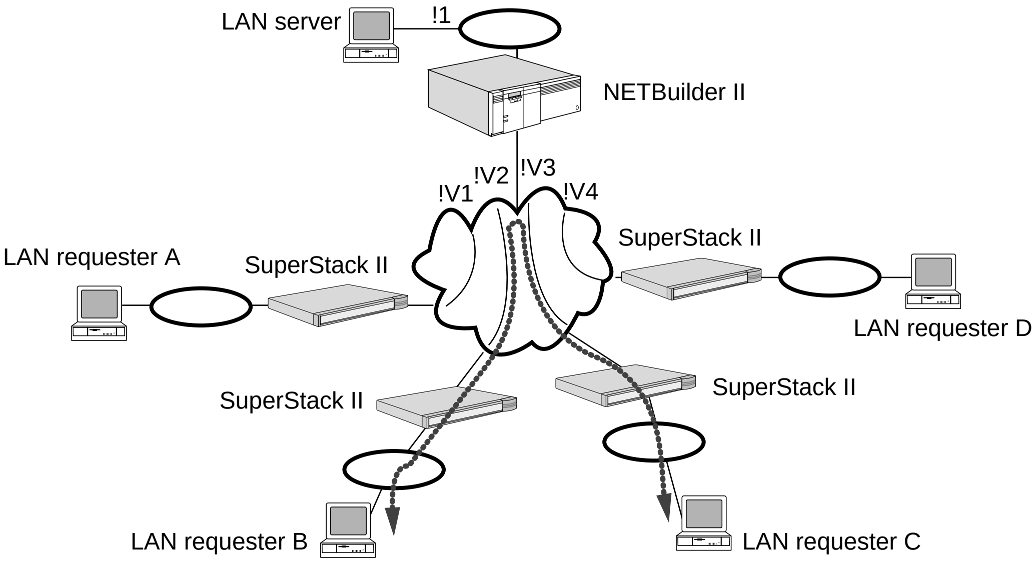


Figure 32-13 Peer Data Exchange In a NetBIOS Topology

You can configure LAN requesters B and C to exchange data by setting the value of the -LLC2 CONTrol parameter to Enable on virtual ports V2 and V3 of the central node. The -LLC2 CONTrol parameter usually is enabled on LAN ports only, for example, on port 1 in the NetBIOS topology. By enabling this parameter on virtual ports V2 and V3 in this topology, you are essentially making these virtual WAN ports operate as LAN ports.

Increased Reliability

The features discussed in the following sections increase the reliability associated with WAN usage.

Local Termination. In software versions 8.1 and earlier, LLC2 sessions initiated in an IBM Boundary Routing topology are considered *end-to-end.* End-to-end LLC2 sessions are those initiated at a terminal and run continuously from terminal, cluster controller, or LAN requester to the peripheral node, to the central node, and terminated at the front-end processor (FEP), mainframe, or LAN server. Figure 32-14 is an example of an end-to-end LLC2 session. The problem with this type of session is that many IBM applications running on an SNA or NetBIOS host are timing-sensitive. Delays or bottlenecks in the WAN can cause these applications to time out, which can cause users at terminals to lose data and to log on to the network again if the LLC2 session goes down.

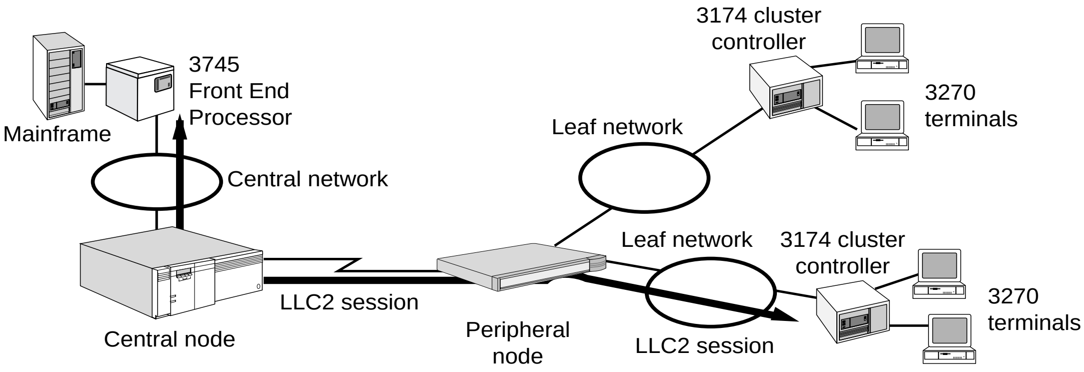


Figure 32-14 End-to-End LLC2 Session

In software versions 8.2 and later, an LLC2 session initiated in the same IBM Boundary Routing topology is considered *logical end-to-end*. A logical end-to-end LLC2 session is one that is terminated on the local port of each 3Com bridge/router and boundary router and then reinitiated at the wide area port. Figure 32-15 shows an example of a logical end-to-end LLC2 session. In this figure, an LLC2 session is initiated at a terminal on one of the leaf networks. The session is locally terminated at the peripheral node. The peripheral node then initiates another session, which is terminated at the central node. The central node initiates another session, which terminates at the FEP.

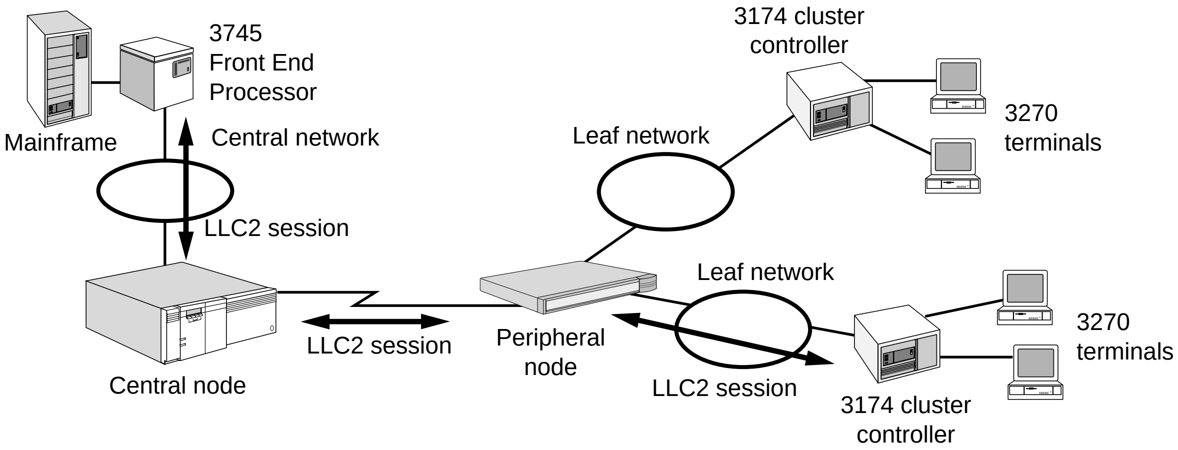


Figure 32-15 Logical End-to-End LLC2 Session

The ability of the 3Com bridge/routers and boundary routers to terminate an LLC2 session on a local port and initiate another LLC2 session on a wide area port or vice versa is called *local termination*. In addition to breaking up a continuous LLC2 session into multiple sessions, local termination switches packets from an SNA or NetBIOS environment to a Boundary Routing environment and reduces the propagation of SNA and NetBIOS broadcast packets on the WAN.

Although a logical end-to-end LLC2 session is broken down into multiple sessions, these still provide a continuous logical link from terminal or workstation to host and vice versa. In fact, breaking a continuous end-to-end LLC2 session into multiple sessions eliminates delays or bottlenecks thereby making the session more reliable.

To activate local termination, set the value of the -BCN CONTrol parameter to IbmTraffic.

Automatic Prioritization of IBM Traffic. Because of the interactive way in which clients and their SNA or NetBIOS hosts interoperate, IBM traffic has the following characteristics:

- It tends to be mission critical.
- It tends to be bursty.

To ensure the access of accurate information in the shortest amount of time possible, SNA and NetBIOS traffic that is sent through a port configured for Boundary Routing has been automatically prioritized as high and medium, respectively. No configuration is necessary.

Non-IBM protocols, such as IP, IPX, and AppleTalk traffic are also automatically prioritized as medium.

For example, if SNA, NetBIOS, and IP traffic must traverse wide area links that have been configured for Boundary Routing, automatic prioritization allows SNA traffic to travel across wide area links first, then NetBIOS or IP traffic, depending on which type of traffic is first in the queue.

Automatic prioritization of IBM traffic is a separate and distinct feature from the prioritization that the APPN class of service feature provides.

Continuous Operation

Boundary Routing software provides continuous operation with the dial-up backup line for disaster recovery or bandwidth-on-demand, with the assignment of network numbers, and with a mechanism for constructing resilient networks.

Dial-up Backup Line for Disaster Recovery or Bandwidth-on-Demand.

You can use dial-up paths to take advantage of disaster recovery or bandwidth-on-demand features in non-IBM and IBM Boundary Routing topologies. The dial-up paths must belong to the same port, must be connected to the same end-points, and must be running PPP as the data link protocol as shown in Figure 32-16. Although this figure shows Ethernet as the LAN media type, token ring can also be used.

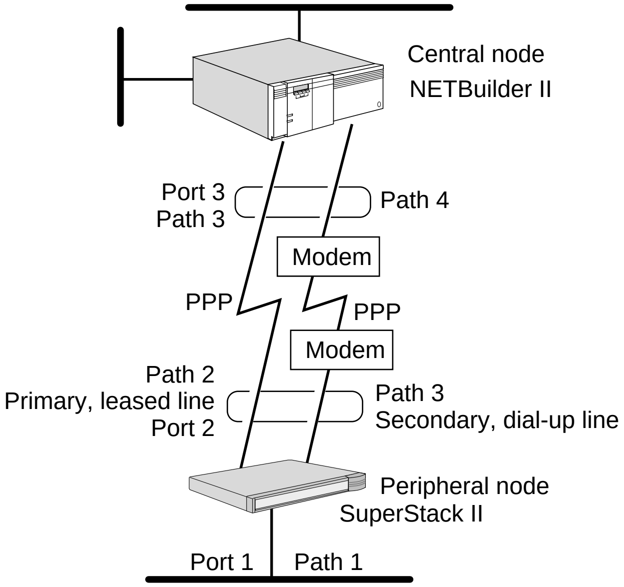


Figure 32-16 Boundary Routing Backup Line for Disaster Recovery or Bandwidth-on-Demand

In software version 8.0 and later, the secondary path to be used for disaster recovery or bandwidth-on-demand can be selected from the dynamic dial path pool.

Beginning with software version 9.1, lines are monitored by *bandwidth management*, which applies static bandwidth, dynamic bandwidth, or a combination of these, to provide a port with the bandwidth it needs to meet current requirements. A line failure that drops the port's bandwidth below a specified level causes bandwidth management to restore the specified bandwidth. If traffic conditions warrant additional bandwidth, the bandwidth-on-demand function also automatically increases the bandwidth accordingly. You can configure a line specifically for disaster recovery or as a general purpose (unrestricted) line that can be allocated for disaster recovery.

At the peripheral node, you need to assign two paths to one port, configure the path attributes for the lines, and enable disaster recovery and bandwidth-on-demand.

At the central node, you need to assign two paths to one port (or use the dynamic dial path pool), configure the path attributes for the lines, and enable disaster recovery or bandwidth-on-demand.

Enabling disaster recovery or bandwidth-on-demand allows bandwidth management to switch traffic to another path or allocate additional resources (disaster recovery) or allocate additional path resources if the traffic threshold on the path is exceeded (bandwidth-on-demand).

For information on configuring modems, refer to the *WAN Cabling and Connectivity Guide*. You can find this guide on the 3Com World Wide Web site by entering:

`http://www.3com.com/`

To configure backup dial-up lines for disaster recovery or bandwidth-on-demand on the central and peripheral node, refer to Chapter 37.

Assigning Network Numbers. Assigning network numbers for routing protocols such as IP, IPX, and AppleTalk in a Boundary Routing topology differs from the same task in a non-Boundary Routing topology. For example, in the non-Boundary Routing topology using IP routing shown in Figure 32-17, an IP network number (IP address) is assigned to each LAN port and to each WAN port that is directly attached to the Frame Relay network.

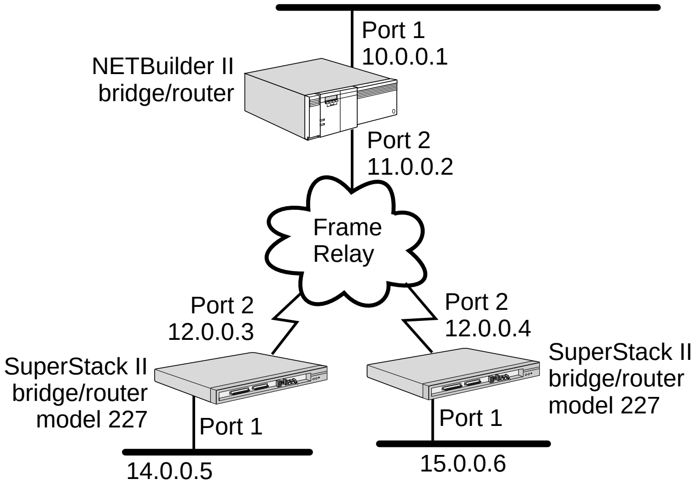


Figure 32-17 Assigning Network Numbers in a Non-Boundary Routing Topology

In a Boundary Routing topology, you assign network numbers to the following ports:

- The LAN port on the central node
- The virtual port on the central node for each remote site

The network number assigned to the virtual port is also used for the remote LAN. (A virtual cable connects the central node to the LAN connector on the peripheral node.) For information on administering IP addresses for peripheral nodes using either a Reverse Address Resolution Protocol (RARP) or BOOTP server, refer to "Configuring for PPP" on page 32-1,"Configuring for Frame Relay" on page 32-6, or "Configuring for X.25" on page 32-11.

For example, in the Boundary Routing topology shown in Figure 32-18, network numbers for IP, IPX, and AppleTalk routing are assigned to the LAN port of the central node (port 1) and to each remote LAN through the use of virtual ports (virtual ports V2 and V3). The dashed lines in Figure 32-18 indicate the virtual connection between the central and peripheral nodes.

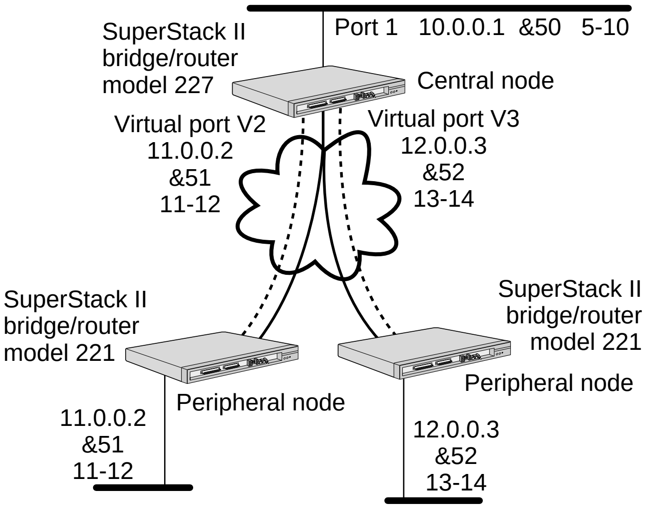


Figure 32-18 Assigning Network Numbers in a Boundary Routing Topology

Table 32-2 lists the ports and virtual ports in Figure 32-18 and the network numbers assigned to them to help you understand how network numbers are assigned specifically for the IP, IPX, and AppleTalk protocols.

Table 32-2 IP, IPX, and AppleTalk Network Numbers For Central Node in Boundary Routing Topology

Port Number (As Shown in Figure 32-18)	IP Network Number (Address)	IPX Network Number	AppleTalk Network Number (Range)
Port 1	10.0.0.1	&50	5 – 10
Virtual port V2	11.0.0.2	&51	11 – 12
Virtual port V3	12.0.0.3	&52	13 – 14

Dual PVCs for IBM Traffic

Dual PVCs can divide IBM and non-IBM traffic over a Frame Relay data link in a Boundary Routing environment. IBM traffic at a leaf node is directed to its own PVC and transmitted to a central site using Boundary Routing. Dual PVCs enhance response time and bandwidth available for IBM traffic and allow network managers to monitor the IBM data link separately.

Only virtual ports are used in the Boundary Routing Frame Relay environment. The DLCI numbers used for the PVCs are specified when the virtual ports are defined. You must define the DLCI pairs using the -BCN LclNonIbmDlci parameter. Specify the DLCI that will be used for IBM traffic at the leaf node using the -BCN RemNonIbmDlci parameter. Use the -PORT CONTrol parameter to enable the port, which transmits the PVC configuration information from the central node to the leaf node.

To separate IBM traffic from non-IBM traffic, the software uses SAP numbers to filter for IBM frames. The software assumes frames whose SAP numbers fall between 0 and 0xF0 and that are divisible by 4 are IBM frames. The exceptions to this assumption are VINES IP SAP frame number 0xBC, IPX SAP frame number 0xE0, and Sync Research special SAP frame number 0xFC.

Network Resiliency

Through hardware and software configuration, you can design a Boundary Routing topology that has a backup or redundant link between a central and peripheral node or a backup or redundant route to an alternate central node. See Figure 32-3 on page 32-23 for an illustration of these two network resiliency configurations.

If your Boundary Routing topology has a redundant link between the central and peripheral nodes and the link fails, the central node will send and receive packets from the peripheral node through the redundant link. If your Boundary Routing topology has a redundant route to an alternate central node and the primary route or primary central node fails, the alternate central node will send and receive packets from the peripheral node through the redundant route.

The peripheral node allows only one active link between a central and peripheral node at a time. In a Boundary Routing topology with a redundant link, the primary link is considered the preferred link. In the topology with the redundant route to an alternate central node, the primary route is considered the preferred route.

When you turn the power on, the auto startup feature brings up one active path only. It does not bring up a second path for network resiliency. You need to configure the second path for network resiliency using the procedure in "Configuring Network Resiliency" on page 32-23.

The peripheral node software uses a set of precedence rules to determine which link or route should be treated as "preferred" when more than one link or route is available to be activated. These precedence rules are as follows:

- Leased-line connections take precedence over dial-up connections.
- Switched service (Frame Relay, X.25) takes precedence over PPP when they coexist. Within the switched services, Frame Relay takes precedence over X.25. PPP is likely to be used over a dial-up connection.

In operation, the precedence rules work as follows:

- When there is no currently active port, a port is allowed to come up.
- If the currently active port is a dial-up port and the new port is a leased-line port, then the leased-line port is allowed to come up and the dial-up port is deactivated.
- If the new port owner is a switched service, for example, Frame Relay, and the currently active port is point-to-point, then the Frame Relay port is allowed to come up and the PPP port is deactivated.
- In all other cases, the currently active port is left activated, and the new port is not allowed to come up.

In a Boundary Routing topology with a redundant route to an alternate central node, you must configure the alternate central node with the same address information as the primary central node. For more information, refer to "Primary and Alternate Central Node Configuration" on page 32-58.

If you plan to bridge or route AppleTalk, IP, or IPX, you must enable a central MAC address. You may also need to configure the alternate central node with the same bridging or routing attributes as the primary central node. For more information, refer to "Using the Central MAC Address" on page 32-59.

Network Resiliency Using a Redundant Link

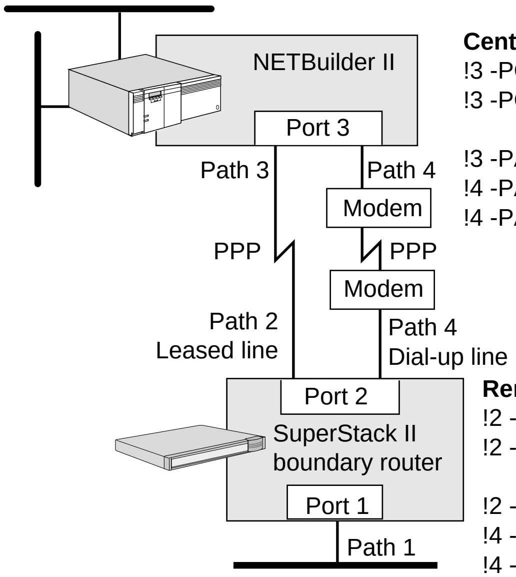
You can configure a redundant link between a central and peripheral node in PPP and Frame Relay environments. You can also configure a redundant link in an IBM Boundary Routing topology that uses PPP or Frame Relay.

PPP Environment. A primary link using a PPP leased line and a redundant link using a PPP dial-up line provides network resiliency in the event that the primary link fails at either the central or peripheral nodes.

To achieve this network resiliency in this configuration, two paths are mapped to a single logical port on both ends of the WAN link. For example, the devices can exchange data over a primary link with a secondary dial-up link for either disaster recovery or bandwidth-on-demand. Disaster recovery activates an additional dial-up line if the primary lines fails. Bandwidth-on-demand (through bandwidth management) activates additional resources in cases where the line experiences congestion. Achieving link redundancy in this way is supported only for PPP-based Boundary Routing environments (dial-up or leased lines), because a Frame Relay or X.25 port does not support multiple physical paths.

In the configuration discussed in the preceding paragraph, you can use either a DTE or ISDN line as the secondary dial-up link for bandwidth-on-demand and disaster recovery. For more information on ISDN, refer to Chapter 35.

The two links between the two nodes maintain the single connection to the central node, which is key to the Boundary Routing system architecture because logical data flows over a single WAN port. Figure 32-19 is an example of the PORT and PATH Service parameter values applicable to this configuration.



Central Site Parameters

```
!3 -PORT OWNer = PPP
!3 -PORT DialCONTrol = DisasterRcvry

!3 -PATH LineType = Leased
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol =  (UnReSTricted, Answer, Originate)
```

Remote Site Parameters

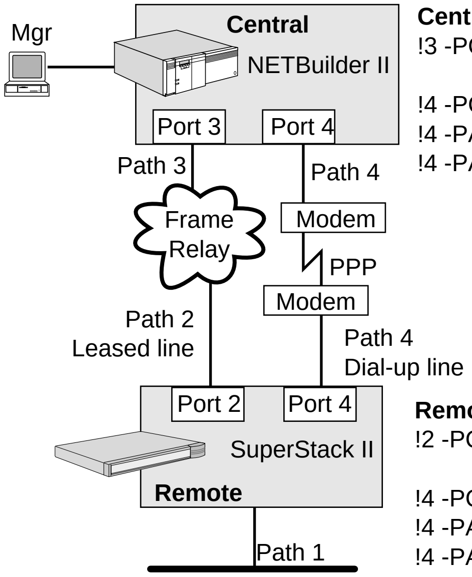
```
!2 -PORT OWNer = PPP
!2 -PORT DialCONTrol = DisasterRcvry

!2 -PATH LineType = Leased
!4 -PATH DialCONTrol = (UnReSTricted, Answer, Originate)
!4 -PATH LineType = Dialup
```

Figure 32-19 Network Resiliency PORT and PATH Parameters in a PPP Environment

Frame Relay Environment. A primary link using a Frame Relay leased line and a redundant link using a PPP dial-up line provides network resiliency if the primary link fails at either the central or peripheral nodes.

In the configuration discussed in the preceding paragraph, a data terminal equipment (DTE) or Integrated Services Digital Network (46) line can be used as the backup dial-up line for network resiliency. For more information on ISDN, refer to Chapter 35.

Figure 32-20 is an example of the PORT and PATH Service parameters that support Frame Relay as the primary link and PPP as the redundant (dial-up) link.



Central Site Parameters

```
!3 -PORT OWNer = FrameRelay

!4 -PORT OWNer = PPP
!4 -PATH DialCONTrol =  (UnReSTricted, Originate)
!4 -PATH LineType = Dialup
```

Remote Site Parameters

```
!2 -PORT OWNer = FrameRelay

!4 -PORT OWNer = PPP
!4 -PATH DialCONTrol =  (UnReSTricted, Answer, NoOriginate)
!4 -PATH LineType = Dialup
```

Figure 32-20 Network Resiliency PORT and PATH Parameters in a Frame Relay Environment

This configuration differs from the configuration discussed in "PPP Environment" on page 32-48 because instead of allowing multiple paths per port as PPP does, Frame Relay requires that a single path and single port mapping must be maintained. Two logical port destinations are possible on the central site router instead of just one as in the PPP configuration.

In this configuration, network resiliency is achieved with additional user or SNMP intervention. Each interface to which the central node is connected must be configured with the identical network address information. Because duplicate network addresses are not allowed on the same NETBuilder II bridge/router or model 227, 327, 427, or 527 SuperStack II bridge/router, macros can be predefined to delete the network address on the Frame Relay port and add that same address to the PPP port as required. When the loss of connection to the remote site is detected, the macro executes to properly address the backup port. A similar macro may be created to reverse this process to change back to the primary port when it recovers.

Macros can be executed manually, or a central site management station can automate the process by monitoring the status of the remote site. When the user or the management station detects that the remote site is no longer reachable, the user or the management station may run a script file that Telnets to the central node and executes the macro. This operation will cause a session disruption.

The central site macro that activates the redundant link must follow these steps:

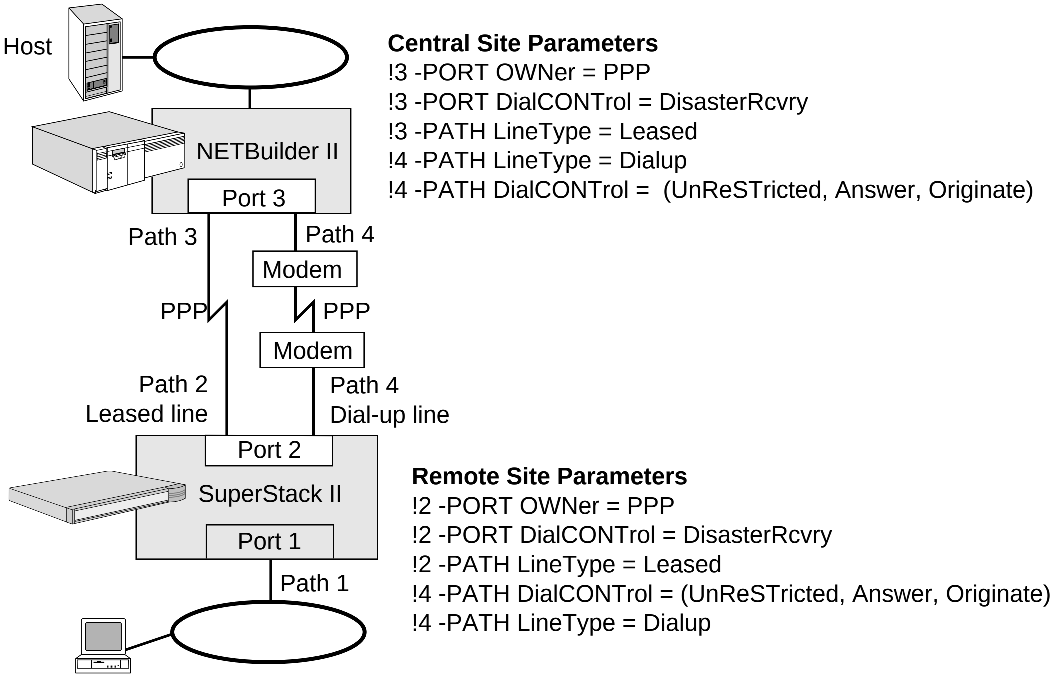
- Disable the primary port.
- Delete addresses from the primary port.
- Add addresses to the backup port.
- Enable the backup port.

The central site macro that reactivates the primary link must follow these steps:

- Disable the backup port.
- Delete addresses from the backup port.
- Add addresses to the primary port.
- Enable the primary port.

IBM Environment. You can configure a redundant link in an IBM Boundary Routing topology that uses PPP or Frame Relay.

Figure 32-21 shows an IBM Boundary Routing topology that has a primary link using a PPP leased line and a redundant link using a PPP dial-up line.



Central Site Parameters
!3 -PORT OWNer = PPP
!3 -PORT DialCONTrol = DisasterRcvry
!3 -PATH LineType = Leased
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol = (UnReSTricted, Answer, Originate)

Remote Site Parameters
!2 -PORT OWNer = PPP
!2 -PORT DialCONTrol = DisasterRcvry
!2 -PATH LineType = Leased
!4 -PATH DialCONTrol = (UnReSTricted, Answer, Originate)
!4 -PATH LineType = Dialup

Figure 32-21 Network Resiliency PORT and PATH Parameters in IBM Environment Using PPP

To achieve network resiliency in this configuration, two paths are mapped to a single logical port on both ends of the WAN link. For example, the devices can exchange data over a primary link with a secondary dial-up link for both disaster recovery and bandwidth-on-demand. Disaster recovery activates the secondary, dial-up line if the primary lines fails. Bandwidth-on-demand activates the secondary line in cases where the primary line experiences congestion. Achieving link redundancy in this way is supported only for PPP-based Boundary Routing environments (dial-up or leased lines), because a Frame Relay or X.25 port does not support multiple physical paths.

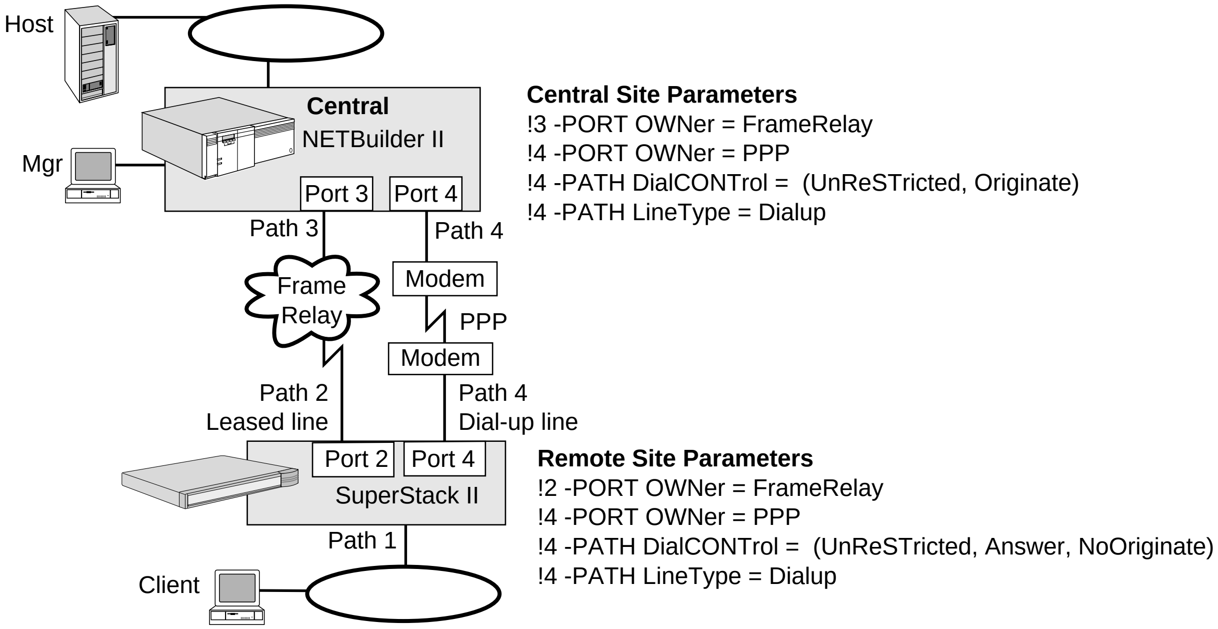
In the IBM Boundary Routing topology with a primary PPP leased line and the secondary PPP dial-up line, you can use a DTE or ISDN line as the secondary dial-up link for disaster recovery and bandwidth-on-demand. For more information on ISDN, refer to Chapter 35.

A problem experienced when performing Boundary Routing in a connection-oriented environment such as SNA and NetBIOS is that when a currently active port deactivates and a new port activates, the session between the central and peripheral nodes is disrupted. In the IBM Boundary Routing topology with a primary PPP leased line and the secondary PPP dial-up line, you will not experience disruption for two reasons:

- This topology requires that two paths are mapped to one port. The local termination feature, which is activated when Boundary Routing over IBM is enabled, isolates the mainframe or LAN server and terminal or LAN requester from disruptions between the central and peripheral nodes.
- Increasing the retry counter on the Boundary Routing port of the central node decreases the possibility that the circuit will be brought down while the current active port deactivates and the new port activates.

Having two links between the two nodes maintains the single connection to the central node that is key to the Boundary Routing system architecture, because logical data flows over a single WAN port.

Figure 32-22 shows an IBM Boundary Routing topology that has a primary link using a Frame Relay leased line and a redundant link using a PPP dial-up line. You can use a DTE or ISDN line as the backup dial-up line for network resiliency. For more information on ISDN, refer to Chapter 35.



Central Site Parameters
!3 -PORT OWNer = FrameRelay
!4 -PORT OWNer = PPP
!4 -PATH DialCONTrol = (UnReSTricted, Originate)
!4 -PATH LineType = Dialup

Remote Site Parameters
!2 -PORT OWNer = FrameRelay
!4 -PORT OWNer = PPP
!4 -PATH DialCONTrol = (UnReSTricted, Answer, NoOriginate)
!4 -PATH LineType = Dialup

Figure 32-22 Network Resiliency PORT and PATH Parameters in IBM Environment Using Frame Relay

As with the IBM Boundary Routing topology using PPP discussed earlier in this section, the LLC2 session between client and host is disrupted if the primary link fails and the redundant link activates in IBM Boundary Routing topology using Frame Relay.

In the IBM Boundary Routing topology using Frame Relay, network resiliency is achieved with additional user or SNMP intervention. Each interface to which the central node is connected must be configured with the identical network address information. Because duplicate network addresses are not allowed on the same NETBuilder II bridge/router or model 227, 327, 427, or 527 SuperStack II bridge/router macros can be predefined to delete the network address on the Frame Relay port and add that same address to the PPP port as required. When the loss of connection to the remote site is detected, the macro executes to properly address the backup port. A similar macro may be created to reverse this process in order to change back to the primary port when it recovers.

Macros can be executed manually, or a central site management station can automate the process by monitoring the status of the remote site. When the user or the management station detects that the remote site is no longer reachable, the user or the management station may run a script file that Telnets to the central node and executes the macro. This operation will cause a session disruption.

The central site macro that activates the redundant link must follow these steps:

- Disable the primary port.
- Delete addresses from the primary port.
- Add addresses to the backup port.
- Enable the backup port.

The central site macro that reactivates the primary link must follow these steps:

- Disable the backup port.
- Delete addresses from the backup port.
- Add addresses to the primary port.
- Enable the primary port.

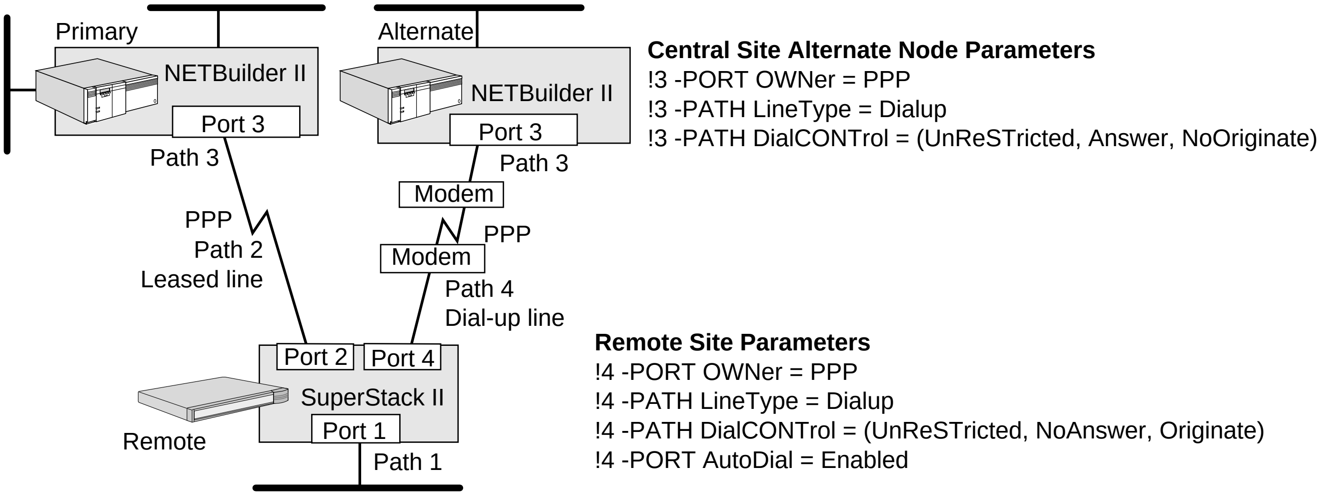
Network Resiliency Using a Redundant Route to an Alternate Central Node

You can configure a redundant route to an alternate central node in PPP, Frame Relay, and X.25 environments. You can also configure a redundant route to an alternate central node in an IBM Boundary Routing that uses PPP, Frame Relay, or X.25.

The precedence rules discussed in "Network Resiliency" on page 32-47 apply at the port level, so when two central nodes are used, a single path and single port mapping must be maintained on both central and peripheral nodes, regardless of the media combination. When using two central nodes, network resiliency does not work properly with multiple paths assigned to a single port.

PPP Environment. A primary route using a PPP leased line and a redundant route using a PPP dial-up line provides network resiliency if the primary route fails at either the primary central or peripheral nodes or if the primary central node fails. You can use a DTE or ISDN line as the backup dial-up line for network resiliency. For more information on ISDN, refer to Chapter 35.

Figure 32-23 shows the PORT and PATH Service parameters required by this configuration.



Central Site Alternate Node Parameters

```
!3 -PORT OWNer = PPP
!3 -PATH LineType = Dialup
!3 -PATH DialCONTrol = (UnReSTricted, Answer, NoOriginate)
```

Remote Site Parameters

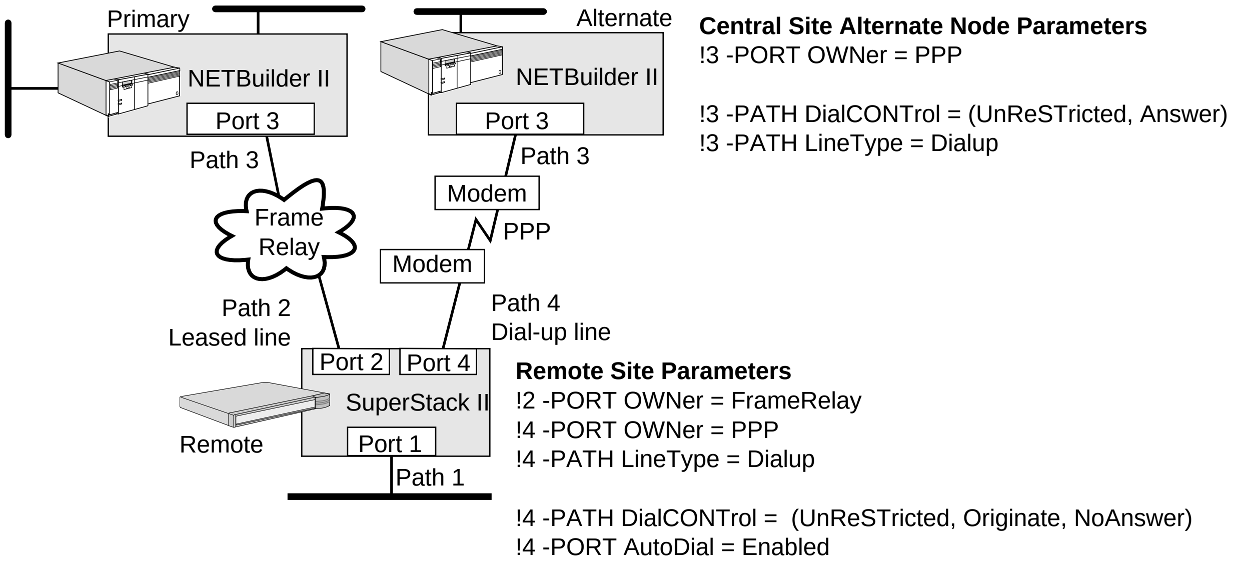
```
!4 -PORT OWNer = PPP
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol = (UnReSTricted, NoAnswer, Originate)
!4 -PORT AutoDial = Enabled
```

Figure 32-23 Network Resiliency PORT and PATH Parameters for Two Central Site Nodes in a PPP Environment

The precedence rules ensure that a leased line is always active when available. In the case of a failure, the dial-up line is activated and the peripheral node attempts to connect to the alternate central site. When the leased line recovers, it again takes precedence over the dial-up line. The peripheral node automatically hangs up the connection and changes back to the leased line.

The precedence rules applied at the remote site determine which connection to keep active. In this case, the connection attempts are fully automated and generated from the peripheral node instead of from the central site.

Frame Relay Environment. A primary route using a Frame Relay leased line and a redundant route using a PPP dial-up line provides network resiliency if the primary route fails at either the primary central or peripheral nodes or if the primary central node fails. You can use a DTE or ISDN line as the backup dial-up line for network resiliency. For more information on ISDN, refer to Chapter 35. Figure 32-24 shows the PORT and PATH Service parameters required by this configuration.



Central Site Alternate Node Parameters
!3 -PORT OWNer = PPP

!3 -PATH DialCONTrol = (UnReSTricted, Answer)
!3 -PATH LineType = Dialup

Remote Site Parameters
!2 -PORT OWNer = FrameRelay
!4 -PORT OWNer = PPP
!4 -PATH LineType = Dialup

!4 -PATH DialCONTrol = (UnReSTricted, Originate, NoAnswer)
!4 -PORT AutoDial = Enabled

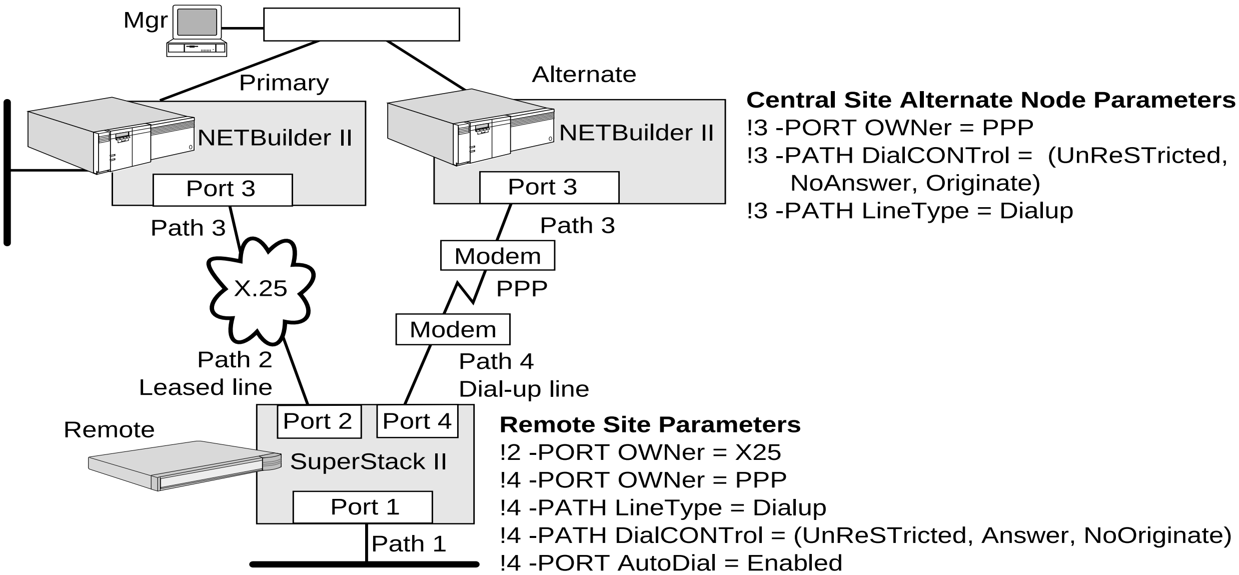
Figure 32-24 Network Resiliency PORT and PATH Parameters for Two Central Site Nodes in a Frame Relay Environment

The precedence rules ensure that the Frame Relay line is always active when available. In the case of a failure, the PPP line is activated and the peripheral node dials the alternate central site node. When the Frame Relay line recovers, it again takes precedence over the PPP line and the peripheral node automatically hangs up the connection and changes back to the Frame Relay link.

In case of a primary route or primary central node failure, the Local Management Interface (LMI) Protocol will no longer report the primary central node data link connection identifier (DLCI) to the peripheral node. This triggers a path-down state at the peripheral node at which time the PPP dial backup attempts to dial the alternate central node. The precedence rules cause this operation to be controlled at the peripheral node.

X.25 Environment. A primary route using an X.25 leased line and a redundant route using a PPP dial-up line provides network resiliency in the event that the primary route fails at either the primary central or peripheral nodes or if the primary central node fails. You can use a DTE or ISDN line as the backup dial-up line for network resiliency. For more information on ISDN, refer to Chapter 35.

Figure 32-25 shows the PORT and PATH Service parameters required by this configuration.



Central Site Alternate Node Parameters

```
!3 -PORT OWNer = PPP
!3 -PATH DialCONTrol = (UnReSTricted,
     NoAnswer, Originate)
!3 -PATH LineType = Dialup
```

Remote Site Parameters

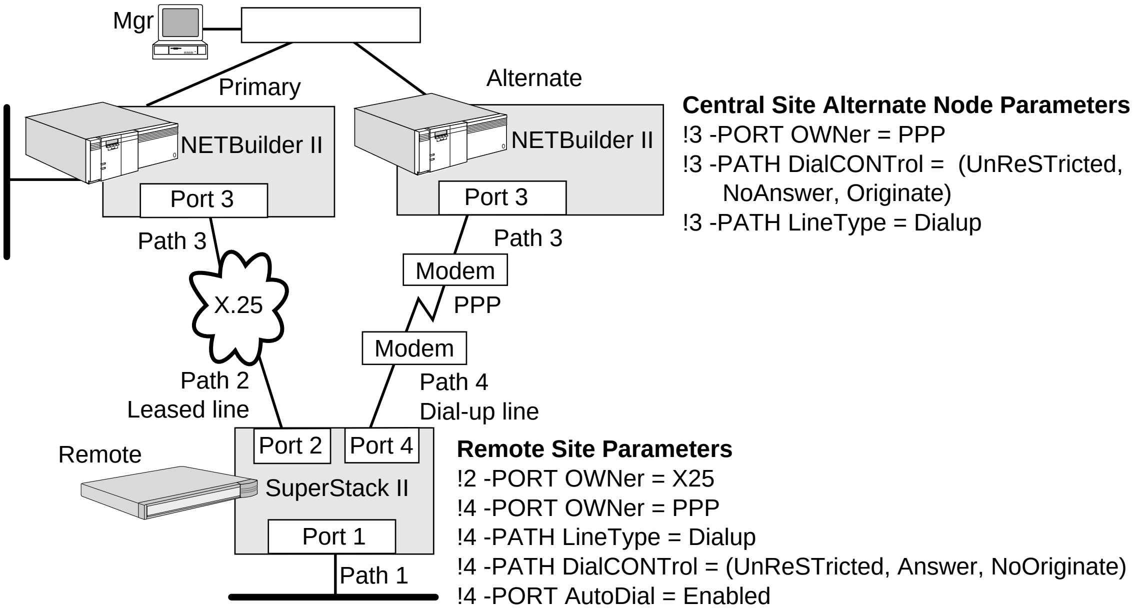
```
!2 -PORT OWNer = X25
!4 -PORT OWNer = PPP
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol = (UnReSTricted, Answer, NoOriginate)
!4 -PORT AutoDial = Enabled
```

Figure 32-25 Network Resiliency PORT and PATH Parameters for Two Central Site Nodes in an X.25 Environment

Network resiliency in an X.25 environment requires manual intervention unless triggered by a management station. X.25 supports switched virtual circuits (SVCs), and circuits are established and torn down as required by traffic flow through the central node. The loss of an X.25 virtual circuit does not cause a path and port down state to occur at the central node. If the peripheral node loses its link to the X.25 network, the virtual port of the central node remains active. If the peripheral node is allowed to automatically dial an alternate central node, the connection is accepted, but the primary central node continues to assert its network layer information for the virtual port onto surrounding networks through routing updates. When the alternate central node accepts the incoming call, it also begins to assert the same routing information onto the network. Two routes are advertised to get to the remote LAN, but only one route is valid.

To avoid this situation, the virtual port on the primary central node must be disabled before the port on the alternate central node is allowed to accept the call. You can disable the virtual port using an SNMP management station. When the management station detects that it can no longer reach the peripheral node, it can execute a script that Telnets to the primary central node and disables the virtual port, then Telnets to the alternate central node and executes the connection attempt to the peripheral node. Special configuration parameters are required to ensure that a call cannot be established until this occurs.

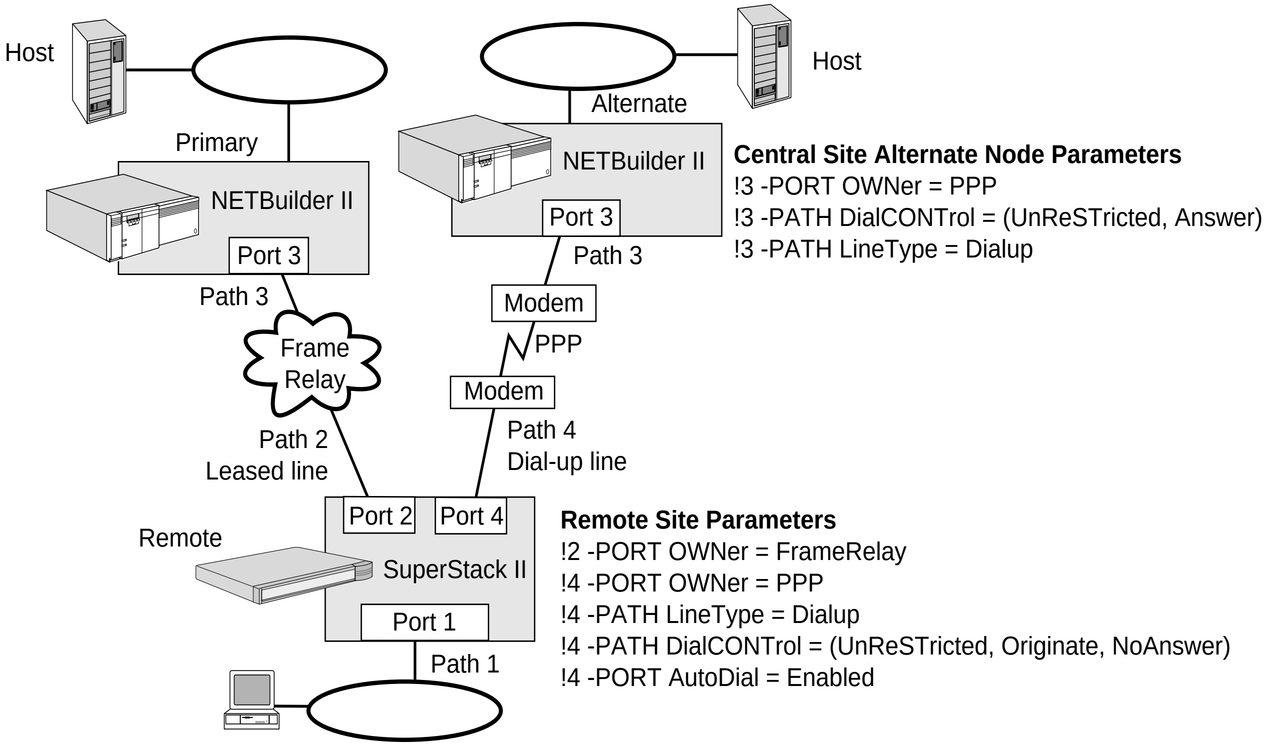
IBM Environment. You can configure a redundant route to an alternate central node in an IBM Boundary Routing topology that uses PPP, Frame Relay, or X.25. Figure 32-26, Figure 32-27, and Figure 32-28 show IBM Boundary Routing topologies that have primary routes using PPP, Frame Relay, and X.25 leased lines, respectively, and redundant routes using PPP dial-up lines.



Central Site Alternate Node Parameters
!3 -PORT OWNer = PPP
!3 -PATH DialCONTrol = (UnReSTricted,
NoAnswer, Originate)
!3 -PATH LineType = Dialup

Remote Site Parameters
!2 -PORT OWNer = X25
!4 -PORT OWNer = PPP
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol = (UnReSTricted, Answer, NoOriginate)
!4 -PORT AutoDial = Enabled

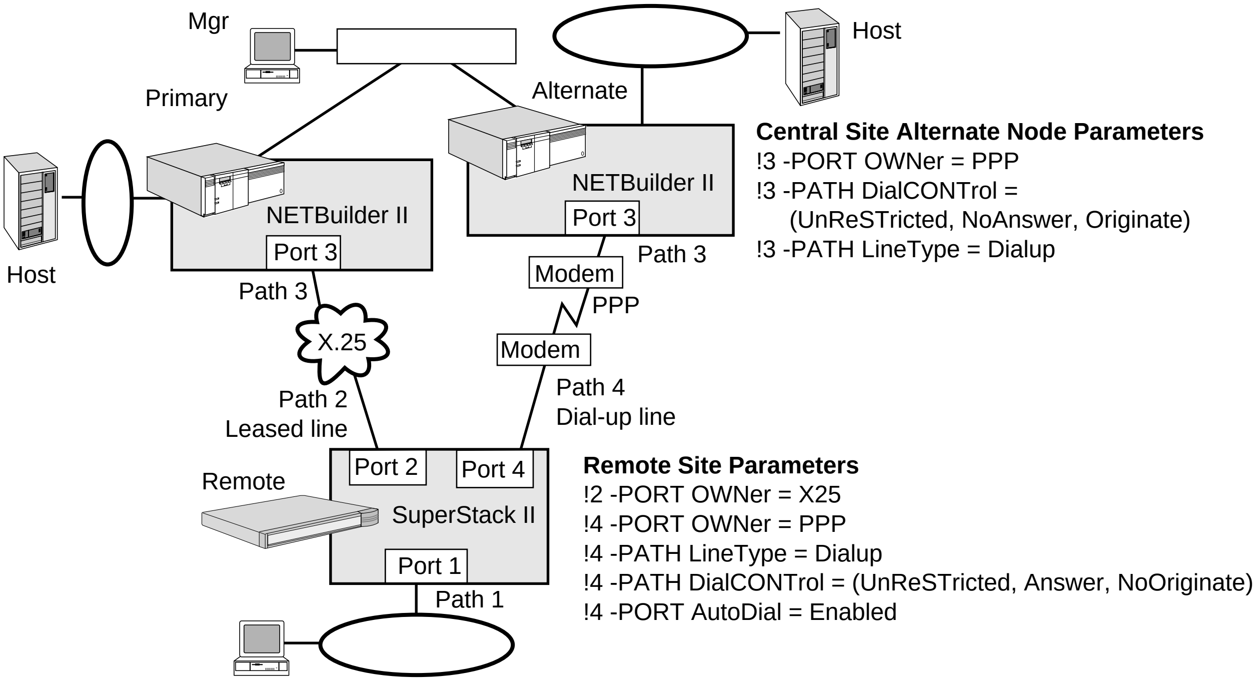
Figure 32-26 Network Resiliency PORT and PATH Parameters for Two Central Nodes in IBM Topology Using PPP



Central Site Alternate Node Parameters
!3 -PORT OWNer = PPP
!3 -PATH DialCONTrol = (UnReSTricted, Answer)
!3 -PATH LineType = Dialup

Remote Site Parameters
!2 -PORT OWNer = FrameRelay
!4 -PORT OWNer = PPP
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol = (UnReSTricted, Originate, NoAnswer)
!4 -PORT AutoDial = Enabled

Figure 32-27 Network Resiliency PORT and PATH Parameters for Two Central Nodes in an IBM Topology Using Frame Relay



Central Site Alternate Node Parameters
!3 -PORT OWNer = PPP
!3 -PATH DialCONTrol =
(UnReSTricted, NoAnswer, Originate)
!3 -PATH LineType = Dialup

Remote Site Parameters
!2 -PORT OWNer = X25
!4 -PORT OWNer = PPP
!4 -PATH LineType = Dialup
!4 -PATH DialCONTrol = (UnReSTricted, Answer, NoOriginate)
!4 -PORT AutoDial = Enabled

Figure 32-28 Network Resiliency PORT and PATH Parameters for Two Central Nodes in an IBM Topology Using X.25

In the topologies shown in each of these figures, you can use a DTE or ISDN line as the backup dial-up line used as the redundant route. For more information on ISDN, refer to Chapter 35.

In the topologies that have primary routes using PPP and Frame Relay leased lines, the precedence rules ensure that the PPP and Frame Relay leased lines are always active when available. In the case of a failure, the PPP dial-up line is activated and the peripheral node dials the alternate central site node. When the PPP or Frame Relay leased line recovers, it again takes precedence over the PPP dial-up line and the peripheral node automatically hangs up the connection and changes back to the PPP or Frame Relay leased line.

In the topology that has a primary route using PPP, the precedence rules applied at the remote site determine which connection to keep active. In this case, the connection attempts are fully automated and generated from the peripheral node instead of from the central site.

In the topology that has a primary route using Frame Relay, if a primary route or the primary central node fail, the LMI Protocol will no longer report the primary central node DLCI to the peripheral node. This triggers a path-down state at the peripheral node at which time the PPP dial backup attempts to dial the alternate central node. The precedence rules cause this operation to be controlled at the peripheral node.

In the topology that has a primary route using X.25, if the primary route or the primary central node fail, manual intervention is required unless you have previously generated scripts on your SNMP management station that automates certain tasks.

X.25 supports switched virtual circuits (SVCs), and circuits are established and torn down as required by traffic flow through the central node. The loss of an X.25 virtual circuit at the peripheral node does not cause a path and port down state to occur at the central node. If the peripheral node loses its link to the X.25 network, the central node virtual port remains active. If the peripheral node is allowed to automatically dial an alternate central node, the connection is accepted, but the primary central node continues to assert its network layer information for the virtual port onto surrounding networks through routing updates. When the alternate central node accepts the incoming call, it also begins to assert the same routing information onto the network. Two routes are advertised to get to the remote LAN, but only one route is valid.

To avoid this situation, the virtual port on the primary central node must be disabled before the port on the alternate central node is allowed to accept the call. This can be done through an SNMP management station. When the management station detects that it can no longer reach the peripheral node, it can execute a script that Telnets to the primary central node and disables the virtual port, then Telnets to the alternate central node and executes the connection attempt to the peripheral node. Special configuration parameters are required to ensure that a call cannot be established until this occurs.

Because the configuration of duplicate MAC addresses is not used in an IBM Boundary Routing environment, if you use one of the topologies discussed in the preceding paragraphs, you will experience a disruption if one port deactivates and another activates. When the session has been disrupted, you will need to log in again and reinitiate a session.

In each of the topologies, imagine that the client has initiated an LLC2 session with the host on the primary central network. Since the primary line is up, the session takes place over this line. If the primary line goes down during the session between client and host, from the user's perspective, the session abruptly terminates or is disconnected. Eventually, the secondary line comes up. If you attempt to log in and reinitiate a session with the host before the secondary line comes up, you will be unsuccessful; if the attempt is made after the secondary line comes up, the attempt will be successful. When the primary line has been repaired and comes up again, the disruption will occur again.

Primary and Alternate Central Node Configuration. End stations on the remote LAN use the logical address of the primary central node WAN port as the next hop when routing data. To provide a transition to the alternate central node, WAN ports on both routers connected to a remote site in non-IBM and IBM Boundary Routing topologies must share the same address information. Because the Boundary Routing system architecture does not allow both connections to be active at the same time, it is possible to configure identical logical addresses (IP addresses, IPX network number, and so forth) on both routers. The address duplication does not interfere with network operation as long as the connections are not simultaneously active.

Some non-IBM protocols, such as IPX, present more of a challenge because they adopt the underlying MAC addresses for use as a logical host address. Duplicated network numbers are not sufficient in this situation; you must also configure the central node to use the same MAC address on those WAN interfaces. For information on configuring both primary and alternate central nodes to use the same MAC address on each WAN port, refer to "Using the Central MAC Address" on page 32-59.

Using the Central MAC Address

If you configure a redundant route to an alternate central node, you may need to configure both primary and alternate central nodes to use a central MAC address, which is a special, internally saved MAC address. This MAC address allows certain protocols to switch to the alternate central node without losing sessions between a client on the leaf network and a host on the central site network.

If you are configuring disaster recovery over Frame Relay on the peripheral node of a Boundary Routing environment, do not configure the central MAC address.

Use the central MAC address with the following protocols:

- IPX
- AppleTalk
- IP
- Bridging

Setting the -BCN CONTrol parameter to CentralMac on the Boundary Routing ports of both the primary and alternate central nodes causes both nodes to use the same MAC address. The transition to an alternate central node, if necessary, is completely transparent to the end stations.

The central MAC address is not used in an IBM Boundary Routing environment.

IPX Routing Example. When routing IPX in a Boundary Routing environment as shown in Figure 32-29, you need to configure the alternate central node #2 with the same routing attributes and network addresses as central node #1. You must also enable IPX routing on both central nodes. If you enable the use of the central MAC address on both central nodes, the switch from the primary to alternate central node is transparent to the user.

For example, if a user on the leaf network has a session established with a server on the central site network when a failure occurs (link #1 or central node #1 fails), the alternate central node brings up link #2, and the session between the client and the server is not disrupted. The client continues the session with the server although the route is established on link #2 and through the alternate central node.

The use of the central MAC address provides a transparent switch to the alternate central node because NetWare clients cache next-hop router information, including the router's MAC address and network number. In addition, clients do not listen to RIP updates to validate router addresses or detect routers that have gone down. By configuring the same network addresses and enabling the central MAC address on both central nodes, if link #1 or central node #1 fails when a session is established between a client on the leaf network and a server on the central site network, the session is not interrupted when the alternate central node becomes active. The client continues to use the MAC address and network address that is stored in its cache, although the client is accessing the server through link #2 and alternate central node #2.

If you do not enable the use of the central MAC address on both of the central nodes, the session is disrupted and manual reconnection and login to the server is required.

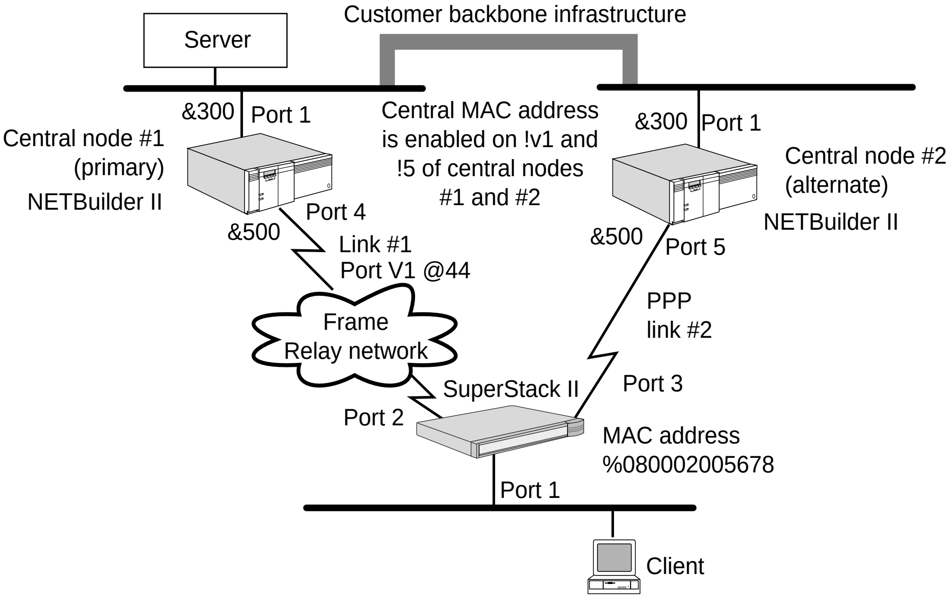


Figure 32-29 Network Resiliency with IPX Routing in a Boundary Routing Environment

AppleTalk Routing Example. When routing AppleTalk in a Boundary Routing environment, you need to configure the alternate central node #2 with the same network range, default zone, seed information, and zone list as central node #1. If you enable the use of the central MAC address on both central nodes, the switch from the primary to the alternate central node is transparent to the user.

The use of the central MAC address provides a transparent switch to the alternate central node. If link #1 or central node #1 fails when a session is established between a client on the leaf network and a server on the central site network, the session is not interrupted when the alternate central node becomes active because the same routing and addressing attributes are used on central node #2.

If you do not enable the use of the central MAC address on both of the central nodes, the session is disrupted. Because AppleTalk clients identify their next-hop router by listening to Routing Table Maintenance Protocol (RTMP) packets, their routing tables are eventually updated with a route to central node #2; however, you need to reconnect and log on to the server.

IP Routing Example. When routing IP in a Boundary Routing environment, you need to configure the alternate central node #2 with the same network addresses and routing attributes as central node #1. In addition, you must enable IP routing on both central nodes. If you enable the use of the central MAC address on both central nodes, the switch from the primary to the alternate central node is transparent to the user.

The use of the central MAC address provides a transparent switch to the alternate central node when clients on the leaf network have a single default gateway configured. These clients use ARP to obtain the gateway address and cache its MAC address. The clients may or may not use unsolicited ARP responses to update their caches. However, by configuring the same network addresses and enabling the central MAC address on both central nodes, if link #1 or central node #1 fails when a session is established between a client on the leaf network and a server on the central site network, the session is not interrupted when the alternate central node becomes active. The client continues to use the MAC address and network address that is stored in its cache, although the client is accessing the server through link #2 and alternate central node #2.

If you do not enable the use of the central MAC address on both of the central nodes when the clients have a single default gateway configured, the session is disrupted. You need to reconnect and log on to the server.

Transparent Bridging Example. When bridging in a Boundary Routing environment, you need to enable transparent bridging on both central nodes. If you enable the use of the central MAC address on both central nodes, the switch from the primary to the alternate central node is transparent to the user.

If link #1 or central node #1 fails when a session is established between a client on the leaf network and a server on the central site network, the session is not interrupted when the alternate central node becomes active. Packets are automatically forwarded on link #2, assuming that the Spanning Tree Protocol is active so that a previously blocked path to the destination is unblocked, and that activation of link #2 occurs before the application’s session times out.

33 CONFIGURING AUTO STARTUP

This chapter describes the tools you need to configure the auto startup feature, how to configure the feature, and how the feature works.

Necessary Configuration for Auto Startup

The auto startup feature enables the peripheral node (SuperStack II NETBuilder boundary router) in a Boundary Routing topology to boot and become operational with no or minimal software configuration.

Before configuring auto startup, 3Com strongly recommends that you read "How Auto Startup Works" on page 33-8.

For the auto startup feature to work, you need to configure the central node and the BOOTP and Trivial File Transfer Protocol (TFTP) servers on the central site network. (For information on the 3Com bridge/routers that can be used as a central node, refer to Table 32-1.)

If you are using a model 42x or 52x SuperStack II bridge/router as a peripheral node and if this node is connected to an Integrated Services Digital Network (ISDN) switch type other than European Telecommunications Standards Institute (ETSI), additional configuration is necessary. For more information, refer to "Configuring Boundary Router Software on Model 42x or 52x Bridge/Routers" on page 33-7.

Preparing for the Configuration

For the auto startup feature to work, the central node requires certain software tools and some software configuration.

Tools

You need to configure a BOOTP server and a TFTP server on the central site network. Table 33-1 lists the software tools that offer the servers, indicates if they are mandatory or optional, and provides a short explanation of each tool.

Table 33-1 Software Tools to Configure Auto Startup Phase 2

Software Tool	Usage	Explanation
3Com's Remote Upgrade Utilities	Optional*	Compatible with the SunOS 4.1.x, Solaris 2.4, and HP-UX A-09.0x environments. These utilities include a BOOTP server.
BOOTP Server	Optional*	BOOTP server supplied by another vendor.
TFTP Server	Mandatory	The TFTP server application runs on any platform. It acts as a repository of firmware, software, and configuration files. A TFTP server is a part of the UNIX system software.

* Use either 3Com's or another vendor's version of a BOOTP server.

For information on configuring the BOOTP and TFTP servers, refer to "Configuring the Central Node and the BOOTP and TFTP Servers" on page 33-2.

Prerequisites Before beginning this procedure, complete the following tasks on the central node:

- Log on with Network Manager privilege.
- Set up ports and paths according to Chapter 1. Note the port numbers and associated IP addresses that will be used for the auto startup phase 2 connection.
- Configure a WAN port for the Boundary Routing feature over PPP or Frame Relay according to Chapter 32.
- If using an ISDN line, configure the ISDN interface according to Chapter 35.
- Examine your network and determine (or assign) the IP address and/or the MAC address for each SuperStack II boundary router that requires auto startup support. (The procedure for assigning an IP and/or MAC address for a boundary router is included in the procedure for configuring the Boundary Routing feature over PPP and Frame Relay in Chapter 32.)
- Determine on which server the BOOTP server will reside.
- Determine on which server the TFTP server will reside.
- For Frame Relay configurations, determine the data link connection identifier (DLCI) for each SuperStack II boundary router that requires auto startup phase 2 support. The DLCI is assigned to the Frame Relay interface by the public data network (PDN) service vendor.
- If using a peripheral node with a token ring interface, configure ring and bridge numbers, which will be downloaded from the central node to the peripheral node, using the -SR RingNumber and -SR BridgeNumber parameters. For more information on these parameters, refer to *Reference for NETBuilder Family Software*.

The DLCI is normally learned automatically by the interface. However, for the auto startup feature to work properly, you may need to include this value in the bootptab file entry if you are using the 3Com Remote Upgrade Utilities.

Configuring the Central Node and the BOOTP and TFTP Servers

Figure 33-1, Figure 33-2, and Figure 33-3 show sample Boundary Routing topologies in which auto startup is configured for Frame Relay, PPP, and ISDN/PPP.

The following items are the same in all the sample topologies

- A NETBuilder II bridge/router is the central node and a SuperStack II boundary router is the peripheral node. Although all sample topologies depict Ethernet as the LAN medium, token ring can also be used.
- A BOOTP server and a TFTP server are set up on Sun, HP, or PC systems. The BOOTP server has the IP address 129.213.201.25; the TFTP server has the IP address 129.213.201.24.
- The central node is functioning as a bridge/router with an IP address of 129.213.201.21. The User Datagram Protocol (UDP) Broadcast Helper feature has been enabled on it.

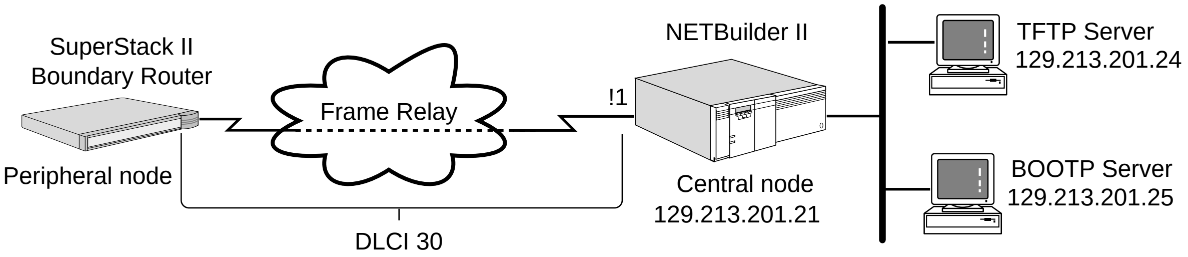


Figure 33-1 Configuring Auto Startup for Frame Relay

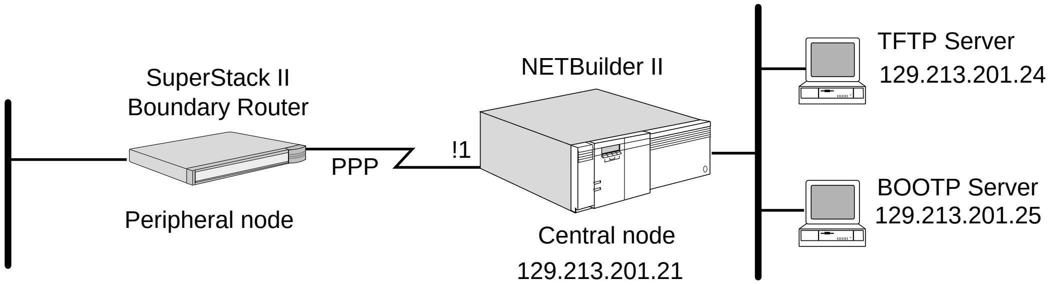


Figure 33-2 Configuring Auto Startup for PPP

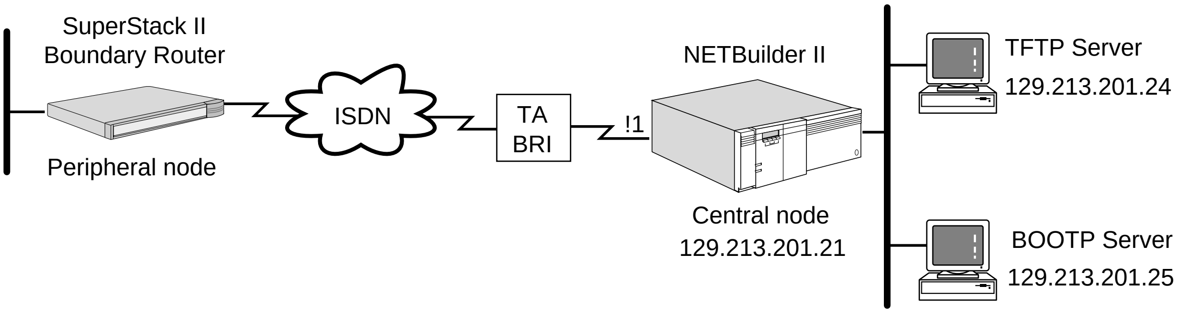


Figure 33-3 Configuring Auto Startup for PPP/ISDN

The Frame Relay connection between the central and peripheral nodes in Figure 33-1 has a DLCI of 30 assigned to it by the Frame Relay service vendor.

Procedure To configure the central node and the BOOTP and TFTP servers on the central site network, follow these steps:

1 Configure the UDP Broadcast Helper on the central node.

a Enable UDP Broadcast Helper by entering:

```
SETDefault -UDPHELP CONTrol = Enable
```

b Add BOOTP server UDP port 67 to the active port list using:

```
ADD -UDPHELP ActivePorts {<UDP port> | <name>}
```

BPSERVER is the name reserved for port 67.

For example, to add UDP port 67 to the active port list, enter:

```
ADD -UDPHELP ActivePorts 67
```

or

```
ADD -UDPHELP ActivePorts BPSERVER
```

c Add the IP address of the BOOTP server into the forward address list using:

```
ADD -UDPHELP ForwardAddress <UDP port or name> <IP address>
```

For example, in the sample topologies shown, add the IP address of the BOOTP server (129.213.201.25) to the forward address list by entering:

```
ADD -UDPHELP ForwardAddress 67 129.213.201.25
```

For more information on the UDP Broadcast Helper, refer to Chapter 20.

2 Install either 3Com's or another vendor's version of the BOOTP server on your Sun, HP, or PC system. If you plan to install another vendor's version of the BOOTP server, skip this step and refer to the documentation that accompanies that product for information.

a Install the 3Com Remote Upgrade Utilities on the Sun, HP, or PC system that will function as the BOOTP server.

These utilities are provided by 3Com on CD-ROM and contain the 3Com implementation of the Bootpd program, which is defined in RFC 951 and RFC 1048. When Bootpd starts, it reads its configuration file /etc/bootptab, then sends a BOOTREPLY packet based on the contents of the /etc/bootptab file for a BOOTREQUEST.

The distribution diskette contains an installation script and UNIX manual pages to document the command line syntax of the utilities. For more information about the Remote Upgrade Utilities, refer to *Upgrading NETBuilder Family Software*.

b Edit the /etc/bootptab file in the BOOTP server file directory.

The /etc/bootptab file contains configuration parameters that must be set up before a SuperStack II boundary router can execute phase 2 of the auto startup process.

The bootptab file has a format similar to that of the termcap file in which two-character, case-sensitive tag symbols are used to represent parameters. The parameter declarations are separated by colons (:). The general format is as follows:

```
hostname:tg=value.......:tg=value.......:tg=value...
```

where:

hostname	is the actual name of a BOOTP client (the peripheral node).
tg	is a two-character tag symbol. Most tags must be followed by an equal sign (=) and a value.

You can access a complete description of the bootptab file and its construction using the online manual page facility that comes with the utilities package.

Read the contents of the bootptab file. At the end of the file, you will find examples that you can edit to fit your network topology.

For each peripheral node that is expected to request a boot load from the central site server, an entry must be made into the bootptab file. The entry for the sample topology shown in Figure 33-1 contains the following information:

```
remote:ip=129.213.201.22:hp=1:sm=255.255.255.0:\
:hd=config-files:bf=boot.68k:bh=129.213.201.21:\
:hh=frame:ci=30:fs=129.213.201.24
```

where:

`remote`	is the name of the SuperStack II boundary router.
`ip=129.213.201.22`	is the IP address that is assigned to the system named “remote.”
`hp=1`	is the number of the central node port that is connected to the peripheral node.
`sm=255.255.255.0`	is the subnet mask.
`hd=config files`	is the pathname of the home directory on the TFTP server where the configuration files exist.
`bf=boot.68k`	is the name of the boot file.
`bh=129.213.201.21`	is the IP address of the central node.
`hh=frame`	is the interface type of the central node port that is connected to the peripheral node. For a PPP link, specify hh=ppp.
`ci=30`	is the connection ID number (assigned by the Frame Relay service vendor). You do not need to specify this tag for a PPP link.
`fs=129.213.201.24`	is the IP address of the TFTP server.

Another important parameter is gs, which specifies the IP address of the gateway. If the central node is functioning as a router, you must specify a gateway address.

3 Set up the TFTP server.

The file server (IP address) pointed to by the fs tag in the bootptab file must have a TFTP server mechanism. Any UNIX-based operating system supports this requirement. TFTP services can also be provided by other network operating systems.

a Create the configuration file directory on the TFTP server.

You can create any directory as long as your BOOTP server can support the Root Path option. When you use the 3Com Remote Upgrade Utilities for your BOOTP server, create the directory that is specified in the “hd” parameter in the /etc/bootptab file. Otherwise, create a directory with the pathname as ..../CLIENTS/<MAC address>, where “....” is the path of the host image files’ (as specified by the bf parameter in step 2b) grandparent directory and <MAC address> is the MAC address of the SuperStack II boundary router. Make sure that this directory name matches the entry in the bootptab file you created on the BOOTP server.

For example, if the host image file is under /tftpboot/image/boot.29K, create your configuration files directory as /tftpboot/CLIENTS/<MAC address>.

The Boot Image file string /tftpboot/image/boot29k appears as the “file” filed in the BOOTREPLY packet.

For the TFTP server, the pathname is case-sensitive. The directory CLIENTS must be uppercase. System-dependent path separators // or \ are both acceptable characters.

b Create configuration files for each peripheral node.

Refer to “Configuring Boundary Router Software From the Central Site” on page 33-6 for information on how to create the configuration files. If your peripheral node is a model 42x or 52x SuperStack II bridge/router, you are using the ISDN interface as the link to the peripheral node, and the ISDN interface is connected to a switch type other than ETSI, also refer to “Configuring Boundary Router Software on Model 42x or 52x Bridge/Routers” on page 33-7.

c Copy the configuration files that you just created from the system you used to create the files to the appropriate directory on the TFTP server.

d Create the CONFFILE file.

Using a text editor, create an ASCII text file named CONFFILE in the same directory on the TFTP server. CONFFILE is a text file that contains configuration filenames. This file must contain the filenames of all the configuration files that the TFTP server provides for the associated client. For example, a CONFFILE can contain the following contents:

```
ip<sep> iprip<sep>rtmnet<sep>system<sep>
```

Where <sep> is the separator of each file. The <sep> can be a blank, a tab, a form feed, a carriage return, or a new-line character.

4 When all the required files and services are in place at the central site, you are ready to set up the SuperStack II boundary router.

For information on installing and cabling the SuperStack II boundary router, refer to the documentation that accompanies the hardware.

5 Plug in the SuperStack II boundary router.

The boundary router starts up. The initiation of the auto startup phase 2 process depends on the following line types used for the physical link:

- When a leased line is used, plug in the appropriate cables between the SuperStack II boundary router and the modem to which the leased line is connected.
- When a dial-up line is used, configure the modem or channel service unit/digital service unit (CSU/DSU) to dial the central node so that a physical link can be established.

For model 42x and 527 SuperStack II bridge/routers, you can either configure a phone number in the modem so that it automatically dials out or set up the modem to accept an incoming call from the central site for the data terminal equipment (DTE) (serial) interface. The ISDN interface can only accept an incoming call and cannot dial the central site at initial power on.

Configuring Boundary Router Software From the Central Site

You must create configuration files containing desired changes to the boundary router software’s default settings using NETBuilder software commands, parameters, and syntax on a platform that is exactly the same as the boundary router that you are configuring. For example, if the boundary router you are configuring is a model 221 SuperStack II bridge/router, then you must create the configuration files on a model 221 bridge/router. Copy the files to the TFTP server. (For complete information on NETBuilder commands and parameters, refer to *Reference for NETBuilder Family Software*.)

Configuring Boundary Router Software on Model 42x or 52x Bridge/Routers

Table 33-2 lists the parameters that you need to include in the configuration files to bring up a Frame Relay, PPP, or ISDN/PPP line on a SuperStack II boundary router with an ISDN interface (model 42x or 52x bridge/routers).

Table 33-2 Parameters That Must Be Configured on Model 42x and 52x Bridge/Routers

Parameter	Applies to Which Interface Type?	Setting
-PORT DialInitState	DTE and ISDN	DialOnDemand.
-PATH LineType	DTE and ISDN	Dialup or Leased for DTE; Dialup for ISDN.
-PATH LocalDialNo	ISDN	Phone number provided by your phone company.
-PORT DialNoList	DTE and ISDN	Phone number for remote site being dialed.
-PATH LocalSubAddr	ISDN	Specify only if you need to configure a subaddress to the phone number specified by -PATH LocalDialNo parameter.
-PATH SPIDdn1 and SPIDdn2	ISDN	Specify for North American (U.S. and Canada) only. Service profile identifiers (SPIDs) and directory numbers (DNs) provided by phone company.
-PATH BAud	DTE and ISDN	Specify only if using a baud rate other than the default of 64 kbps.
-PATH RateAdaption	ISDN	Specify only if using a 56K ISDN line speed.
-PATH SwitchType	ISDN	Specify only if connecting to a switch type other than the default of ETSI.*
-PATH CONNector	DTE	Explicitly associates a path with serial connector marked A/B on a model 42x SuperStack II bridge/router or marked B on model 32x and 52x bridge/routers. You must enter this command with -PORT OWNer to enable auto startup on model 32x and 52x bridge/routers.
-PORT OWNer	DTE	Explicitly configures auto startup on SuperStack II boundary routers. You must enter this command with -PATH CONNector to enable auto startup model 32x and 52xbridge/routers.
-PATH DialMode	DTE	Specify only if using a modem other than the default of a V.25bis-compatible modem.

* If the boundary router is configured to interface with the wrong switch type, the path will not come up and auto startup will fail during phase 1. This parameter must be configured on the boundary router itself. For complete information, refer to "Configuring Boundary Router Software From the Central Site" on page 33-6.

For complete information on the syntax for the parameters listed in Table 33-2, refer to *Reference for NETBuilder Family Software*.

Upon initial startup, the boundary router attempts to download the configuration files from the TFTP server only if no configuration files exist on the boundary router. This process does not occur when you are using the ISDN interface on model 42x and 52x SuperStack II bridge/routers as your link to the central node, and when the ISDN interface is connected to a switch type other than the default of ETSI. In this situation, you must attach a terminal to the

Console connector on the peripheral node and reconfigure the -PATH SwitchType parameter. Reconfiguring this parameter locally causes a ppm configuration file to be created and stored on the SuperStack II boundary router.

You must also create a SYSTEM file on the peripheral node that sets the -SYS GetConfigFiles parameter to ON. Reconfiguring this parameter prevents the peripheral node from assuming that the ppm configuration file that already exists on the peripheral node is sufficient and that configuration file download from the TFTP server is not necessary.

CAUTION: *Do not reconfigure the setting of the -SYS GetConfigFiles parameter to ON except in the situation described in this section. Otherwise, configuration files will be downloaded each time you reboot your boundary router, potentially overwriting more current configuration files that may exist on your boundary router.*

How Auto Startup Works

The auto startup feature is a two-phase process. This section describes what happens during each phase.

Auto Startup Phase 1

Phase 1 of the auto startup process begins when the SuperStack II boundary router is plugged in.

During phase 1, a peripheral node (the SuperStack II boundary router) in a Boundary Routing topology automatically detects certain local and wide area port and path attributes. Table 33-3 lists the detected attributes.

Table 33-3 Detected Peripheral Node Attributes

Local Area Path Attribute Detected	Wide Area Path and Port Attributes Detected
Token ring speed*	WAN protocol (PPP or Frame Relay) that runs on a port
	Line type, for example, leased or dial-up
	DTE connector type that you have cabled†

* Applies to model 32x and 52x bridge/routers only.
† Applies to model 42x bridge/routers only

Phase 1 for model 42x bridge/routers detects the path attributes instantly on initial system startup; however, if the connector is changed during normal operation, it can take several minutes for the auto detection software to sense the connector. For more information, refer to "Automatic Attribute Detection for DTE Ports on Model 42x Bridge/Routers" on page 33-8.

After the attributes listed in Table 33-3 are detected, the boundary router establishes a physical link and then a data link between itself and the central node. Phase 1 is complete, and phase 2 of the auto startup process begins. For more information, refer to "Auto Startup Phase 2" on page 33-9.

Automatic Attribute Detection for DTE Ports on Model 42x Bridge/Routers

For model 42x SuperStack II bridge/routers, the auto startup phase 1 process also detects the DTE port you have cabled. This process only works only on DTE ports and only when the -PORT OWNer parameter is set to AUTO (default). Autostartup can take several (three to five) minutes if the cable is changed during normal system operation.

When establishing the physical and data links between themselves and the central node, model 42x bridge/routers attempt to detect the connector type, owner, and line type of the path associated with the cabled DTE port. The boundary router detects these path characteristics by first attempting to try connector and line type combinations. The connector and line types are tried in the following order:

- RS-232 with leased line
- RS-232 with dial-up line
- RS-449/V.36 with leased line
- RS-449/V.36 with dial-up line
- V.35 with leased line
- V.35 with dial-up line

The boundary router scans each line quickly to determine the connector type most likely being used. After it successfully detects the connector and line types, the boundary router tries to detect the owner using a similar process. The scanning process continues periodically so that connector changes can be quickly determined.

The possible owners are tried in the following order:

- PPP
- Frame Relay

Knowing the order in which the connector and line types and owners are tried can help you anticipate how long the establishment of the physical and data links will take. For example, the detection of a V.36 dial-up line running Frame Relay will take longer than the detection of an RS-232 dial-up line running Frame Relay because the V.36 dial-up line is tried later than an RS-232 dial-up line.

To determine the progress of the establishment of the physical and data links, follow these steps:

1 Enter:

```
SHow -PORT DIAGnostics
```

A display shows you which connector type, port owner, and line type has been tried and deemed a failure, and which is currently being tried.

2 Look at the LEDs on model 42x bridge/routers associated with the DTE connectors to determine which connector is currently being tried.

Because of a signal irregularity in the RS-449 connector, the auto startup detection feature occasionally reports that the RS-449 connector is a V.35 connector. This condition eventually corrects itself.

Auto Startup Phase 2

During phase 2 of the auto startup process, the peripheral node obtains necessary configuration information from central site servers across the PPP or Frame Relay line. The peripheral node obtains information including:

- IP address
- Boot file location

- Configuration files
- Bridge and ring numbers (applies only to token ring interfaces on model 32x and 52x SuperStack II bridge/routers)

For phase 2 to work, you must configure the UDP Broadcast Helper feature on the central node and you must configure two servers on the central site network: a BOOTP server and a TFTP server. The BOOTP server “listens” for BOOTP requests and forwards an IP address and boot file location information to the peripheral node. The TFTP server forwards configuration files to the peripheral node.

During phase 2:

- The peripheral node boundary router broadcasts a BOOTP request packet to the central node.
- The central node forwards the BOOTP request to the BOOTP server on the central site network using the UDP Broadcast Helper feature.
- The BOOTP server replies to the broadcast BOOTP request packet. This reply contains IP addresses for the boundary router and TFTP server and the location of the appropriate configuration files.
- The boundary router sends read request packets that request certain configuration files from the TFTP server on the central site network.
- When the TFTP server receives the read request packet, it begins to transfer the requested configuration files to the boundary router. The file transfer proceeds as a series of transfers and acknowledgments until the file transfer is complete.
- When the file transfer is complete, the boundary router automatically reboots and applies the newly acquired configuration files.
- The central node detects a boundary router with a token ring interface and downloads bridge and ring numbers to this boundary router.

34 WIDE AREA NETWORKING USING PPP AND PLG

This chapter describes how to set up and configure wide area networking using the Point-to-Point (PPP) and Phone Line Gateway (PLG) Protocols.

The wide area bridge/router supports PPP and PLG for point-to-point communication. PPP is a standard protocol that provides serial line connectivity between two NETBuilder bridge/router or between a NETBuilder bridge/router and a bridge/router built by another vendor running PPP.

PLG is a 3Com-proprietary wide area protocol that is designed to provide optimal use of the serial line. It provides serial line connectivity between 3Com bridge/routers (NETBuilder II) only. You cannot bridge packets received from token ring and FDDI interfaces over PLG, but you can bridge packets received from Ethernet.

For conceptual information about PPP, refer to "How PPP Works" on page 34-5.

Setting Up Point-to-Point Protocol Communication

Only one wide area protocol is allowed to run over one port, regardless of the number of paths assigned to the port. Figure 34-1 is an example in which only one path has been assigned to one port. In this figure, bridge/router 1 is running PPP over port 3 and PLG over port 4. Figure 34-2 is an example in which two paths have been assigned to one port. In Figure 34-2, bridge/router 1 is running PPP over port 3, which has paths 3 and 4 assigned to it.

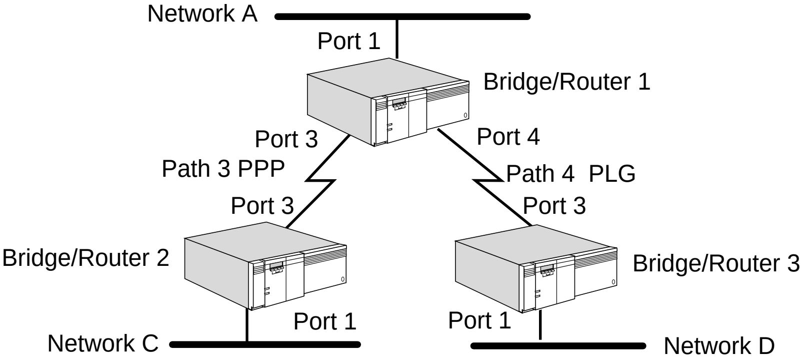


Figure 34-1 One Path per Port Configuration

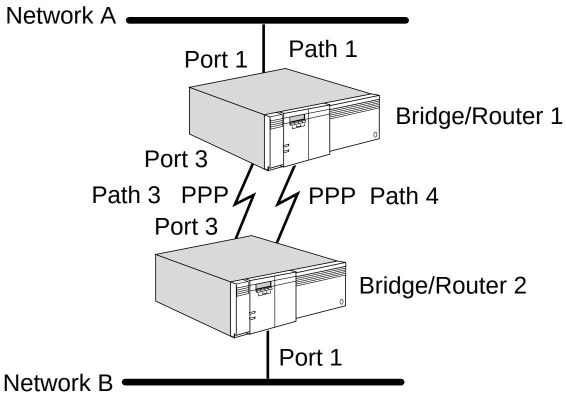


Figure 34-2 Two Paths per Port Configuration

If you assign multiple paths to one port, as shown in Figure 34-2, the load sharing feature is enabled. For more information on load sharing, refer to "Load Sharing and Load Balancing" on page 34-7.

Serial lines running PPP or PLG can bridge or route all protocols supported by the NETBuilder bridge/router [Bridging, Transmission Control Protocol/Internet Protocol (TCP/IP), Xerox Network Systems (XNS), open system interconnection (OSI), internetwork packet exchange (IPX), DECnet, AppleTalk, and VINES].

By default, PPP is enabled on serial interfaces.

Enabling PPP or PLG

If your bridge/router is built by another vendor, follow that vendor's instructions for enabling PPP.

While enabling PPP, keep in mind the following considerations:

- Before configuring PPP or PLG, you must be logged on as Network Manager.
- PPP is the default protocol for serial interfaces on NETBuilder II bridge/routers, and it is automatically enabled. On other platforms, the owner is AUTO.
- When you enable PPP on the NETBuilder bridge/router, you must also enable PPP on the bridge/router at the other end of the serial connection.

If the owner of a port is not PPP, use:

```
SETDefault !<port> -PORT OWNer = PPP
```

If PORT OWNer is set to AUTO, PPP is detected and automatically configured so you must override it and statically configure PLG.

To enable PLG, use:

```
SETDefault !<port> -PORT OWNer = PLG
```

Setting an Authentication Protocol

PPP can be configured to prevent unauthorized access especially for dial-up lines over the Public Switched Telephone Network (PSTN), and also to administer multiple remote users. PPP handles authentication using either the Password Authentication Protocol (PAP) or the Challenge Handshake Authentication Protocol (CHAP).

When you are setting up authentication, you must specify a userid and password pair as part of the -PPP AuthLocalUser and AuthRemoteUser parameters. How the pair is used depends on whether PAP or CHAP is configured as the authentication protocol with the AuthProTocol parameter. With CHAP, you can also specify an interval value in minutes to repeat authentication and to ensure that the identity of a peer has not changed after a link is established. The AuthReptIntvl parameter sets the interval value.

If you are running pre-7.1 NETBuilder software on remote peers, you must specify None as the protocol. This setting maintains compatibility with configurations you may have set up using earlier software versions.

Setting Up PAP

To set up PAP, follow these steps:

If you have pre-7.1 software running on the remote bridge/router, you must choose None as the userid when specifying AuthLocalUser and AuthRemoteUser. The password used for both AuthLocalUser and AuthRemoteUser must be the same as the password used by the remote bridge/router. You must also specify PAP as the AuthProTocol.

1 Specify the AuthLocalUser parameter with a userid and password using:

```
SETDefault !<port> -PPP AuthLocalUser = [“<userid>” | None],
   “<password>”
```

Remember to enclose the password in double quotes. The userid and password are case-sensitive and can be up to 16 ASCII characters long.

You can set up PAP so that either end of the link may initiate authentication for a session. However, you must have version 7.1 or higher software configured at both ends of the link.

2 Configure your remote user identification information.

Both the userid and password must be specified so that multiple remote users have unique user identification information. Do this using:

```
ADD !<port> -PPP AuthRemoteUser [“<userid>” | None], “<password>”
```

If you are setting up PAP for remote use, make sure that the remote userid and password pairs are added as AuthRemoteUser entries at the local end, and that the local userid and password pair is added at the remote end.

3 Enable PAP as your authentication protocol using:

```
SETDefault !<port> -PPP AuthProTocol = Pap
```

4 Enable all the settings you have specified in the previous steps using:

```
SETDefault !<port> -PPP CONTrol = Enabled
```

Setting Up CHAP

To set up CHAP, follow these steps. You must always specify a userid.

1 Specify the AuthLocalUser parameter with a userid and password using:

```
SETDefault !<port> -PPP AuthLocalUser = “<userid>”, “<password>”
```

The password and userid are case-sensitive and can be up to 16 printable ASCII characters long.

2 Configure your remote user identification information.

Both the userid and password must be specified so that multiple remote users have unique user identification information. Do this using:

```
ADD !<port> -PPP AuthRemoteUser “<userid>”, “<password>”
```

If you are setting up CHAP for remote use, make sure that the remote userid and password pairs are added as AuthRemoteUser entries at the local end, and that the local userid and password pair is added at the remote end.

3 Optionally, specify how often CHAP will repeat authentication to verify the identity of the remote user using:

```
SETDefault !<port> -PPP AuthReptIntvl = <minutes> (0–255)
```

If you specify 0, repeat authentication will be disabled.

4 Enable CHAP as your authentication protocol using:

```
SETDefault !<port> -PPP AuthProTocol = Chap
```

5 Enable all the settings you have specified in the previous steps using:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

Verifying Your Configuration

To verify that you have configured PAP and CHAP with settings you intended, follow these steps:

1 Verify your settings for the AuthLocalUser parameter using:

```
SHowDefault !<port> -PPP AuthLocalUser
```

The AuthLocalUsers are displayed for each port you have configured.

2 Verify your settings for AuthRemoteUser using:

```
SHowDefault !<port> -PPP AuthRemoteUser
```

The userids are displayed for remote users you added.

3 Verify the protocol that was configured using:

```
SHowDefault !<port> -PPP AuthProtocol
```

The authentication protocol you have configured is displayed.

Activating LAPB to Reduce Noisy Lines

Normally, when the PPP Protocol is used, the LAPB Service is not required to be active. However, to solve the problem of noisy lines when using these protocols, you need to activate the LAPB Service. The bridge/router has noisy lines if it experiences the following problems:

- A temporary lack of response occurs even though the other end is active.
- LAPB assumes that a line is down, even when it is not, because the clock has gone down temporarily.
- Frames need to be retransmitted frequently.

While LAPB provides a reliable data link, it does add some protocol overhead. Consequently, you need to evaluate the need for LAPB by monitoring the error rates on your lines. To activate LAPB, refer to the link-level compression procedure in Chapter 39.

To configure LAPB for noisy lines, follow these steps:

1 Increase the T1 parameter to lengthen the amount of time that LAPB waits for an acknowledgment.

This ensures that LAPB does not retransmit a frame unnecessarily.

For example, to increase the amount of time to 4,000 milliseconds on path 3, enter:

```
SETDefault !3 -LAPB T1 = 4000
```

In selecting your value for the T1 parameter be aware that the value you enter is internally divided by 250 milliseconds. As a result, any value you enter less than 250 actually equals zero.

2 Increase the N2 parameter.

This increases the maximum number of times a frame is sent after a timeout.

For example, to increase the N2 parameter value to 12 on path 3, enter:

```
SETDefault !3 -LAPB N2 = 12
```

3 Decrease the size of the FrameSeq parameter.

When the line is noisy, keep the window size low to keep the number of unacknowledged frames low. In addition, a low window size reduces the number of retransmittals required when the software encounters a corrupted packet.

For example, to decrease the FrameSeq parameter to basic sequencing on path 3 enter:

```
SETDefault !3 -LAPB FrameSeq = Basic
```

How PPP Works

After you have set up and checked the serial lines, PPP performs the following functions:

- Negotiates the maximum size of a packet that can be received over a serial line
- Manages a serial line
- Maintains serial line quality

This section describes the concepts involved in the PPP activities and explains how you can customize PPP operations under different circumstances.

Packet Size Negotiation

During bridge/router startup, the two bridge/routers connected by the serial line negotiate the maximum size of a packet that each can receive. Once the bridge/routers agree on the size of the packet, the negotiation is complete and bridging or routing can begin.

To change the size of the packet, use:

```
SETDefault !<port> -PPP MaxRcvUnit = <bytes>(1–4500)
```

For example, if you want to change the size of the packet from the default of 1,500 (4,500 in the case of token ring) to 1,000 bytes on port 3, enter:

```
SETDefault !3 -PPP MaxRcvUnit = 1000
```

In this example, even if the packet size is negotiated between both bridge/routers to be 1,000 bytes, the bridge/router will continue to receive packets up to 1,500 bytes in size. However, it will discard packets greater than 1,500 bytes in size.

Serial Line Management

A common problem experienced with T1 lines is the loopback of packets. If a T1 line goes down, packets transmitted are looped back as shown in Figure 34-3. Because the packets are looped back, the bridge/routers may perceive that they are still being sent and received and may not realize that the line is down. To detect a physical loopback problem, each link management packet sent from a bridge/router includes a *magic number* that is checked upon receipt.

The actual time to bring down a line varies depending on the baud rate on the link.

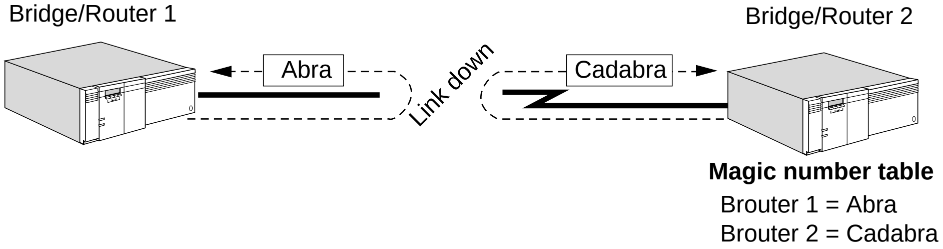


Figure 34-3 Detecting Loopback of Packets with Magic Numbers

In the configuration shown in Figure 34-3, imagine that the bridge/router 1 sends out a Link Control Protocol (LCP) packet with a magic number (Abra). When the packet is transmitted to bridge/router2, bridge/router 2 checks the magic number in the received packet against its unique magic number that it has negotiated earlier. If the magic numbers are different, it concludes that a loopback is not occurring.

If upon receipt, bridge/routers 1 and 2 check the magic numbers in their respective returned packets and detect their own magic numbers, both bridge/routers conclude that a loopback is occurring and declare the line down. (When a loopback is detected, the system message "Path <n> loopback" appears on the system console.) After the loopback is removed, the line will come back up. When PPP completes handshaking, the message "Path *x* is up" appears on the console. This message indicates that PPP is back in the normal state.

The magic number is an option that is negotiated during bridge/router startup. If both bridge/routers at the ends of the serial line connection support this option, then unique numbers are negotiated. These unique numbers are used later in the line quality management packets for loop detection.

Serial Line Quality Maintenance

PPP attempts to maintain the quality of a serial line. If data packets are not received and no echo reply is received in response to echo requests for approximately eight seconds, PPP brings the line down. Once the line goes down, PPP continuously attempts to negotiate until it successfully brings the line up.

How Authentication Works

The Password Authentication Protocol (PAP) uses a two-way handshake method to establish the identity of the peer before a link is established. PAP provides greater security than basic PPP settings. With PAP, a peer determines when the Authentication Request is made. The PeerID/Password pair is sent over the wire in a plain text form.

The CHAP uses a three-way handshake sequence to establish the identity of a peer before establishing a link. It may be repeated at any time if the AuthReptIntvl parameter is set, to ensure that the identity of the peer has not changed. CHAP relies on the equivalent of userid and password, a “name/secret” pair, to produce a challenge value used for authentication. Because the name/secret pair is never sent directly on the circuit, CHAP provides a higher level of security than PAP authentication. In the CHAP, the authenticator controls when the authentication request is made.

Load Sharing and Load Balancing

When multiple serial links are assigned to a port running PPP, load sharing or load balancing can be used on that port to make more efficient use of the bandwidth of the links.

With load sharing, the fastest link mapped to each port is selected as the primary link, and the other links are considered secondary. When data needs to be sent on that port, it is sent on the primary link until the link is saturated, then the balance of the data is sent over secondary links.

The advantage of load sharing is that if you configure the secondary link for bandwidth-on-demand, it is used only when the data exceeds the bandwidth of the primary link. The disadvantage of load sharing is that it can cause packet misordering, which may be undesirable for some network protocols.

Use load sharing only when data traffic on the port is not sensitive to packet misordering. If the data traffic has a combination of network protocols, some of which are sensitive to packet misordering, and some that are not, you need to select the sequencing feature using the mnemonic filtering scheme to ensure that packets are not misordered for sequencing sensitive protocols. Remember that sequencing works for bridged packets only.

With load balancing, data is split over parallel serial links while preserving sequencing. If the link speeds are the same, the load is split evenly. However, if the link speeds differ, the data is split in proportion to the difference between the speed of the links. For example, if you have two links, and the speed of one is 60% greater than the speed of the other, the faster link receives approximately 60% more data traffic. Through load balancing, all active links can be used to their full capacity.

Load balancing is accomplished only by using the PPP MultiLink Protocol (MLP) (RFC 1717). With MLP, each packet is assigned a sequence number to guarantee in-sequence delivery. In addition, packets may be divided into fragments, which are also assigned sequence numbers.

Packets and packet fragments are sent over the available serial links. The receiving station, also running MLP, reassembles the fragments into packets based on the sequence numbers. Once a packet is completely reassembled, it is released to the client protocol. Packets that cannot be completely reassembled due to lost fragments are discarded.

Whether packets are fragmented depends on their size and the traffic on the links. For example, some packets are too small to benefit from being fragmented. Moreover, if you have only two links, and one of them is saturated, you do not gain a performance advantage by sending packet fragments to this link.

If you use bandwidth-on-demand to back up a single primary link, and you also use load balancing, packets are sent unfragmented on the primary link. As soon as the data traffic exceeds the bandwidth of the primary link, the secondary link is brought up, which in turn enables load balancing. Once the data traffic drops below a user-specified threshold, the secondary link is brought down and the packets are once again sent unfragmented on the primary link.

If you use bandwidth-on-demand to back up multiple primary links, and you also use load balancing, packets can be fragmented over the multiple primary links. As soon as the data traffic exceeds the bandwidth of all the primary links, the secondary link is brought up, and load balancing is extended to the primary and secondary links. When traffic drops below a specified threshold, the secondary link is brought down. When this happens, packets are again load-balanced, but only across the primary links.

When you use load balancing, paths manually assigned to the same port are referred to as a bundle. Another way to create a bundle is to dynamically bind paths to a port through dial pooling. This is accomplished by using the SysCallerID port parameter, which allows you to tell the remote router the port number it must assign to the dial pooling path. Remember that you can only use SysCallerID with NETBuilder routers. You can use dial pooling with either load sharing or load balancing.

For more information about configuring the SysCallerID parameter, refer to Chapter 58 in *Reference for NETBuilder Family Software*. For more information about dial pooling paths, refer to Chapter 37.

If MLP is turned off (-ppp MlpCONTrol = Disabled), then load sharing is automatically used.

35 WIDE AREA NETWORKING USING ISDN

This chapter describes how to configure the Integrated Services Digital Network (ISDN) interface on model 42x and 52x SuperStack II NETBuilder bridge/routers.

Table 35-1 lists the steps you must perform to configure the ISDN interface and where to find the information related to each step. 3Com recommends performing these steps in the order in which they are listed.

Table 35-1 Configuring the ISDN Interface

Step	Where To Find Information
Determine the topology of your ISDN network.	"Planning Your ISDN Network" on page 35-2.
Determine how you want to use the ISDN interface.	"Deciding How to Use the ISDN Interface" on page 35-3.
Acquire ISDN services from the telephone company.	*WAN Cabling and Connectivity Guide** or the installation guide that you received with your SuperStack II NETBuilder bridge/router.
Disable phantom power if necessary.	"Disabling Phantom Power" on page 35-6.
Configure the paths, ports, and virtual ports (if necessary) associated with the B channels you plan to use.	Chapter 1 or the installation guide that you received with your SuperStack II NETBuilder bridge/router.
Configure the dial-up feature.	Chapter 37.
Configure bridging, routing, and Boundary Routing as desired over the ISDN line.	Refer to the chapters for the bridging and routing protocols you want to configure. Refer to Chapter 32 for information on Boundary Routing.
Configure PPP to run over the ISDN line.	Chapter 34.
Configure the ISDN device at the other end of the ISDN network.	"Setting Up the Remote Device" on page 35-6.

* The *WAN Cabling and Connectivity Guide* can be found on the 3Com Corporation World Wide Web site by entering: http://www.3com.com/

ISDN operates at the physical layer of the Open System Interconnection (OSI) Model. Since bridging and all routing protocols and Point-to-Point Protocol (PPP) operate at higher layers of the OSI Model, you can configure these protocols exactly as you would over a LAN or any other type of WAN interface. To configure these protocols to run over an ISDN interface, you do not need to perform additional ISDN-related steps.

For conceptual information, refer to "How the ISDN Interface Works" on page 35-6.

In this chapter, the term *ISDN interface* refers to the two B channels and the D channel. The term *B channel* refers to a specific B channel. The term *ISDN line* refers to the physical line that connects one ISDN device to another. When the ISDN line is used, it is not assumed that both B channels are being used.

Planning Your ISDN Network

3Com offers the following ISDN systems:

Models:	Model 42x and 52x SuperStack II NETBuilder bridge/routers
Function:	Boundary router or bridge/router
Number and type of ISDN interface:	1 basic rate interface (BRI) (2B+D)
Number of B channels that transmit data:	2

3Com also recommends some BRI terminal adapters (TAs) that allow a non-ISDN bridge/router, such as the NETBuilder II bridge/router, to connect to an ISDN network. For information on accessing TA recommendations, refer to the *WAN Cabling and Connectivity Guide*. You can find this guide on the 3Com World Wide Web site by entering:

`http://www.3com.com/`

The ISDN systems described above are commonly used in a few different topologies. The first is a Boundary Routing topology; the second is a traditional routed environment where all devices are meshed (connected to one another).

Figure 35-1 shows a Boundary Routing topology with a NETBuilder II system as the central node and three model 421 SuperStack II NETBuilder bridge/routers as peripheral nodes. Three 3Com-recommended TAs with BRIs connect to the NETBuilder II system with an HSS V.35 3-port module installed.

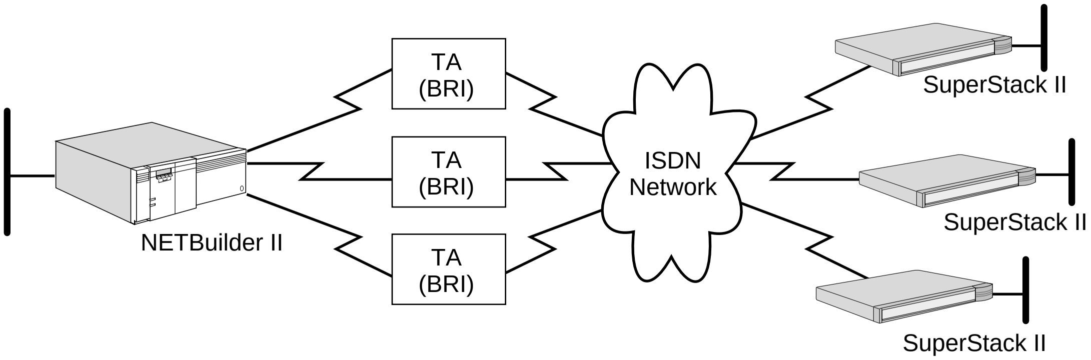


Figure 35-1 Boundary Routing Topology Using Multiple TAs with BRIs

Figure 35-2 shows a traditional routed topology where four SuperStack II NETBuilder bridge/routers that represent small offices are connected to one another. In this topology, each small office communicates directly with one another.

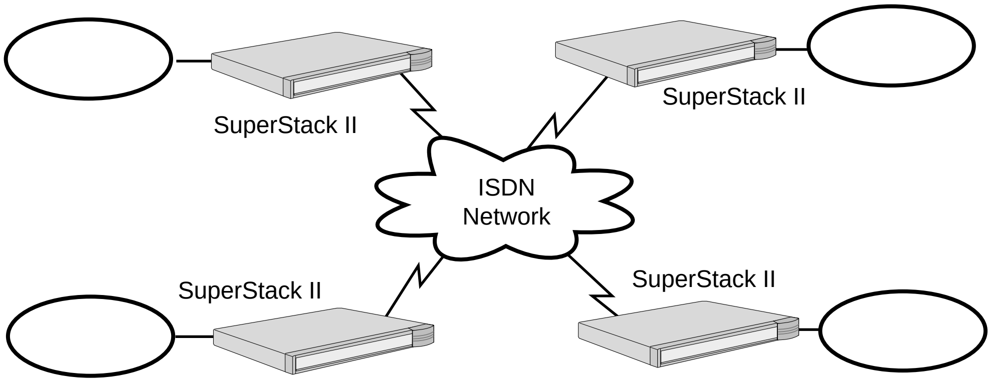


Figure 35-2 Traditional Routed Topology (Meshed)

Although the ISDN topologies discussed in this section are the most common, many others can be used. For complete information on the 3Com-recommended TAs, refer to the documentation that accompanies these devices.

Deciding How to Use the ISDN Interface

Before cabling and configuring the ISDN interface, you need to decide how you want to configure the B and D channels.

The following are some basic questions that should help you decide:

- Do you want to use one B channel only, a combination of one B channel and one data terminal equipment (DTE) interface, or both B channels?
- How do you want to use each B channel? As an interface to a primary line or as an interface to a secondary or backup line?
- If you use a B channel as an interface to a primary, dial-up line, then do you want to manually dial and hang up or use dial-on-demand?

  Dial-up is the method through which a call is placed through the D channel from one ISDN device to another ISDN device, and a 64 kbps circuit-switched B channel connection is established between the two devices. After a connection is established, data is transmitted over the B channel.
- If you use a B channel as a backup means of transmitting data, will the backup line connect the same site or a different site? Do you want the backup line to come up if the primary line fails (disaster recovery if backup to same site or network resiliency if backup to a different site) or if the primary line becomes congested and needs more bandwidth (bandwidth-on-demand)?
- Do you want to use static or dynamic paths?

Table 35-2 lists three common scenarios in which the ISDN interface is used. The figures that follow this table provide more information.

Table 35-2 Common Topologies Using ISDN Interface

Type of Topology	Primary Interface	Primary Line Type	Secondary Interface	Secondary Line Type
Boundary Routing topology using redundant routes for network resiliency/ISDN used as backup to Frame Relay or X.25	DTE interface running X.25 or Frame Relay	Leased	ISDN interface	Dial-up
Boundary Routing topology using disaster recovery, bandwidth-on-demand, and network resiliency/ISDN as backup to same site or different site	DTE interface running PPP	Leased	ISDN interface	Dial-up
Traditional routed topology/ISDN as primary using dial-on-demand	ISDN interface	Dial-on-demand	None	None

Software version 9.1 introduces the concept of *bandwidth management*, a process that applies static bandwidth, dynamic bandwidth, or a combination of these, to provide the ISDN and serial ports using PPP with the bandwidth they need to meet current requirements. Bandwidth management thinks in terms of *unrestricted*, available resources, or resources configured for a specific function such as disaster recovery only, instead of in terms of primary and secondary lines. Bandwidth management dynamically allocates or de-allocates available resources as necessary to manage link traffic. After reading the conceptual information, refer to Chapter 37 for configuration steps.

Figure 35-3 shows a Boundary Routing topology where one NETBuilder II bridge/router is connected to a SuperStack II NETBuilder bridge/router through a DTE interface over which Frame Relay or X.25 is running, while the other NETBuilder II bridge/router is connected to the same SuperStack II NETBuilder bridge/router through an ISDN line over which PPP is running. The line running Frame Relay or X.25 is considered the primary line, while the ISDN line is considered the secondary line. In this topology, the secondary line is configured to come up only if the primary line fails, which provides a redundant route for network resiliency.

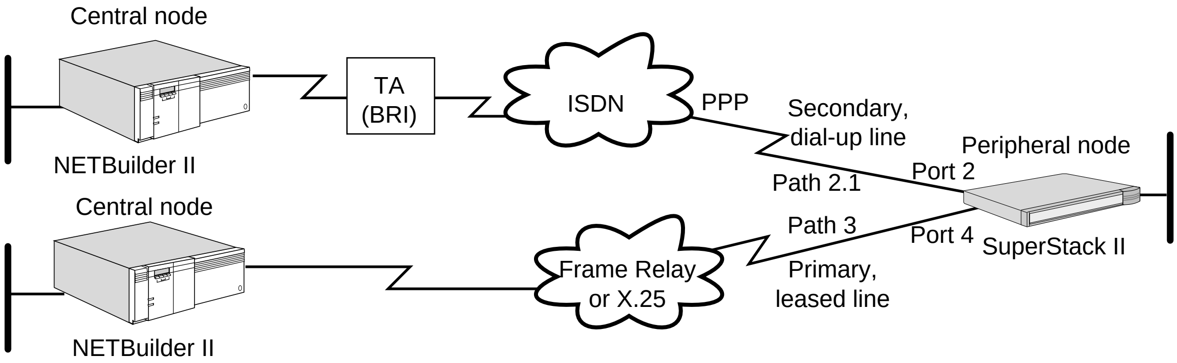


Figure 35-3 ISDN Used as Backup to Cloud Technology

Figure 35-4 shows additional Boundary Routing topologies. These topologies illustrate using an ISDN line as a backup to the same site or device, or as a backup to a different site or device.

In the first topology shown in this figure, two lines connect the NETBuilder II bridge/router to the SuperStack II NETBuilder bridge/router. The first line connects the two devices through a DTE interface over which PPP is running, while the second line connects the two devices through an ISDN path over which PPP is running. Both interfaces or paths are mapped to port 3. The DTE line is considered the primary line, and the ISDN line is considered the secondary line. In this topology, the secondary line is configured to come up only if the primary line fails (disaster recovery), or is overwhelmed by traffic and needs additional bandwidth (bandwidth-on-demand).

In the second topology shown in this figure, two lines connect two NETBuilder II bridge/routers to a SuperStack II NETBuilder bridge/router. The first line connects the two devices using a DTE interface running PPP, while the second line connects the two devices using an ISDN path running PPP. The DTE line is considered the primary line, while the ISDN line is considered the secondary line. In this topology, the secondary line is configured to come up only if the primary line fails (redundant route for network resiliency).

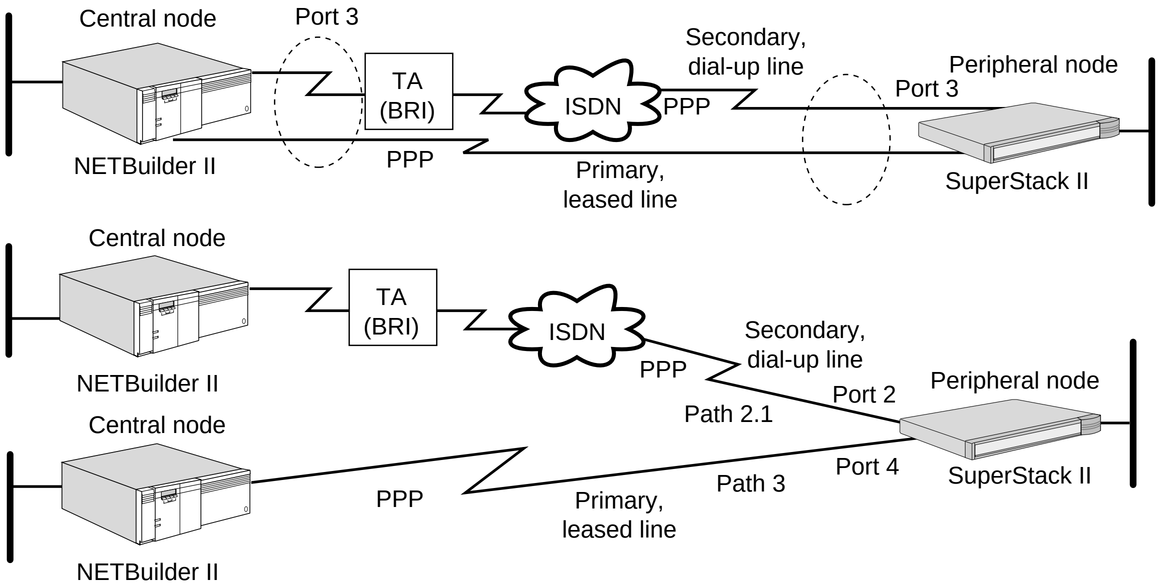


Figure 35-4 ISDN as Backup to Serial Line Running PPP (Same or Different Site)

Figure 35-5 shows a traditional routed topology where a model 527 SuperStack II NETBuilder bridge/router is connected to another model 527 bridge/router through an ISDN path over which PPP is running. Because the ISDN line provides the only connection between these two devices, it is considered the primary line. In this topology, this line is configured to come up only when there is a demand for it (dial-on-demand).

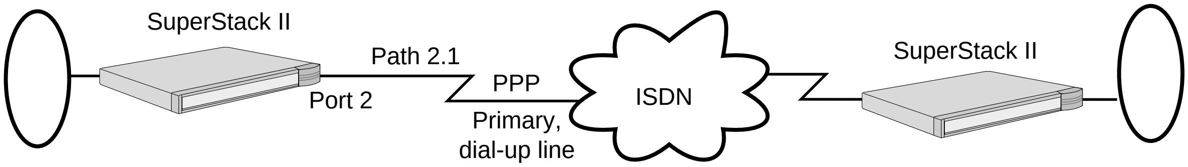


Figure 35-5 ISDN as Primary Using Dial-on-Demand

Although the examples described in the preceding paragraphs are the most common, many other examples exist. For more information on Boundary Routing using network resiliency, disaster recovery, and bandwidth-on-demand, refer to Chapter 32. For information on dial-up, including more information on disaster recovery and bandwidth-on-demand, refer to Chapter 37.

Disabling Phantom Power

A Network Termination 1 (NT1) and a power supply are required for every ISDN line in North America. Your service provider or telephone company may provide you with an NT1 and power supply for a small monthly fee, or, you may want to purchase it from an ISDN equipment vendor. The NT1 and power supply may come in a single standalone box or the two may be in separate units. In this discussion, the two units together will be referred to as an NT1.

Two kinds of NT1s are currently in use in North America, differentiated by the data encoding scheme used in the transmission of data between the NT1 and the telephone company's equipment. The two data encoding schemes are called 2B1Q (two bits mapped into one quaternary symbol) and AMI (Alternate Mark Inversion). The 2B1Q scheme is the dominant method in use today. The AMI scheme is older and rarely used.

Two power sources are available from an NT1 for CPE equipment. An ISDN telephone uses one power source. The SuperStack II NETBuilder bridge/router does not use either one for power. Instead, it detects the presence or absence of phantom power and can determine whether or not a telephone cord is plugged in.

However, not all NT1s provide phantom power. The AMI NT1 from AT&T does not. If you are connecting the SuperStack II NETBuilder bridge/router to an NT1 that does not provide phantom power, you must turn off phantom power detection before you can dial successfully. To turn off phantom power detection, set the value of the -PATH PhantomPower parameter to Disable. For more information on this parameter, refer to Chapter 42 in *Reference for NETBuilder Family Software*.

Setting Up the Remote Device

After you have configured the ISDN device at the other end of the ISDN network (use the documentation that accompanies that device), no additional configuration is necessary for that device to interoperate with your SuperStack II NETBuilder bridge/router with an ISDN interface.

How the ISDN Interface Works

This section provides conceptual information on aspects of the ISDN interface that require further explanation.

Basic Rate Interface

SuperStack II NETBuilder bridge/routers with an ISDN interface provide ISDN connectivity through a BRI. This interface consists of two full-duplex B channels operating at 64 kbps and one full-duplex D channel operating at 16 kbps (2B + D). The two B channels transmit data, while the D channel is used for call processing with the ISDN switch.

Because the BRI consists of multiple B channels over which data can be transmitted, a path numbering convention has been devised. For complete information on this convention, refer to Chapter 1 or the installation guide that you received with your SuperStack II NETBuilder bridge/router.

Paths 2.1 and 2.2 correspond to the B channels. However, a particular B channel is not statically bound to a particular path. At one time path 2.1 could use B channel 1, while at another time the same path could use B channel 2.

Some parameters that you must configure to set up the ISDN environment are connector-specific, that is, they are generically applicable to the ISDN operating environment, not to any one specific B channel. Connector-specific parameters require that you specify the connector number only. Other parameters are channel-specific, that is, they apply specified configurations to an individual channel and the physical connector which it is associated with.

Channel-specific parameters require that you specify the connector number and the channel number. If you are unsure as to whether you need to specify the connector and channel numbers or just the connector number, refer to the description of that particular parameter in *Reference for NETBuilder Family Software*.

Point-to-Point and Point-to-Multipoint Configurations

Your SuperStack II NETBuilder bridge/router with an ISDN interface can be a device in a point-to-point or point-to-multipoint configuration. A point-to-point configuration is a topology where a single device is connected to an ISDN line. A point-to-multipoint configuration is a topology where up to eight devices, including bridge/routers (SuperStack II NETBuilder bridge/router with an ISDN interface and other non-ISDN 3Com bridge/routers connected to a TA) and telephones (ISDN and non-ISDN connected to a TA), are connected to an ISDN line. The point-to-multipoint configuration is implemented using a passive S-bus.

Figure 35-6 shows two network topologies. The first topology (far left) is a point-to-point configuration; the second topology shows a point-to-multipoint configuration. Each configuration is connected to an ISDN switch through an ISDN line.

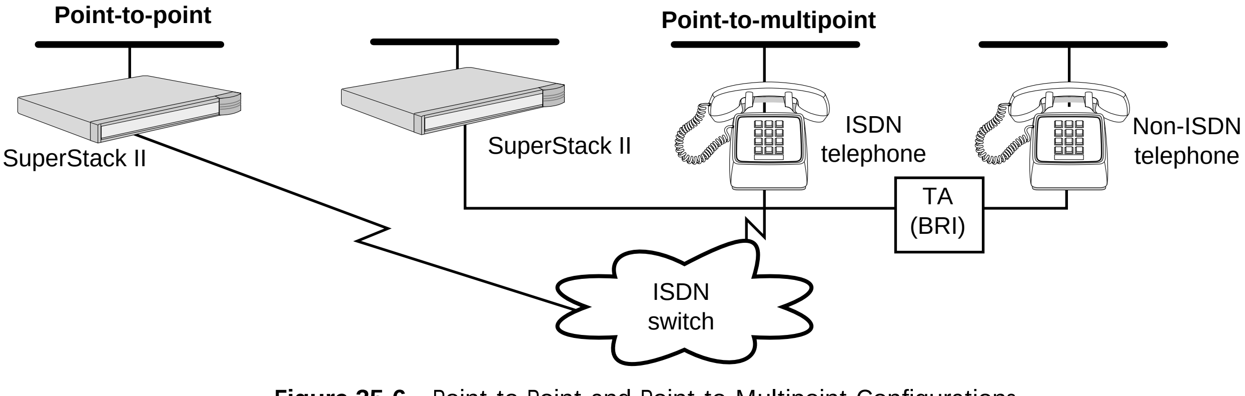


Figure 35-6 Point-to-Point and Point-to-Multipoint Configurations

How Incoming Calls Are Accepted

This section explains how SuperStack II NETBuilder bridge/routers with an ISDN interface decide to accept an incoming call from an ISDN switch.

SuperStack II NETBuilder bridge/routers with an ISDN interface use the following types of call compatibility criteria to determine whether or not to accept an incoming call from an ISDN switch:

- Bearer capability
- ISDN addressing

Some bearer capability criteria are fixed and cannot be changed, while others are determined by user configuration. The following sections describe each type of call compatibility criteria.

The ISDN specifications provide other compatibility criteria called low-layer compatibility and high-layer compatibility information elements that can be used to determine incoming call acceptability. SuperStack II NETBuilder bridge/routers with an ISDN interface do not use low-layer compatibility and high-layer compatibility information elements as criteria to determine whether or not to accept an incoming call from an ISDN interface.

Bearer Capability Compatibility

SuperStack II NETBuilder bridge/routers with an ISDN interface have the following fixed bearer capability criteria for an incoming call:

- It must be a 64K or 56K unrestricted digital data call. A voice call may be made to a telephone on a multipoint ISDN line to which the SuperStack II NETBuilder bridge/router is also connected. However, the SuperStack II NETBuilder bridge/router will not answer the call.
- It must be made in the circuit mode.

Calls that do not fulfill these criteria are rejected or ignored.

You can specify the rate at which data is to be transferred on a B channel that is to be connected by a call by using the -PATH RateAdaption parameter.

ISDN Addressing Compatibility

After an incoming call fulfills the bearer capability criteria, the following items must be determined:

- Which bridge/router or bridge/routers will answer the call?
- Which path will accept the call?

In a point-to-point configuration, where a single device is connected to an ISDN line, it is assumed that the bridge/router on the ISDN line will answer the call. Therefore, it is not necessary to configure the bridge/router as the device that will answer incoming calls.

In a point-to-multipoint configuration, where up to eight devices, including bridge/routers and ISDN and non-ISDN telephones, can be connected to an ISDN line, you must configure at least one bridge/router to answer the incoming calls.

After the incoming call is answered by at least one bridge/router in either the point-to-point and point-to-multipoint configurations, the bridge/router must also determine whether the call is to be connected to path 2.1 or 2.2. The ISDN switch selects a B channel over which to transmit a call and the bridge/router must determine the path that is to be connected with that B channel.

The selection of which bridge/router will answer a call and subsequently which path will accept a call is determined by how you address your ISDN paths. ISDN addresses consist of the following components:

- Phone number
- Subaddress

For more information, refer to “ISDN Addressing” on page 35-11. You can assign a phone number and a subaddress to an ISDN path using the -PATH LocalDialNo and -PATH LocalSubAddr parameters, respectively.

Assign a phone number to an ISDN path under the following circumstances:

- You have a point-to-point or point-to-multipoint configuration and want to use static path and port binding. For example, you may want calls from the Boston office to be accepted by a particular path and port on a particular bridge/router, and calls from the Washington office to be accepted by another path and port on another bridge/router.
- You have a point-to-multipoint configuration and plan to use dynamic dial path pooling. For example, you may want calls to be accepted by any path in a dial path pool and then dynamically bound to a port. You should specify a phone number for at least one ISDN path on a bridge/router in this topology to ensure that at least one bridge/router will answer incoming calls.
- The telephone number you plan to specify for that path is unique among the other telephone numbers specified for ISDN paths in your point-to-point or point-to-multipoint configuration. Figure 35-7 shows a point-to-point configuration where both ISDN paths are assigned a unique phone number using the -PATH LocalDialNo parameter.

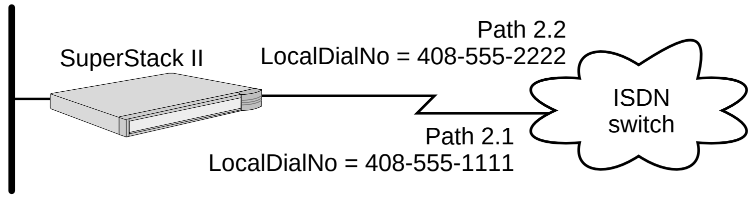


Figure 35-7 Assigning a Unique Phone Number to Multiple ISDN Paths

Suppose an incoming call specifying phone number 408-555-1111 arrives. As long as it is not engaged in another call, path 2.1 accepts the call based on the phone number specified in the incoming call. If path 2.1 was already engaged in another call or the phone number specified in the incoming call is different from that assigned to path 2.1, the call is rejected. If path 2.2 also cannot be used, the call is rejected by the bridge/router.

In addition to assigning a phone number, you should assign a subaddress to an ISDN path if the phone number is the same one that you plan to assign to another ISDN path in your point-to-point or point-to-multipoint configuration. Assigning the same phone number to all or some of the ISDN paths in a topology presents a problem: more than one ISDN path may attempt to accept a call. To resolve this problem, you can assign a subaddress to each of the bridge/router's ISDN paths with the same phone number using the -PATH LocalSubAddr parameter. For example, in the point-to-multipoint topology shown in Figure 35-8, four ISDN paths have been assigned the phone number 408-555-1234. Unique subaddresses have also been assigned to the paths of each of these devices.

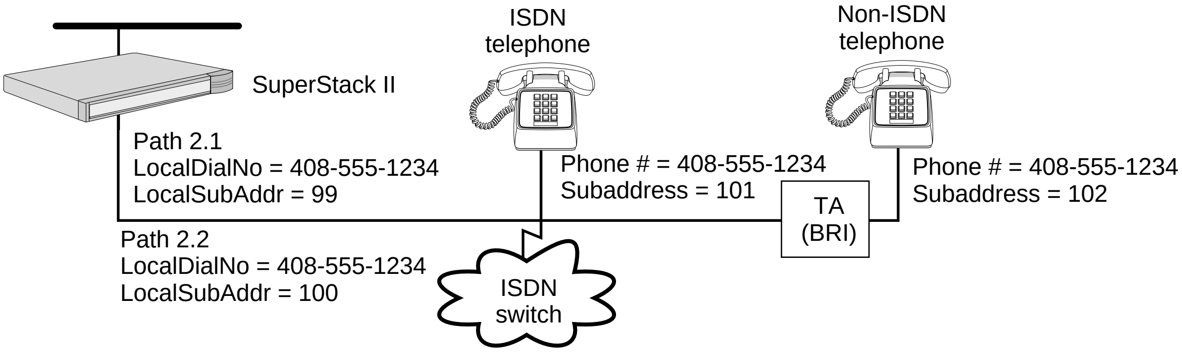


Figure 35-8 Assigning the Same Phone Number to Multiple ISDN Paths

Suppose an incoming call specifying phone number 408-555-1234, subaddress 99, arrives. The SuperStack II NETBuilder bridge/router answers the call based on the phone number specified, and as long as it is not engaged in another call, path 2.1 accepts the call based on the subaddress specified. If path 2.1 was already engaged in another call or the phone number and subaddress specified in the incoming call is different from that assigned to path 2.1, the call is rejected. If path 2.2 also cannot be used, the call is rejected by the bridge/router.

Not all telecommunications carriers allow you to assign the same phone number to multiple paths. When you contact your carrier to acquire support services, verify that they support this feature. You must also specify that you will be using subaddresses.

Do not assign a phone number or a subaddress to an ISDN path if you have a point-to-point configuration and plan to use dynamic dial path pooling. Figure 35-9 shows a point-to-point configuration where a phone number and subaddress have not been assigned for both ISDN paths.

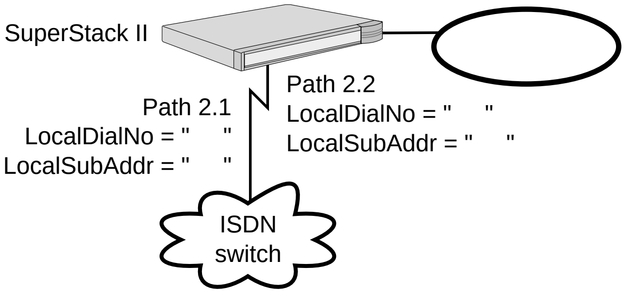


Figure 35-9 Point-to-Point Configuration Without Specified Phone Number and Subaddress

Suppose an incoming call arrives that specifies a particular phone number and subaddress. The bridge/router ignores the ISDN addressing information provided by the incoming call and not use it as criteria to determine which path should accept the call. Either path can accept the call provided that they are not engaged in another call. Criteria at higher layers of the OSI Model will determine the port to which the path will be bound to transmit this particular call.

After you have assigned phone numbers to ISDN paths and an incoming call arrives, an algorithm attempts to match the incoming phone number with the phone number specified using the -PATH LocalDialNo parameter. This algorithm compares the numbers in sequence from the end of the numbers toward the beginning. The length of the incoming phone number can be shorter than the length of the phone number configured using the -PATH LocalDialNo parameter. For example, although you can specify an international phone number using elements such as a dial prefix, country code, area code, and phone number through the -PATH LocalDial No parameter, a phone number composed of only an area code and phone number will be considered a match as long as the phone number you specified and the phone number that is received through the incoming call are the same.

Specifying an international phone number using the -PATH LocalDialNo parameter allows you to accept all calls, including international and local. If you want to restrict incoming calls to local calls only, then specify at most an area code and local phone number using the -PATH LocalDialNo parameter.

An algorithm also attempts to match the incoming subaddress with the subaddress specified using the -PATH LocalSubAddr parameter. The characters for the subaddress in the incoming message must exactly match those specified in the -PATH LocalSubAddr parameter.

For more information on static and dynamic paths and dynamic dial path pools, refer to Chapter 37. For more information on the -PATH LocalDialNo and -PATH LocalSubAddr parameters, refer to Chapter 42 in *Reference for NETBuilder Family Software.*

ISDN Addressing

An ISDN address is a phone number provided by your telecommunications carrier. The address can consist of the following elements:

Dial prefix	Identifies an international dialing code used when calling from one country to another.
Country code	Identifies the destination country or geographic area and is from 1 to 3 digits long.
Area code	Identifies a particular ISDN network within the previously defined country or geographic area.
Local phone number	Identifies a subscriber's phone number in the previously defined area code.
Remote phone number	Identifies the destination phone number.
Subaddress	Identifies the destination device within the subscriber's passive bus topology.

The telecommunications carrier does not provide a subaddress; you must create your own subaddress. For information on when to use a subaddress, refer to "ISDN Addressing Compatibility" on page 35-8.

Not all telecommunications carriers allow you to assign the same phone number to multiple paths. When you contact your carrier to acquire support services, verify that they support this feature. You must also specify that you will be using subaddresses.

When setting up certain parameters such as -PORT DialNoList, you may need to specify a dial or phone number string consisting of a phone number and, if applicable, a subaddress. If you specify a subaddress, you must separate the phone number and the subaddress with a semicolon (;). The phone number can consist of a maximum of 30 characters, while the subaddress can consist of a maximum of 20 characters.

When specifying a phone number, valid characters include the digits 0 through 9, an asterisk (*), and the pound sign (#). Because the software ignores all other characters to the left of the semicolon that separates the phone number and subaddress, you can also specify special characters such as parentheses and dashes to distinguish the different elements that compose a phone number, and text characters to embed descriptive text in the string.

When you specify a phone number, each character entered (whether the software considers it valid or invalid) counts toward the maximum allowable number of characters.

When you specify a subaddress, valid characters include all ASCII or IA5 characters.

The following string is an example of a dial or phone number string that includes a subaddress:

Los Angeles Office 1-213-555-1000;200

In this dial or phone number string, the phone number consists of long distance dial prefix 1 (assuming that the bridge/router being configured is located in Santa Clara), phone number 213-555-1000, and the subaddress 200. The descriptive text to the left of the semicolon indicates that the phone and subaddress numbers are for the Los Angeles office.

36 CONFIGURING THE NETBUILDER II TO USE A WAN EXTENDER

This chapter describes how to configure the NETBuilder II bridge/router to use one or more WAN Extender systems to interconnect large numbers of remote LANs to a central site using channelized leased circuit services and switched circuit services.

This chapter also describes how to use the commands and parameters that are used for WAN Extender.

For conceptual information, refer to "How the WAN Extender Works" on page 36-30 of this chapter.

This chapter should be used in connection with the following guides:

- *WAN Extender 2T/2E Installation Guide*
- *WAN Extender 2T/2E Manager User's Guide*
- *Reference for NETBuilder Family Software*

Circuit Services Supported

The WAN Extender provides virtual paths to be used by ports for leased and switched circuit services. The NETBuilder II bridge/router supports up to 75 virtual paths.

The following types of services are supported and can be used in configuring a NETBuilder II bridge/router to use a WAN Extender:

- Leased (permanent) circuit-based services:
  - Channelized T1
  - Channelized E1
- Switched circuit-based services:
  - Switched 56 (available with WAN Extender model 2T only)
  - ISDN Primary Rate Interface (PRI) (available with WAN Extender models 2T and 2E)

Configuring WAN Extender and NETBuilder II for Remote Connections

To enable a NETBuilder II bridge/router to use a WAN Extender to interconnect remote sites with a central site you need to configure the WAN Extender and the NETBuilder II bridge/router.

Only 3Com NETBuilder bridge/routers at the remote sites can be interconnected with a WAN Extender to a NETBuilder II bridge/router at a central site. The WAN Extender uses SysCallerID to identify remote callers and only 3Com bridge/routers can be identified with SysCallerID.

You configure the WAN Extender according to the procedures in the *WAN Extender 2T/2E Manager User's Guide*. The configuration procedures in this chapter include only the WAN Extender configuration steps necessary to perform the NETBuilder II bridge/router configuration.

This chapter describes how to configure leased circuit lines. The sample configuration describes how to configure a channelized T1 leased line. If you are configuring E1 leased lines, enter E1 instead of T1 and WAN Extender 2E instead of WAN Extender 2T in the configuration procedure. This chapter also describes how to configure ISDN PRI switched-circuit lines, and briefly describes how to configure switched 56 lines, which is a similar configuration.

Refer to "Remote Connection Configuration Considerations" on page 36-12 for additional information to help you with the configuration procedures.

The configuration procedures in this chapter are basic guidelines to be used together with the information in the guides listed at the beginning of this chapter on page 36-1 to tailor a configuration to suit your specific needs.

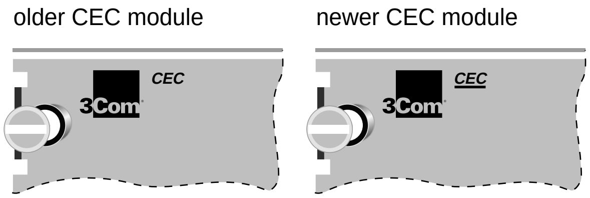
Requirements

To configure a NETBuilder II bridge/router to use a WAN Extender, the following hardware and software requirements must be met:

- WAN Extender software, any version up to 1.15, but version 1.15 is recommended
- NETBuilder II software version 9.1 or later
- An installed HSS RS-449 module

  The HSS RS-449 *3-Port* module is not supported.
- An installed Communications Engine Card (CEC) module with 20 MB or a Dual Processor Engine (DPE) module, which support up to 75 virtual ports.

  Only newer CEC modules support 20 MB (12 MB plus the 8 MB memory expansion board). The newer CEC module has connectors on the right half of the connector-and-LED panel as well as a line under the word "CEC," as shown in Figure 36-1:



\

Figure 36-1 Older and Newer CEC Modules

Refer to the *NETBuilder II Communications Engine Card (CEC) Module Installation Guide* and the *NETBuilder II CEC Memory Expansion Installation Guide* for more information on the CEC module. Refer to the *Installing the NETBuilder II Dual Processor Engine (DPE) Module* for more information on the DPE module.

Interconnecting Leased DS0s to Channelized T1

This configuration example shows how to achieve point-to-point interconnection of remote sites to a central site, using a leased DS0 (64 Kbps) circuit at each remote site and one or more channelized T1 (1536 Kbps) circuits at the central site.

Although this example describes remote sites using single DS0s, remote sites can use a bundle of multiple DS0s. The WAN Extender can accommodate bundles of various numbers of DS0s running the channelized profile configuration, which results in a virtual path being automatically created on the NETBuilder II bridge/router at the central site with the appropriate baud rate for a bundle of DS0s. For the bundles of DS0s to operate properly, a network services provider may be required to perform the proper mapping of DS0s through the network.

Figure 36-2 shows two channelized T1 circuits coming into a WAN Extender 2T, which is connected to a NETBuilder II bridge/router at the central site.

The bridge/routers at the remote sites are labelled as SuperStack II bridge/routers, although they could be any NETBuilder platform that supports a serial interface that can connect to a 64 Kbps (or larger) leased circuit with a suitable Channel Service Unit/Data Service Unit (CSU/DSU). NETBuilder II bridge/routers and SuperStack II bridge/routers all qualify as a remote site.

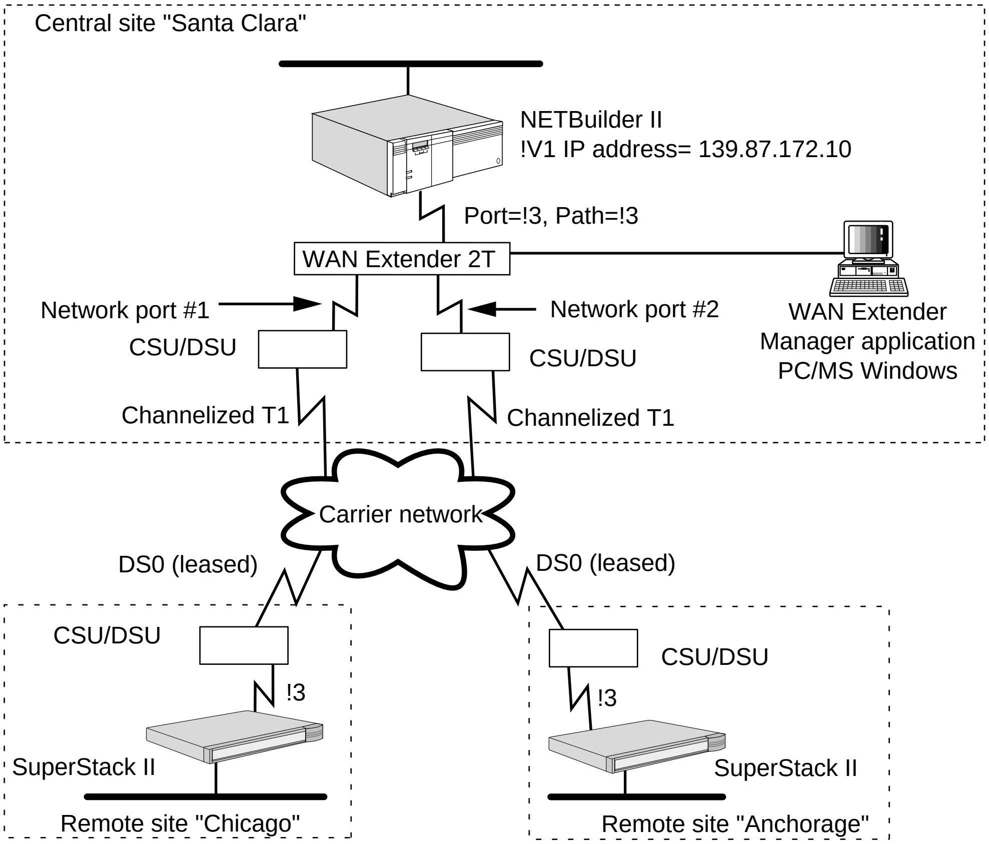


Figure 36-2 Interconnecting Leased DS0s to Channelized T1

Configuring the WAN Extender

To configure the WAN Extender, use the *WAN Extender 2T/2E Manager User's Guide* and follow these steps with the WAN Extender Manager application:

For E1 configurations, substitute "E1" for "T1" and "WAN Extender 2E" for "WAN Extender 2T" in the steps that follow.

1 At the System Parameters window on the PC connected to your WAN Extender:

a Select WAN Extender 2T for System Type.

b Select Channelized T1 for Call Control for one or both network ports. If you are using only one port, it must be port number 1, and port number 2 must be set to Unused.

c Set up the remaining parameters based on the type of existing network connections and according to the instructions in the *WAN Extender 2T/2E Manager User's Guide*.

2 From the Remote Site Profiles window, open and define a profile for each remote site to describe how the underlying connectivity to that site is achieved. In the Profile screen:

a Select Channelized T1 for Profile Type.

b Select the appropriate Outgoing Network Port.

c Click the check box for the channel that has been assigned for connectivity to the remote site.

d Complete your configuration by selecting a value for the Outgoing Call Circuit Type, deciding whether to enable or disable the Inverted HDLC on Selected Channels, and selecting values for the rest of the configuration items required for the profile creation as described in the *WAN Extender 2T/2E Manager User's Guide*.

3 Download the completed configuration file to the WAN Extender.

4 Reboot or reset the WAN Extender.

Configuring the NETBuilder II Bridge/Router

To configure the NETBuilder II bridge/router to use the WAN Extender, follow these steps using the terminal connected to the NETBuilder II bridge/router:

1 Set the owner of the NETBuilder II bridge/router port, which corresponds to the NETBuilder II bridge/router physical path to which the WAN Extender is connected, to WAN Extender.

For example, enter:

```
SETDefault !3 -PORT Owner = WanExtender
```

2 Set the baud rate on the WAN Extender-to-NETBuilder II bridge/router path to 4096 by entering:

```
SETDefault !3 -PATH Baud = 4096
```

The 4096 value is the only accepted baud rate value for the NETBuilder II bridge/router configuration to use the WAN Extender.

3 Set the clock for the physical path to External by entering:

```
SETDefault !3 -PATH CLock = External
```

4 Enable the physical path that corresponds to the serial connection by entering:

```
SETDefault !3 -PATH CONTrol = Enable
```

The WAN Extender and the NETBuilder II system will now synchronize using the 3Com proprietary WNI Protocol. After the synchronization, the NETBuilder II system path is in an UP state.

5 Create NETBuilder II bridge/router virtual ports, one for each remote site, to represent the logical attachment between the central site and the remote site by entering:

```
ADD !V1 -PORT VirtualPort SCID “Chicago”
ADD !V2 -PORT VirtualPort SCID “Anchorage”
```

The string within quotes uniquely identifies the remote site. This string must correspond to the configured -SYS SysCallerID parameter string of the remote site NETBuilder II bridge/router. PPP packets received from the remote site with those strings will bind the ports to the appropriate virtual paths that represent the data channels through the WAN Extender.

6 You must specify that the virtual port represents a permanent connection (as opposed to one that requires a dial-up) by entering:

```
SETDefault !V1 -PORT DialInitState = NoDialOut
SETDefault !V2 -PORT DialInitState = NoDialOut
```

7 Specify the size of the largest packet that is transmitted or received between the NETBuilder II bridge/router and the SuperStack II bridge/router at the remote site.

If packets up to the maximum Ethernet size (1518) are to be bridged or routed across the connection, set the MaxRcvUnit parameter for a virtual port to 1518 by entering:

```
SETDefault !V1 -PPP MaxRcvUnit = 1518
SETDefault !V2 -PPP MaxRcvUnit = 1518
```

If packets up to the maximum Fiber Distributed Data Interface (FDDI) or token ring size (4500) are to be bridged or routed across the connection, set the MaxRcvUnit parameter for a virtual port to the maximum value of 4500 by entering:

```
SETDefault !V1 -PPP MaxRcvUnit = 4500
SETDefault !V2 -PPP MaxRcvUnit = 4500
```

For all NETBuilder II bridge/router virtual ports using WAN Extender virtual paths, the -PPP MaxRcvUnit parameter must be set to the appropriate maximum size value (4500 in this example). Failure to do so may result in the loss of network connections, and you may need to reboot the WAN Extender to recover connections.

When configuring the WAN Extender, the Maximum Data Buffer Size field must also be set properly to avoid losing network connections and to avoid rebooting the WAN Extender to recover connections. The Maximum Data Buffer Size field is configured on the System Parameters window of the WAN Extender Manager program. Refer to the *WAN Extender 2T/2E Manager User’s Guide* for configuration instructions.

If all the following three conditions exist, set the Maximum Data Buffer Size field to a value of 1750:

- Any WAN Extender virtual port on the NETBuilder II bridge/router is configured with PerPacket compression.
- The remote site is using a NETBuilder product.
- Only Ethernet packets are being compressed and bridged or routed through the WAN Extender (although the maximum Ethernet packet size is typically 1518 bytes, additional buffer space is required).

Configuring Other Protocols

After you have configured the NETBuilder II bridge/router to use the WAN Extender, you can configure bridging and other protocols (such as IP, IPX, and so on) on your NETBuilder II bridge/router for the virtual ports.

For example, if you are using IP over !V1, you can enter:

```
ADD !V1 -IP NETaddr = 139.87.172.10
SETDefault !V1 -IP CONTrol = ROute
SETDefault !V1 -RIPIP CONTrol = (TAlk, Listen)
```

Verifying the Configuration

To verify that the WAN Extender and NETBuilder II bridge/router are operating as configured, follow these steps on the NETBuilder II bridge/router:

1 Check that the paths established through the WAN Extender are in the UP state by entering:

```
SHow -PATH CONFiguration
```

Because a WAN Extender virtual path does not bind to a port until a connection is established, some of the path parameters may not show on the Current Path Parameters display. Virtual paths used for leased channelized T1 or E1 are bound to a port when the NETBuilder II bridge/router and the WAN Extender synchronize with each other and the PPP negotiation is completed.

2 Check that all the ports that use WAN Extender virtual paths are in the UP state by entering:

```
SHow -PORT CONFiguration
```

3 Check the status of the PPP Protocol on each of the WAN Extender paths and ports by entering:

```
SHow -PPP STATUS
```

The Link Control Protocol (LCP) and all configured Network Control Protocols (NCPs) should be in the Open state under normal operation.

4 Display the configuration information for port 3 by entering:

```
SHow !3 -WE Configuration
```

Refer to "Sample Configuration Verification Displays" on page 36-20 for sample displays generated by this command. Refer to Chapter 64 in *Reference for NETBuilder Family Software* for information about the -WE CONFiguration parameter.

5 Display the connection and data packet statistics between NETBuilder II bridge/router and the WAN Extender for all the ports by entering:

```
SHow -WE DevSTATistics
```

Refer to “Sample Configuration Verification Displays” on page 36-20 for sample displays generated by this command. Refer to Chapter 64 in *Reference for NETBuilder Family Software* for information about the -WE DevSTATistics parameter.

6 Retrieve detailed information about incoming and outgoing calls for profile number 3 of the NETBuilder II bridge/router port 3, which is connected to the WAN Extender by entering:

```
SHow !3 -WE ProFile 3 DETail
```

Refer to “Sample Configuration Verification Displays” on page 36-20 for sample displays generated by this command. Refer to Chapter 64 in *Reference for NETBuilder Family Software* for information about the -WE ProFile parameter.

If you find problems with the configuration after verification, refer to “Troubleshooting” on page 36-22 in this chapter.

Interconnecting ISDN BRI Circuits to ISDN PRI

This configuration example (see Figure 36-3) shows how to achieve point-to-point interconnection of two remote SuperStack II bridge/routers (through a WAN Extender) to a central site NETBuilder II bridge/router using ISDN Basic Rate Interface (BRI) circuits at the remote sites and ISDN PRI circuits at the central site. There are two channels running through the same port at each of the remote sites. Northern Telecom DMS-100 is used as the example ISDN switch-type service carrier.

The Multilink Protocol (MLP), which provides load balancing, is enabled in this example. With load balancing, data being sent is split over the available parallel PPP serial lines, while providing the proper sequencing at the receiving end so that the data is received in order. For more information about the Multilink Protocol, refer to Chapter 32.

Although the routers at the remote site are labelled as SuperStack II, they can be any NETBuilder II bridge/router or a SuperStack II bridge/router that supports:

- A serial interface that can connect to an ISDN BRI terminal adapter (TA).
- The ability to communicate with the TA for call establishment and teardown.
- Any NETBuilder platform with an integrated ISDN BRI TA.

Part of the following configuration instructions requires setting up the dial-up procedure, which includes selecting dial-up and remote-site identification options. Refer to “Remote Connection Configuration Considerations” on page 36-12 for information about these options. For more information about configuring and using dial-up, refer to Chapter 37 in this guide.

The software version 9.1 has introduced two switch-type options for the PORT DialNoList parameter, WE and WEH0. WE indicates 64 kbps circuits and WEH0 indicates 384 kbps circuits. If you have a NETBuilder software version earlier than 9.1 and have configured a profile for a 384 kbps circuit, you must reconfigure the port using WEH0 as the port type option for the PORT DialNoList parameter.

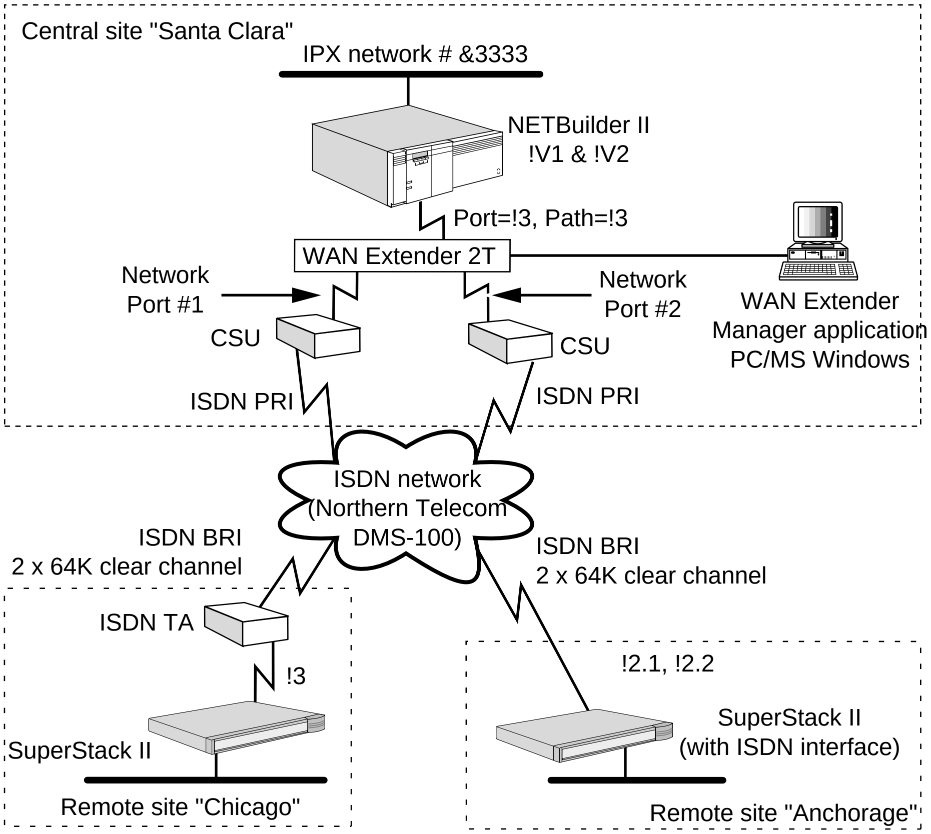


Figure 36-3 Interconnecting ISDN BRI Circuits to ISDN PRI

Configuring the WAN Extender

To configure the WAN Extender, use the *WAN Extender 2T/2E Manager User's Guide* and follow these steps:

1 At the System Parameters window on the PC connected to your WAN Extender:

  a Select WAN Extender 2T for System Type.

  b Select ISDN for Call Control for both network ports.

  c Set up the remaining parameters based on the type of existing network connections. Refer to the *WAN Extender 2E/2T Manager User's Guide* for more information.

2 In the Port Parameters screen for each port:

  a Select the switch specified by your network service provider from the Switch Type menu list.

  b Select the switch variant specified by your network service provider from the Variant menu list.

  c Select the call type specified by your network service provider from the Network Call Types Allowed field.

  d Enable the B channels that are available for use by the WAN Extender in the Enabled Network B Channels check boxes.

e Set up the remaining parameters based on the type of existing network connections and other configuration values entered. Refer to the *WAN Extender 2E/2T Manager User's Guide* for more information.

3 Using the Remote Site Profiles window, create a profile for each remote site to describe how the connection to that site is achieved.

In the Profile screen:

a Select ISDN for Profile Type.

b Select either Network Port 1 or Network Port 2 for Outgoing Call Network Port.

Both WAN Extender ports are connected to the same ISDN network in this example. Either port can be used to originate calls to any remote site or receive calls from any remote site.

If you are connected to two separate ISDN networks, you must assign each remote profile to a specific port.

c Select a circuit type supported by your ISDN configuration for Outgoing Call Circuit Type.

d Enter the remote site telephone number in the Outgoing Called Number field.

e Enter the telephone number used for calling-party number matching when the WAN Extender receives an incoming call for Incoming Calling Number.

f Select the proper Number Type and Numbering Plan.

g Set up the remaining parameters based on the type of existing network connections and other configuration values entered. Refer to the *WAN Extender 2E/2T Manager User's Guide* for more information.

4 Make a note of the Profile ID that corresponds to each remote site.

This information is required to correctly configure the NETBuilder II bridge/router to establish the end-to-end connection. For this example, assume that the profile corresponding to the remote site "Chicago" was numbered 1 and 2. The profile corresponding to the remote site "Anchorage" was numbered 3 and 4.

5 Download the completed configuration file to the WAN Extender.

6 Reboot or reset the WAN Extender.

Refer to the *WAN Extender 2T/2E Manager User's Guid*e for more details on this procedure as well as other configuration options you may want to use.

Configuring the NETBuilder II Bridge/Router

To configure the NETBuilder II bridge/router, follow these steps:

1 Set the owner of the port that corresponds to the

WAN Extender-to-NETBuilder II bridge/router connection to WAN Extender by entering:

```
SETDefault !3 -PORT OWner = WanExtender
```

2 Set the baud rate on the WAN Extender-to-NETBuilder II bridge/router connection to 4096 by entering:

```
SETDefault !3 -PATH BAud = 4096
```

The 4096 value is the only accepted value for the baud rate when configuring the NETBuilder II bridge/router to use the WAN Extender.

3 If you have configured the WAN Extender for caller identification through the PPP system identification data, you must allow the central site to identify itself to the remote sites by entering:

```
SETDefault -SYS SysCallerID = “Santa Clara”
```

4 Enable the path that corresponds to the serial connection by entering:

```
SETDefault !3 -PATH CONTrol = Enable
```

5 Create NETBuilder II bridge/router virtual ports, one for each remote site, to represent the logical attachment between the central site and the remote site by entering:

```
ADD !V1 -PORT VirtualPort SCID “Chicago”
ADD !V2 -PORT VirtualPort SCID “Anchorage”
```

The string within quotes uniquely identifies the remote site. This string corresponds to the string configured on the remote router with -SYS SysCallerID parameter. This identification is used during PPP link establishment to map the incoming call to the virtual port associated with the remote site.

6 Specify the mapping between the virtual port that represents the logical attachment to the remote site and the WAN Extender profile that describes the underlying physical connection to the remote site by entering:

```
ADD !V1 -PORT DialNoList “3 1"
ADD !V1 -PORT DialNoList “3 2”
ADD !V2 -PORT DialNoList “3 3”
ADD !V2 -PORT DialNoList “3 4”
```

In these commands, the first number within quotes is the number of the physical port to which the WAN Extender is connected. The second number is the profile ID on that WAN Extender that is used to make a call. V1 and V2 have two paths assigned to them. Two paths come up and become available when a call is put through the virtual port. This enables the Multilink Protocol (MLP) to use load balancing and split the data over the two paths if it becomes necessary.

The NETBuilder II bridge/router virtual paths go to available and UP states when the calls are completed, and if both sites are configured correctly.

7 Set the normal bandwidth to 128 kbps for virtual ports V1 and V2 by entering:

```
SETDefault !V1 -PORT NORMalBandwidth = 128
SETDefault !V2 -PORT NORMalBandwidth = 128
```

This setting makes the Bandwidth Manager application bring up both paths for the virtual port to satisfy the port’s normal bandwidth requirement of 128 kbps.

8 Enable MLP for V1 and V2 by entering:

```
SETDefault !V1 -PPP MlpCONTrol = Enable
SETDefault !V2 -PPP MlpCONTrol = Enable
SETDefault !V1 -PORT CONTrol = Enable
SETDefault !V2 -PORT CONTrol = Enable
```

9 Specify the dial-up initiation condition for each virtual port by entering:

```
SETDefault !V1 -PORT DialInitState = ManualDial
SETDefault !V2 -PORT DialInitState = ManualDial
```

10 Specify the size of the largest packet that will be transmitted or received between this NETBuilder II bridge/router and the SuperStack II bridge/router at the remote site.

If packets up to the maximum Ethernet size (1518) are to be bridged or routed across the connection, set the MaxRcvUnit parameter for a virtual port to 1518 by entering:

```
SETDefault !V1 -PPP MaxRcvUnit = 1518
SETDefault !V2 -PPP MaxRcvUnit = 1518
```

If packets up to the maximum FDDI or token ring size (4500) are to be bridged or routed across the connection, set the MaxRcvUnit parameter for a virtual port to the maximum value of 4500 by entering:

```
SETDefault !V1 -PPP MaxRcvUnit = 4500
SETDefault !V2 -PPP MaxRcvUnit = 4500
```

The -PPP MaxRcvUnit parameter must be set to the appropriate maximum size value (4500 in this example) for all NETBuilder II bridge/router virtual ports using WAN Extender virtual paths. Failure to do so may result in the loss of network connections, and you may need to reboot the WAN Extender to recover connections.

When configuring the WAN Extender, the WAN Extender Maximum Data Buffer Size field must also be set properly to avoid losing network connections and to avoid rebooting the WAN Extender to recover connections. The WAN Extender Maximum Data Buffer Size field is configured on the System Parameters window of the WAN Extender Manager program. Refer to the *WAN Extender 2T/2E Manager User's Guide* for configuration instructions.

If all the following three conditions exist, set the Maximum Data Buffer Size field to a value of 1750:

- Any WAN Extender virtual port on the NETBuilder II bridge/router is configured with PerPacket compression
- The remote site is using a NETBuilder product
- Only Ethernet packets are being compressed and bridged or routed through the WAN Extender (although the maximum Ethernet packet size is typically 1518 bytes, additional buffer space is required)

Configuring Other Protocols

After you have configured the NETBuilder II bridge/router to use the WAN Extender, configure bridging and other protocols (such as IP, IPX, and so on) on your NETBuilder II bridge/router for the virtual ports.

For example, if you are using IPX over !V1, you enter:

```
SETDefault !V1 -IPX NETnumber = &123
SETDefault -IPX InternalNET = &3333
SETDefault !V1 -NRIP CONTrol = NoPEriodic
SETDefault !V1 -SAP CONTrol= NoPEriodic
SETDefault !V1 -IPX CONTrol = IpxWan
```

Verifying the Configuration

To verify that the configuration is correct, and that the WAN Extender and NETBuilder II bridge/router are operating as configured, follow these steps on the NETBuilder II bridge/router:

1 Try to establish a link with the remote site by entering:

```
DIal !V1
DIal !V2
```

2 Check that all paths established through the WAN Extender are in the UP state by entering:

```
SHow -PATH CONFiguration
```

Because a WAN Extender virtual path does not bind to a port until a connection is established, some of the path parameters may not show on the Current Path Parameters display. Virtual paths used for dial-up connections do not bind with a port until an outgoing call is completed or an incoming call is accepted.

3 Check that all virtual ports defined through the WAN Extender are in the UP state by entering:

```
SHow -PORT CONFiguration
```

4 Check the status of the PPP protocol on each of the ports established through the WAN Extender by entering (the LCP and all configured NCPs should be in the Open state):

```
SHow -PPP STATus
```

If you find problems with the configuration after verification, refer to "Troubleshooting" on page 36-22 of this chapter.

Configuring Switched 56 Circuits

The WAN Extender 2T can be configured to support remote sites that are connected through switched 56 circuits. Switched 56 circuits share many of the dial-up characteristics described in "Interconnecting ISDN BRI Circuits to ISDN PRI" on page 36-7.

The WAN Extender supports called ID and PPP system identification, but does not support incoming caller identification based on caller ID.

To configure the WAN Extender to support switched 56, follow these steps:

1 On the System Parameters window, select Switched 56 for call Control.

2 From the Remote Site Profiles window, select a profile for a remote site connected through a switched 56 circuit.

3 In the Profile window:

a Select Switched 56 for Profile Type.

b Select Either (Port 1 preferred) or Either (Port 2 preferred) for Outgoing Call Network Port.

c Enter the remote site's telephone number in the Outgoing Called Number field.

4 Repeat steps 2 and 3 for each remote site connected with a switched 56 circuit.

Remote Connection Configuration Considerations

This section describes information you may need to consider before configuring the NETBuilder II bridge/router to use ISDN or switch 56 switch-circuit services through the WAN Extender to link remote sites with a central site.

Dial-Up Options

ISDN circuits are switched circuits and require the execution of end-to-end call set-up procedures before a link between a remote site and the central site can be established. The NETBuilder software offers several alternatives for determining when to establish the link. All of the options described here are available for and apply to links that are established through the WAN Extender. For more details about the dial-up alternatives, refer to Chapter 37.

Operator-Initiated Dialing (Manual Dial)

In this mode, you can use the DIal command from the NETBuilder II bridge/router to initiate the link between the remote site and the central site. The HangUp command terminates or tears down the link.

Scheduled Dial

This mode is a variation of the operator-initiated (manual) dialing. Instead of entering the DIal and HangUp commands, you can create NETBuilder macros containing these commands, and use the SCHeduler Service to specify when (day of week, time of day) these macros should be executed.

Auto Dial

This mode is another variation of the operator-initiated (manual) dialing. Instead of entering the DIal command to establish the link, the software automatically makes a call to the remote site at system initialization whenever the virtual port is enabled or whenever the WAN Extender-to-NETBuilder II bridge/router port or path is enabled.

Dial-on-Demand

In this mode, the NETBuilder software automatically establishes the link when any user data needs to be forwarded between the remote and central sites, and disconnects the link when there is no outgoing traffic.

Because ISDN circuits can be established or disconnected on demand, configurations can be created where the number of remote sites is greater than the total number of ISDN B channels at the central site. This is called *oversubscription*.

Oversubscription is useful for internetworking a large number of remote sites to a central site if the number of remote sites that need to be simultaneously connected to and communicating with the central site does not exceed the total number of available ISDN B channels at the central site.

The total number of remote sites cannot exceed the maximum number of central site virtual ports the NETBuilder II software can support. Each remote site attachment, whether active or not, must be represented as a unique virtual port at the central site.

Remote Site Identification Options

The WAN Extender or the NETBuilder II bridge/router must identify the originator of an incoming call so that the call can be mapped to the NETBuilder II bridge/router port associated with the remote site. For ISDN-based WAN Extender channels, there are several caller identification options available (for more information about remote site identification options, refer to the *WAN Extender 2T/2E Manager User's Guide*).

ISDN Caller ID on the WAN Extender

This is the most efficient and cost-effective way to map incoming calls to NETBuilder II bridge/router ports, because the call does not have to be completed (and therefore no charges incurred) before validating the call or the caller. The PPP system identification method requires the call to be completed before the caller can be validated.

In this method, the WAN Extender attempts to match the ISDN caller ID to the Incoming Calling Number fields in the ISDN remote site profiles, and passes the incoming call (referencing the profile ID) to the NETBuilder II bridge/router.

The WAN Extender can be configured to reject any incoming call if no matching ISDN caller ID profile (or called ID profile) can be found on the WAN Extender. This capability is called *call filtering*.

Make sure incoming caller ID is supported across all network providers between sites before attempting this method of remote site identification.

ISDN Called ID on the WAN Extender

This caller identification method can be used if there are multiple ISDN numbers (one for each remote site) subscribed at the central site, for example, using a direct inward dialing (DID) numbering plan. The WAN Extender attempts to match the ISDN called ID to the Incoming Called Number fields in the ISDN remote site profiles, and passes the incoming call (referencing the profile ID) to the NETBuilder II bridge/router. This form of caller identification occurs after checking for an ISDN caller ID match.

PPP System ID Data on the NETBuilder II Bridge/Router

With this caller identification method (which is 3Com NETBuilder proprietary), the WAN Extender relays the call to the NETBuilder II bridge/router if no ISDN caller ID-based or ISDN called ID-based mapping can be done. The NETBuilder II bridge/router accepts the call, establishes the link using PPP, and waits for PPP system identification data to arrive. This data is then used to associate a virtual port with the caller. This method uses the unique string specified in the creation of a WAN Extender virtual port. The string that is received from the remote site is the value of the -SYS SysCallerID parameter of a remote NETBuilder bridge/router.

Call filtering must be disabled on the WAN Extender for this option. When call filtering is disabled, the WAN Extender passes through any calls from remote sites to the NETBuilder II bridge/router, whether a matching profile was found or not.

The NETBuilder systems at the remote sites must also use the PPP system identification data to identify the central site as the caller when they receive incoming calls.

Customizing the Configurations

This section describes some WAN Extender configuration alternatives so that you can customize the configuration to your needs. These configurations are done through the Windows-based WAN Extender Manager application running on your PC connected to the WAN Extender console port.

ISDN H0 Support (WAN Extender 2T Only)

The WAN Extender 2T can be configured to establish H zero (H0) (384 kbps) ISDN PRI calls if you have purchased that capability from your ISDN service provider.

To configure for H0 calls, follow the steps described in “Interconnecting ISDN BRI Circuits to ISDN PRI” on page 36-7, and then follow these steps:

1. In the Port Parameters window for each network port capable of accepting or originating H0 calls, select 384Kbps for Network Call Types Allowed.
2. In the Remote Site Profiles window, select a profile that represents a remote site capable of accepting an H0 call.
3. In the Profile window, select 384Kbps for Outgoing Call Circuit Type.
4. Repeat steps 2 and 3 for each profile that represents a remote site capable of accepting an H0 call.
5. On the NETBuilder II bridge/router terminal console, set DialPathLimit to H0 for each WAN Extender port where H0 calls can be initiated or received.
6. Download the customized configuration to the WAN Extender, and then reset the WAN Extender.

Call Filtering

Call filtering limits caller identification to caller ID or called ID methods. The WAN Extender rejects incoming calls whose caller ID or called ID (based on the Incoming Calling Number field in the ISDN remote site profiles) does not match the WAN Extender profiles. This is the most efficient and cost-effective way to map incoming calls to NETBuilder II bridge/router virtual ports, because the call does not have to be completed (and no charges occur) before validating the call or the caller. The PPP system identification method requires the call to be completed first before the caller can be validated.

Make sure incoming caller ID is supported across all network providers between sites before attempting this method of remote site identification. Refer to “ISDN Caller ID on the WAN Extender” on page 36-13 for more information.

To enable call filtering on the WAN Extender for ISDN and switched 56 circuits, check the Call Filtering Enabled check box on the corresponding Port Parameter window.

Channel Bundling

The WAN Extender permits the bundling of channels or slots on a channelized T1 or E1 circuit to connect to a site that has a fractional T1or E1 circuit provisioned, or is connected through a WAN Extender with a similar configuration.

To configure the WAN Extender for channel bundling, select more than one channel in the Channelized Profile window of WAN Extender Manager. Refer to the *WAN Extender 2T/2E Manager User’s Guide* for more information on this feature.

NETBuilder II Configuration Commands and Parameters

This section provides a brief description of the commands, Path Service parameters, and Port Service parameters that are used in the configuration of the NETBuilder II bridge/router for the WAN Extender.

For a description of all the WE Service parameters, refer to Chapter 64 in *Reference for NETBuilder Family Software.*

Commands

This section describes the command used in the configuration with a WAN Extender. For information on all commands, which you may use with the WAN Extender, refer to Chapter 1 in *Reference for NETBuilder Family Software.*

DLTest

If the local link cable, local port, or NETBuilder II bridge/router serial interface adapter are not functioning correctly, the NETBuilder II bridge/router physical path connected to the WAN Extender will never transition to an UP state (visible by using the SHow -PATH CONFiguration command). By using the appropriate loopback connector on the NETBuilder II bridge/router serial interface adapter, and using the DLTest command as described in Chapter 1 in *Reference for NETBuilder Family Software*, you can determine if the interface adapter is functioning correctly.

The WAN Extender can be placed into the loopback mode to use the DLTest command by placing the WAN Extender into loopback, by changing the baud rate from 4096 to 2048, and by changing the -PATH Clock parameter setting from External to TestMode. Placing the WAN Extender into the loopback mode allows you to use the DLTest command loopback test to verify the local link cable or WAN Extender limitations on passing packets.

PATH Service Parameters

This section describes the Path Service parameters used in the configuration with a WAN Extender. For a detailed description of the Path Service parameters, refer to Chapter 42 in *Reference for NETBuilder Family Software*.

Baud

This parameter sets the correct baud rate for the local link between the NETBuilder II bridge/router and the WAN Extender. In this software release, the baud must be set to 4096. This parameter does not apply to WAN Extender virtual paths because the baud rate for virtual paths is supplied to the NETBuilder II bridge/router by the WAN Extender.

CLock

This parameter determines how a bridge/router using serial interfaces derives its transmit clock . The WAN Extender provides clocking for the HSS RS-449 module in the NETBuilder II bridge/router, so this parameter must be set to External. This parameter does not apply to WAN Extender virtual paths.

CONFiguration

This parameter displays the configuration and the current state of all paths, including WAN Extender-based virtual paths. The following is a sample display:

```
                 ----------------------Current Path Parameters-----------------------
Path    Name      Port   Ctrl   State   T1Mode   Baud (kbps)   Conn     Clock   Line
2       Path_2    2      Ena    Up      -        10000         -        -       -
3       Path_3    3      Ena    Up      -        4096          RS449    Ext     Leased
6       Path_6    6      Ena    Dwn     -        64            RS449    Ext     Leased
8       Path_8    8      Ena    Dwn     -        64            V35      Ext     Leased
V2      Path_V2   V2     Ena    Up      -        64            WE       Ext     Leased
V4      Path_V4   V4     Ena    Up      -        64            WE       Ext     Dialup
V5      Path_V5   -      Ena    Down    -        64            -        Ext     Dialup
```

The WAN Extender virtual path does not bind to a port until the connection is established. If the WAN Extender virtual path has not bound to a port, the Conn column on the Current Path Parameters display shows a hyphen instead of a value.

For WAN Extender virtual paths used as dial-up lines, a connection is established when an outgoing call is completed or when an incoming call is accepted. For channelized lines, the connection is established when the NETBuilder II bridge/router synchronizes with the WAN Extender and PPP negotiation is completed.

CONNector

This parameter specifies the connector type for a serial interface. When you change this parameter setting, you need to re-enable the corresponding path for the new parameter value to take effect for the path to which the WAN Extender is connected. This parameter must be set to RS449. This parameter does not apply to WAN Extender virtual paths.

CONTrol

This parameter enables or disables a path on the bridge/router. By disabling and enabling the path, all the values associated with the CONTrol parameter take effect.The only options that apply to WAN Extender-based virtual paths, or to the path the WAN Extender is connected to, are Enable and Disable. For the path the WAN Extender is connected to, Disable causes all virtual paths established through that WAN Extender to go down and become unavailable.

Enable causes the WAN Extender and the NETBuilder II bridge/router to go through a resynchronization procedure on the local link. The NETBuilder II bridge/router first attempts to retrieve WAN Extender system information and global and network port level configuration settings.

DialCONTrol

This parameter is a bit-mapped control parameter, which sets the path attributes for the dial-up paths. When configuring a NETBuilder II bridge/router for a WAN Extender, the WAN Extender virtual paths available for dial-up paths are set automatically to the default values for the DialCONTrol parameter, except that the virtual paths are automatically set to DYNamic and not STAtic.

DialPool

This parameter displays the dial pool status and configuration. This display shows all paths in the dial pool, all dynamic paths, both physical and virtual, the last time the path was used, the time when the current path became active, the external device type, and which ports have reserved the dial paths through the -PORT PathPreference parameter.

Because WAN Extender virtual paths do not bind to a port until a connection is established, virtual dial paths will not be reserved for specific ports through the -PORT PathPreference parameter. When you enter the SHow -PATH DialPool command, the virtual paths provided by WAN Extender to the dial-up pool are displayed, but the reservation of WAN Extender virtual paths to a particular port are not displayed.

For WAN Extender virtual paths used as dial-up lines, a connection is established when an outgoing call is completed or when an incoming call is accepted.

ExDevType

This parameter specifies and displays the external device type attached to a DTE connector. The HSS modules installed in a NETBuilder II bridge/router have an RS-232 or RS-449 DTE connector type. This parameter is used only with the dial-up path selection algorithm for matching destination phone numbers with dynamic dial ports. For NETBuilder II bridge/routers with a WAN Extender, this parameter is set automatically to WE or WEH0. This setting can be viewed, but not changed with the ExDevType parameter.

LineType

This parameter sets the type of line being used on a wide area interface. The options are Leased or Dial-up. For the physical path to which the WAN Extender is connected, this parameter must be set to Leased. The LineType for virtual paths is set automatically by the WAN Extender device driver to Dial-up for a dial-up channel, such as an ISDN or Switched 56 channel, and to Leased for a channelized virtual path. The LineType settings for virtual paths can be viewed but not changed with this parameter.

PORT Service Parameters

This section describes the Port Service parameters used in the configuration with the WAN Extender. For a detailed description of the Port Service parameters, refer to Chapter 43 in *Reference for NETBuilder Family Software*.

COMPressType

This parameter determines the compression type for virtual ports. The only type of compression available for virtual ports that are based on the WAN Extender is per-packet compression. The per-packet link-level option looks for repetitive patterns within a packet and replaces them with shorter length codes.

For more details on this parameter, refer to Chapter 39 in this guide and to Chapter 43 in *Reference for NETBuilder Family Software*.

CONFiguration

This parameter displays the configuration associated with WAN Extender-based virtual ports. For these ports, the Owner column contains PPP and the Paths column contains the SCID “SysCallerID” that was entered when the virtual port was added. For the NETBuilder II bridge/router physical port to which the WAN Extender is connected, the Owner column contains WE (WAN Extender), and the Paths column contains the number of the path to which the WAN Extender is connected.

```
          ---------------------Current Port Parameters----------------------
Port      Name       Ctrl      State     Owner     Paths
1         Port_1     Ena       Up        Eth       1
2         Port_2     Ena       Up        Eth       2
3         Port_3     Ena       Dwn       Eth       3
4         Port_4     Ena       Dwn       WE        4
V2        Port_V2    Ena       Up        PPP       v1 SCID“SanDiego”
V4        Port_V4    Ena       Up        PPP       v2 SCID“SanJose”
V5        Port_V5    Ena       Down      PPP       -
```

DialNoList

This parameter adds, deletes, edits, and displays a list of phone numbers with their associated attributes (baud rate, phone number, and position in the list). The following is the syntax for this parameter:

```
ADD !<port> -PORT DialNoList “<phone no>" [Baud = <rate>
  (1.2–16000)] [Type = Modem | Bri | Sw56 | WE | WEH0]
  [Pos = <number>]
DELete !<port> -PORT DialNoList “<phone no>"
SHow [!<port> | !*] -PORT DialNoList
```

If you specify WE or WEH0 as the Type value, the value entered for “<phone no>” is the NETBuilder II system port number the WAN Extender is connected to and a WAN Extender remote site’s profile ID. (The remote site profile has the remote site phone number.)

For WAN Extender, the Baud rate specification is ignored. The baud rate associated with a virtual port is derived from the actual connection bandwidth.

DialStatus

This parameter displays a WAN Extender virtual port’s path number, B channel number, and the network port if the path is up. The other fields in the display are the same as for other ports.

OWNer

This parameter indicates which NETBuilder II bridge/router physical ports are connected to WAN Extenders. All ports that use WAN Extender virtual paths use PPP as the data link protocol. This parameter does not indicate the data link protocol for a given port.

You do not need to configure any services on a port set to OWNer = WanExtender. The physical port and associated physical path are only used to support the WAN Extender virtual paths used by other ports in the system.

For example, to indicate that a WAN Extender is connected to the associated path for port 3, enter:

```
SETDefault !3 -PORT OWNer = WanExtender
```

PAths

This parameter assigns a path or multiple paths to the specified port, or assigns dial pool path resources to the specified port. Default ports must have a SCID string associated with them to allow SCID-based mapping of WAN Extender virtual paths to the ports over which incoming calls will arrive. For a complete description of this parameter, refer to Chapter 43 in *Reference for NETBuilder Family Software*.

PathPreference

This parameter is not used for WAN Extender.

VirtualPort

This parameter creates a virtual port that represents a logical attachment with the WAN Extender to a network at a remote site. The virtual port specification includes a system identifier string, which may be used during incoming call setup time to associate the caller (the remote site) with the corresponding virtual port. For example:

```
ADD !V1 -PORT VirtualPort SCID”Chicago”
```

Before version 9.1, there were “WAN Extender” virtual ports. These ports have been upgraded to SCID virtual ports in release 9.1.

Sample Configuration Verification Displays

This section provides sample displays and descriptions of the display elements for the SHow command and various WE Service parameters that are used to verify the configuration. For a detailed description and syntax of all WE Service parameters, refer to Chapter 64 in *Reference for NETBuilder Family Software*.

Configuration Setting Displays

To display the current WAN Extender system and network configuration settings for the port entered as well as for its Local Management Interface (LMI) parameters and their settings, use the configuration parameter. If no port is specified, then the configuration information for all WAN Extender ports (and their owners) are displayed in ascending order.

For example, enter:

```
SHow -WE CONFiguration
```

A display similar to the following appears:

```
WAN Extender Configuration
System Port Parameters
KeepAliveInt                      10
FullStatusFreq                     6
ErrorThreshold                     3
DialPathLimit                     10, 2
WAN Extender Device Parameters
Version                            WANExtender Rel 9.1 T1 7/96
Type                               WAN Extender 2T
Name                               NB2_4
Max Data Pkt Size                  1518 bytes
Network Port 0
switch type                        AT&T 5ESS
service variant                    AT&T Custom
rate adapted                       Disabled
call types allowed                 56KB, 64KB_Clear
Network Port 1
enabled channels                   0x00FFFFFF
circuit id                         Network Port 2
```

Connection and Data Packet Statistics Displays

You can display the connection and data packet statistics accumulated between the NETBuilder II bridge/router port and the WAN Extender that is connected to the port, and the statistics generated for the WAN Extender network ports by using the DevSTATistics parameter.

For example, enter:

```
SHow !4 -WE DevSTATistics
```

A display similar to the following appears:

```
Statistics from WAN Extender out port !4:
Network Port 1:
   Profile misses                                       2
   Calls made                                           24
   Calls received                                       2
   Out calls blocked                                    0
   Outgoing call bad profile                            0
   Internal errors                                      0
Network Port 2:
   Channelized profile errors                                  0
   Internal errors                                             0
Packet Transfer Statistics:
   Packets out network ports                            35167
   Pkts from NETBuilder dropped due to full queue       0
   Packets received from network ports                 34601
   Pkts from network ports dropped due to full queue    0
   LMI packets received from NETBuilder                 9842
   LMI Packets sent to NETBuilder                       9923
   Invalid DLCI occurrences from NETBuilder             9
   Idle DLCI occurrences from NETBuilder                2
```

This command can only be entered as a UI command at the local console. This parameter is not available through Scheduler or Remote commands.

Incoming and Outgoing Calls Displays

You can retrieve information from the WAN Extender that is connected to the NETBuilder II bridge/router port for the incoming and outgoing calls made through the port using the ProFile parameter.

For example, enter:

SHow !4 -WE ProFile 4

A default summary display similar to the following appears:

```
----- Profile #4 from WAN Extender out port 4 -----
Outgoing called number..... 4962134
 Outgoing calling number..... 9868404
Incoming called number..... 4962134
 Incoming calling number.....
```

If you enter:

SHow !4 -WE ProFile 4 Detail

A display similar to the following appears:

```
----- Profile #4 from WAN Extender out port 4 -----
Calls made........ 7
 In called number matches..... 0
 In calling number matches..... 0
 Version........ 0xA005
 Description..... 9868404 -> 4962134
 Profile type..... ISDN (0)
 Network port..... 1
 Outgoing called number type..... Subscriber
 Outgoing called number plan..... ISDN
 Outgoing called number..... 4962134
 Outgoing calling number type..... Subscriber
 Outgoing calling number plan..... ISDN
 Outgoing calling number..... 9868404
Incoming called number..... 4962134
 Incoming calling number.....
```

If you enter:

SHow !4 -WE ProFile 4 STATistics

A display similar to the following appears:

```
----- Profile #4 from WAN Extender out port 4 -----
Calls made........ 7
 In called number matches..... 0
 In calling number matches..... 0
```

Packet Counts Displays

You can display the WAN Extender-to-NETBuilder II bridge/router connection statistics as counted on each NETBuilder II bridge/router port with WAN Extender set as Owner by using the -SYS Service STATistics parameter. The connection statistics displayed include packet counts for WAN Extender virtual paths and WAN Extender local-link operation statistics.

For example, enter:

SHow -SYS STATistics -WanExtender

Refer to Appendix H for a sample display that comes up with this command and a description of the display elements.

Troubleshooting

If you have verified your configuration and have found problems with your system, you must troubleshoot the problems. This section describes what to check for in channelized leased-line configurations and switch-circuit configurations.

This section also provides information about WAN Extender and NETBuilder II bridge/router troubleshooting commands you can use to further verify and troubleshoot a configuration.

Troubleshooting Channelized Leased Configurations

To troubleshoot leased line problems, check to see if one or more of the following situations has occurred:

- The remote site is down or not connected.
- No profile is configured for the remote site, or if a profile is configured, it is configured incorrectly.
- The SysCallerID (SCID) string set for the virtual port designated for the remote site does not match the remote site system's Service SysCallerID parameter string.
- The local port cable connecting to network is not connected properly or is faulty.
- The network port cabling is not connected properly or is faulty.

Troubleshooting Switch Circuit Configurations

To troubleshoot an ISDN or switch 56 switch circuit problems, check to see if one or more of the following situations has occurred:

- The remote site is down or not connected.
- No profile is configured for the remote site, or if a profile is configured, it is configured incorrectly.
- The SysCallerID (SCID) string set for the virtual port designated for the remote site does not match the Service SysCallerID parameter string of the remote site system.
- The local port cable connecting to network is not connected properly or is faulty.
- The network port cabling is not connected properly or is faulty.

Using WAN Extender Troubleshooting Commands

The WAN Extender console port provides access to a set of commands for verifying the WAN Extender configuration and for troubleshooting the system operation. These commands can be used to:

- Display the contents of the configuration file the WAN Extender is currently running, including system parameters, port parameters, and remote site profiles.
- Trace messages passing between the WAN Extender and the NETBuilder II bridge/router.
- Trace call control messages on ISDN links.
- Display statistics on profile usage, packet transfers, and incoming/outgoing call completions.
- Retrieve diagnostic information following a WAN Extender failure.
- Reboot the WAN Extender.

CAUTION: *The WAN Extender troubleshooting commands perform actions that may seriously impact the ability of the WAN Extender to accept and initiate calls. These commands should only be used under the close supervision of qualified 3Com support technicians and only during periods of light or no-call traffic.*

Accessing the WAN Extender Console Interface

Connect a PC to the console port on the WAN Extender rear panel using the same console link cable as used for the WAN Extender Manager application. On the PC, run a terminal emulation program that is configured as follows:

- 9600 baud
- 8 bits
- No parity
- 1 stop bit
- No software flow control (XON and XOFF are ignored)

If you are using the same PC as the WAN Extender Manager, make sure you have exited from that program before using a terminal emulation program. Failure to do so will result in an error message that the port is in use.

When the terminal emulation program is running, press the Enter key several times until an > prompt appears, indicating that the WAN Extender is ready to accept troubleshooting commands.

Command Descriptions

You can display the list of troubleshooting commands by entering:

```
WE??
```

The WAN Extender troubleshooting commands are case-sensitive. The following list describes these commands:

WEvs — Displays system status.

WEvc — Displays connection states.

WEss — Displays system and network port-related configuration parameters, including version numbers.

WEsp <profile-id> — Displays contents of the remote site profile specified in <profile-id> and the statistics associated with that profile.

WEpc — Displays packet count statistics for the local port and network ports.

WErb — Reboots the attached WAN Extender.

WElb — Sets the attached WAN Extender in local loopback mode. In this mode, all frames transmitted by the NETBuilder II system on the local port are looped back to the NETBuilder II system without any change.

`WEdw` — Disables watchdog timer.

WEtp — **Do not use—for testing only.** Toggles profiles mode (O = loaded, I = static).

WEts — **Do not use—for testing only.** Toggles system parameter mode (O = loaded, I = static).

WErx — **Do not use—for testing only (use WEst instead)**. Toggles RX tracing (O = Off, I = On).

WEtx — **Do not use—for testing only (use WEst instead)**. Toggles TX tracing (O = Off, I = On).

WEes — **Do not use—for testing only.** Edits on-board system profile.

WEcr — Dumps 68ec030 registers after a board-level panic.

WEsa — **Do not use—for testing only.** Sets system to stand-alone mode to make test calls.

WEca <profile-id> — **Do not use—for testing only.** Makes a test call in stand-alone mode using a remote site profile specified in <profile-id>.

WEhu <dlci> — **Do not use—for testing only.** Hangs up a test call made in stand-alone mode. <dlci> is the connection identifier associated with that call.

WEtl — **Do not use—for testing only.** TSI loopback for ACCUNET testing (no host).

WEst <trace-level> Turns on or turns off tracing. A variety of trace options are available, each expressed as a hexadecimal value.

Use of the WEst<trace-level> command can seriously affect the performance of the WAN Extender. Do not turn on large combinations of trace types during periods of high-call traffic on the WAN Extender.

To get a combination of trace types, add up the hexadecimal values corresponding to the individual types. For example, if you want to enable these three trace types:

- Trace SMI messages exchanged between the NETBuilder II bridge/router and the WAN Extender (0x0020)
- Trace the handling of calls (connection establishment and teardown) (0x0004)
- Trace unexpected execution paths (0x0001)

Enter:

```
Superuser WAN Extender TraceLevel 25
```

Table 36-1 lists the traces and their hexadecimal values.

Table 36-1 Traces and Their Hexadecimal Values

Value	**Meaning**
0x0000	Turns trace off.
0x0001	Traces all LMI messages.
0x0004	Traces race conditions.
0x0008	Traces minor debug information.
0x0010	Traces the processing flow.
0x0020	Traces timer processing.
0x0040	Traces restart state machine.
0x0080	Traces span state machine.
0x0100	Traces error conditions.
0x0200	Traces control messages.
0x0400	Traces call control error paths.
0x0800	Traces call control flow.

In addition to commands, the WAN Extender also supports ISDN link-level tracing on the network ports:

- To get basic level of tracing, at the WAN Extender console prompt, enter lowercase L and digit one. For example:

  ```
  l 1
  ```

- To get expanded tracing, at the WAN Extender console prompt, enter lowercase L and digit two. For example:

  ```
  l 2
  ```

- To turn off link-level tracing, at the WAN Extender console prompt, enter lowercase L and digit zero. For example:

  ```
  l 0
  ```

The trace displays ISDN Layer 2 and Layer 3 call control messages exchanged between the WAN Extender and the network as shown in a display similar to the following:

```
Ch#   Time  Direct SAPI   TEI   C/R   Type     N(s)  N(r) P/F  Size
-------------------------------------------------------------------
00    1AAF  Xmit   00     00    0     SABME               1    0003
00    1AB3  Rcvd   00     00    1     SABME               1    0003
00    1AB3  Xmit   00     00    1     UA                  1    0003
01    1AB4  Rcvd   00     00    1     UA                  1    0003
00    1B7A  Xmit   00     00    0     Setup    00    00   0    0027
01    1B7C  Rcvd   00     00    0     Setup    00    00   0    0027
01    1B7C  Xmit   00     00    1     Prcdng   00    01   0    000E
00    1B7D  Rcvd   00     00    1     Prcdng   00    01   0    000E
00    1B7D  Xmit   00     00    1     RR             01   0    0004
01    1B7D  Xmit   00     00    1     Alrtng   01    01   0    0009
01    1B7E  Rcvd   00     00    1     RR             01   0    0004
00    1B7E  Rcvd   00     00    1     Alrtng   01    01   0    0009
00    1B7E  Xmit   00     00    1     RR             02   0    0004
01    1B7E  Xmit   00     00    1     Connct   02    01   0    0009
01    1B7F  Rcvd   00     00    1     RR             02   0    0004
00    1B7F  Rcvd   00     00    1     Connct   02    01   0    0009
00    1B7F  Xmit   00     00    0     ConAck   01    03   0    0009
01    1B81  Rcvd   00     00    0     ConAck   01    03   0    0009
```

Using NETBuilder II Troubleshooting Commands

NETBuilder II bridge/router troubleshooting commands consist of the SuperUser command with WAN Extender parameters, and the SHow command with the normal WE Service parameters. This section describes the WAN Extender Service parameters that are used with the SuperUser command and shows the displays that they generate.

For a description of the normal -WE Service parameters, refer to Chapter 64 in *Reference for NETBuilder Family Software*; to see display samples, refer to "Sample Configuration Verification Displays" on page 36-20 in this chapter.

NETBuilder II troubleshooting commands allow qualified 3Com technicians to monitor, diagnose, or troubleshoot the WAN Extender and NETBuilder II bridge/router operations.

CAUTION: *The NETBuilder II troubleshooting commands for WAN Extender perform actions that may seriously impact of the WAN Extender ability to accept and initiate calls. These commands should only be used by qualified 3Com support technicians or under their close supervision and only during periods of light or no-call traffic.*

The commands are grouped into the following categories:

- Tools for configuration verification

  To verify that the downloaded configuration on the WAN Extender is correct, use:

  ```
  SuperUser WanExtender !<WE-port> SystemInfo
  SuperUser WanExtender !<WE-port> GlobalSystemParms
  SuperUser WanExtender !<WE-port> NetPortParms
  SHow !<WE-port> -WE ProFiles
  SHow !<WE-port> -WE CONFiguration
  ```

- Tools for verification of correct operation

  To verify that the NETBuilder II system in conjunction with the WAN Extender is operating as configured, use:

  ```
  SuperUser WanExtender DisplayActiveConnections
  SHow !<WE-port> -WE DevSTATistics
  ```

- Tools for problem diagnosis

  To analyze and diagnose a problem when the NETBuilder II system or the WAN Extender are not operating as they were configured, enter:

  ```
  SuperUser WanExtender TraceLevel
  SuperUser WanExtender DlciTrace
  ```

- Tools for detailed debugging

  The detailed debugging commands are reserved for use by engineers to debug problems that are difficult to analyze and diagnose with the other tools, and are not described in this chapter.

WAN Extender Service Parameters

This section provides a description, the syntax, and a display sample for the WAN Extender Service parameters, which are used with the SuperUser command.

SystemInfo. This parameter retrieves and displays system type, memory configuration, and software version data from the attached WAN Extender using:

```
SuperUser WanExtender !<WE-port> SystemInfo
```

For example, enter:

```
SuperUser WanExtender !4 SystemInfo
```

A display similar to the following appears:

```
Global System Info parameters from WAN Extender out path 4:
isdn_version:
PRIS48M Rev5.20g 4/16/96 5.2.g
we_version:
WanExtender Rel1.15E4 4/96
pcmcia_mem 524288 bytes
mem_size ..2097152 bytes
type ......WAN Extender 2T
```

GlobalSystemParms. This parameter retrieves and displays the system-level parameters from the attached WAN Extender using:

```
SuperUser WanExtender !<WE-port> GlobalSystemParms
```

For example, enter:

```
SuperUser WanExtender !4 GlobalSystemParms
```

A display similar to the following appears:

```
Global System parameters from WAN Extender out path 4:
version ..............0xA005
name .................NB2_4
clock source .........from Net Port 1
configured WE type ...WAN Extender 2T
baud .................4096 Kbps (local link)
max data pkt size ....1518 bytes
Console trace level ..NONE
lapb .................DISABLED
```

NetPortParms. This parameter retrieves and displays the parameters configured for each network port of the attached WAN Extender using:

```
SuperUser WanExtender !<WE-port> NetPortParms
```

For example, enter:

```
SuperUser WanExtender !4 NetPortParms
```

A display similar to the following appears:

```
System parameters for Network Port 1 from WAN Extender out path 4:
call control .......ISDN
hunting ............ASCENDING
framing ............Extended Superframe
line code ..........B8ZS
equalization .......0-133 ft from CSU
port digits ...............986404
port digits number type ...Subscriber
port digits number plan ...ISDN
switch type ...............AT&T 5ESS
service variant ...........AT&T Custom
enabled bchannels .........0x007FFFFF
inverted HDLC .............Disabled
rate adapted ..............Disabled
link termination type .....User Side
call types allowed ........56KB 64KB_Clear
call filtering ............Disabled
ISDN Low Level Parameters:
T200 ..............DEFAULT USED
T203 ..............DEFAULT USED
N200 ..............DEFAULT USED
transmit_window ...DEFAULT USED
```

Network Port 2 on WAN Extender out path 4 is configured as UNUSED.

DisplayActiveConnections. This parameter displays summary information related to currently active connections that have been established by the NETBuilder II system through the attached WAN Extender. Activate this parameter using:

```
SuperUser WanExtender DisplayActiveConnections
```

For example, enter:

```
SuperUser WanExtender DisplayActiveConnections
```

A display similar to the following appears (the first column values are virtual paths):

```
V11   path:4   pid:4    dlci:1    kbps:64   Nport:2   channels:x00000001
V56   path:7   pid:56   dlci:32   kbps:64   NPort:2   channels:x01000000
V57   path:7   pid:57   dlci:33   kbps:64   Nport:2   channels:x02000000
V58   path:7   pid:58   dlci:34   kbps:64   Nport:2   channels:x04000000
V59   path:7   pid:59   dlci:35   kbps:64   Nport:2   channels:x08000000
V60   path:7   pid:60   dlci:36   kbps:64   Nport:2   channels:x10000000
v61   path:7   pid:61   dlci:37   kbps:64   Nport:2   channels:x20000000
V62   path:7   pid:62   dlci:38   kbps:64   Nport:2   channels:x40000000
```

TraceLevel. This parameter is used to turn on and off NETBuilder II bridge/router tracing on WAN Extender channels. Activate this parameter using (the <hex-mask> value is a bit mask that is used to indicate the types of tracing):

```
SuperUser WanExtender TraceLevel <hex-mask>
```

The following lists shows the hex values and the type of tracing each represents:

- 0x0000 disables tracing.
- 0x0001 traces unexpected execution paths.
- 0x0002 traces Simple Message Interface (SMI) messages exchanged between the NETBuilder II system and the WAN Extender. SMI messages are part of the WNI protocol.
- 0x0004 traces, with detailed information, SMI messages exchanged between the NETBuilder II system and the WAN Extender.
- 0x0008 traces the flow of messages through the system.
- 0x0010 provides a raw (hexadecimal) dump of all SMI messages received and sent by the NETBuilder II system.

To enable more than one type of tracing, the hex values corresponding to the types should be added and the resulting value specified for <hex-mask>.

For example, if you want to enable the following tracing types:

- Provide a raw (hexadecimal) dump of all SMI messages received and sent by the NETBuilder II system. (0x0010)
- Trace SMI messages exchanged between the NETBuilder II bridge/router and WAN Extender, (0x0004)
- Trace unexpected execution paths, (0x0001)

Enter:

```
SuperUser WanExtender TraceLevel 15
```

DlciTrace. This parameter turns tracing on and off for all data and WNI frames on a single data link connection Identifier (DLCI) or all DLCIs currently established between the NETBuilder II system and the WAN Extender. DLCI traffic for multiple interfaces is displayed without differentiating the traffic of one interface from the other if more than one WAN Extender is connected to a NETBuilder II system.

Use this parameter only with one WAN Extender port enabled at a time or if DLCIs are not used by more than one WAN Extender interface. Refer to "DisplayActiveConnections" on page 36-28 to determine which DLCIs are in use.

To activate this parameter, use:

```
SuperUser WanExtender DlciTrace OFF | <dlci> | ALL
```

How the WAN Extender Works

A WAN Extender provides virtual paths to be used for interconnecting remote NETBuilder devices to a central site NETBuilder II bridge/router running PPP. The interconnection is established using channelized T1 and E1 leased-circuit services, and switched 56 and ISDN PRI switched-circuit services.

WAN Extender Models

There are two WAN Extender models:

- WAN Extender 2T

  The WAN Extender 2T is intended for WAN networks that support the T1 interface. It provides two network interfaces, each of which can be independently connected to channelized T1, switched 56, or ISDN PRI services. The WAN Extender 2T supports 24 channels on each network interface for channelized T1 and switched 56 environments. It supports 23 B channels and one D channel on each network interface for ISDN PRI environments.

- WAN Extender 2E

  The WAN Extender 2E is intended for WAN networks that support the E1 interface. It provides two network interfaces, each of which can be independently connected to channelized E1 or ISDN PRI services. The WAN Extender 2E supports 31 channels on each network interface for channelized E1 environments. It supports 30 B channels and one D channel on each network interface for ISDN PRI environments.

How Virtual Paths are Created

The WAN Extender provides an RS-530 connector (called the local port), which connects to a NETBuilder II bridge/router high-speed serial (HSS) RS-449 module, to provide a synchronous link between the two devices. A 3Com proprietary interface protocol, called the WAN Extender/NETBuilder II Interface (WNI) Protocol, runs on this link.

The WAN Extender virtual paths are created automatically by the NETBuilder II bridge/router after it synchronizes with the WAN Extender over this link. Each virtual path can initiate a call to the WAN Extender and accept a call from the WAN Extender. There are three types of virtual paths: leased, DS0 dial, and H0 dial virtual paths.

Leased Virtual Paths

During the synchronization between the NETBuilder II bridge/router and the WAN Extender, the NETBuilder II bridge/router reads the profiles residing in the WAN Extender and sets aside a virtual path for each channelized T1 or E1 leased-line profile configured.

Although each virtual path is allotted one channel with a baud rate of 64 kbps when created, the channel expands to the size of the sum of all the channels specified in the profile when a connection is established.

The leased virtual paths occupy the bottom of the virtual path ID range.

DS0 Dial Virtual Paths

The virtual paths that are not used for leased lines are automatically available as dynamic paths in a dial-up path pool for interconnecting remote devices over switched ISDN or switch 56 lines.

The number of DS0 Dial virtual paths that are actually created is determined by the DialPathLimit setting, which considers the following information:

- The NETBuilder II bridge/router supports a maximum of 75 virtual paths.
- The number of virtual paths configured to be used for channelized leased lines.
- The number of virtual paths already configured for dial-up.
- The maximum number of channels that can be supported per port of the WAN Extender model being used (T1 supports 23 and E1 supports 30).

If the DialPathLimit setting is greater than the number of virtual paths that can be supported by the WAN Extender port, the number of virtual paths created will be the number of virtual paths supported, which is the smaller amount. For details on setting the DialPathLimit for DS0 virtual paths, refer to Chapter 64.

The DS0 Dial virtual paths occupy the top of the virtual path ID range.

H0 Virtual Paths

The number of H0 Dial virtual paths created is determined entirely by the value set for the H0 path count with the -WE DialPathLimit parameter. The range of H0 virtual paths is 0 to 3. Each H0 virtual path is 384 kbps, or equal to six DS0 dial virtual paths (6 x 64 kbps = 384 kbps). H0 and DS0 virtual paths can run on the same port at the same time.

For details on setting the DialPathLimit for H0, refer to Chapter 64.

For a complete description on ports and paths including how to number them, and for a description on how to set up ports and paths for a bridge/router using wide area interfaces, refer to Chapter 1.

For information on setting up physical and virtual paths in dial-up pools for ISDN and switch 56 lines, refer to Chapter 37.

How the WAN Extender Operates

The WAN Extender is managed by an external software application called the WAN Extender Manager, which runs under Microsoft Windows 3.1 or later on a PC. The PC connects to the console port on the WAN Extender.

The WAN Extender maps each WAN connection to a data channel and makes that data channel available to the NETBuilder II bridge/router through a virtual path. The NETBuilder II bridge/router operates as follows:

- Views the data channel (a virtual path) as the underlying link for a virtual port
- Uses the data channel as if it were a clear channel
- Transparently establishes the end-to-end data link through the WAN Extender

When a connection to a remote site is first made using the WAN Extender, the end-to-end data link is established using PPP. After the end-to-end link is established, various higher-layer PPP NCP negotiations occur, depending upon your configuration at either end of the link, and then network layer protocol connection is established. Figure 36-4 shows the WAN Extender connection.

NETBuilder II bridge/router PPP-based virtual ports can be used to establish bridging, routing, and Boundary Routing connectivity. The NETBuilder software operates the same way for WAN Extender virtual path-based ports as for any other point-to-point virtual port or port running PPP.

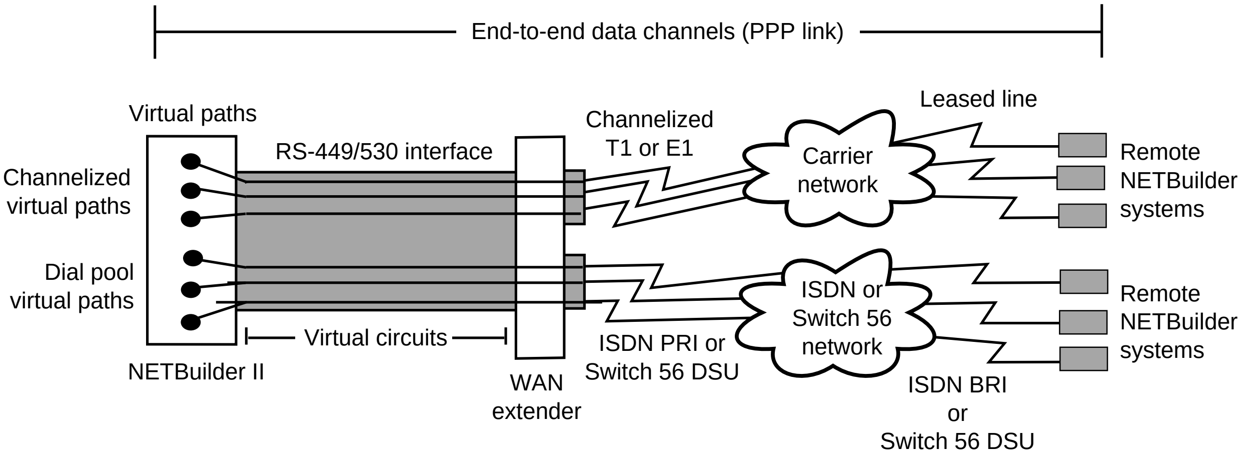

End-to-end data channels (PPP link)
Virtual paths
Channelized virtual paths
Dial pool virtual paths
NETBuilder II
RS-449/530 interface
Virtual circuits
WAN extender
Channelized T1 or E1
ISDN PRI or Switch 56 DSU
Carrier network
ISDN or Switch 56 network
Leased line
ISDN BRI or Switch 56 DSU
Remote NETBuilder systems
Remote NETBuilder systems

Figure 36-4 WAN Extender Connections From NETBuilder II to Remote NETBuilder Systems.

When a frame is received by the WAN Extender from the NETBuilder II bridge/router, frame forwarding proceeds as follows:

1. The WNI protocol header is stripped.
2. The data channel identifier is extracted from the WNI protocol header.
3. The data channel identifier is mapped to the corresponding network channel.
4. The frame is transmitted on the network channel.

When a frame is received by the WAN Extender on a network channel, frame forwarding proceeds as follows:

1. The network channel is mapped to the corresponding data channel.
2. The frame is prepended with a WNI protocol header containing a data channel identifier.
3. The frame is transmitted on the WAN Extender-to-NETBuilder II bridge/router connection.

When either the NETBuilder II bridge/router or the WAN Extender are initialized, or any time the link between the two systems is connected or enabled, the two systems engage in a local link synchronization process. On the NETBuilder II bridge/router side, the path associated with the WAN Extender connection goes to an UP state. After synchronization, the data channels may also get established, and the virtual port associated with each data channel may also go to an UP state. The data channel that gets established, and its associated virtual port that goes to an UP state, depends on the type of network the WAN Extender is connected to and the NETBuilder II bridge/router ports configurations.

37 CONFIGURING PORT BANDWIDTH MANAGEMENT

This chapter describes how to configure communication resources (telephone lines and digital circuits such as ISDN, T1/E1, and T3/E3 lines) for use with your dial-up wide-area network (WAN) lines. In a 3Com WAN, you use *port bandwidth management* to control your communication resources. The concepts and configuration examples provided in this chapter will help you to decide how to use port bandwidth management to make effective and efficient use of your WAN communication resources.

In software version 9.1, port bandwidth management is applicable to PPP WAN paths only.

Communication Resources Supported

Bandwidth management supports a broad range of communication resources, from public telephone lines utilizing inexpensive analog modems to digital circuits providing throughput at rates up to 45 Mbps.

The virtual pipe can consist of any of the communication resources listed in Table 37-1.

Table 37-1 Communication Resources Supported by Bandwidth Management

Resource	Throughput
Telephone line; analog line utilizing the public telephone system	Up to 28.8 kbps
Leased line; dedicated higher quality analog line	Up to 56 or 64 kbps
ISDN BRI; dedicated narrowband digital service	56/64 or 112/128 kbps
ISDN PRI; dedicated narrowband digital service	23 or 30 channels at 64 kbps
Switched-56; nondedicated digital service	Up to 56 kbps
Fractional T1; dedicated digital circuit	Increments of 64 kbps
T1 or E1 line; dedicated digital circuit	24 or 30 channels at 64 kbps each
T1 or E1 channel; dedicated digital circuit	1.544 Mbps or 2048 kbps
T3 or E3 line; dedicated digital circuit	Up to 45 Mbps

"Configuring WAN Resources" on page 37-8 provides instructions to configure these resources.

DTE Serial Lines

When using serial Data Terminal Equipment (DTE) dial-up lines, the bridge/router software supports the V.25 bis standard, which allows a DTE device to communicate with a Data Communication Equipment (DCE) modem or terminal adapter (TA). With V.25 bis, you can configure and store phone numbers in software for the modem or TA on the bridge/router. The phone number is sent to the bridge/router modem or TA when dialing occurs.

The software can also activate modems or TAs that store a phone number in the firmware of a data terminal ready (DTR) dialed modem or TA. DTR modems can only be used in static configurations; dynamic paths selected from a resource pool rely upon telephone numbers stored in the bridge/router software.

ISDN Lines

The Integrated Services Digital Network (ISDN) interface is supported on model 42x and 52x SuperStack II NETBuilder bridge/routers. These bridge/routers offer one basic rate interface (BRI) with two B channels (2B+D). 3Com also has a list of BRI terminal adapters it recommends that allow non-ISDN routers such as the NETBuilder II bridge/router to connect to an ISDN network.

WAN Extender Virtual Paths

A WAN Extender virtual path is either ISDN 64 kbps, ISDN H0, switched-56, or channelized leased lines. Bandwidth management considers a WAN Extender virtual path as a generic path that can be allocated as a resource for the virtual pipe. Refer to Chapter 36 for more information.

Associating Paths to Ports

3Com software uses the concepts of *ports* and *paths* to address interface connections. The basic interface connection is a port. A port is a logical interface that represents a connection to a network. The next logical connection is the path, which is the physical interface that connects the bridge/router to a physical medium such as an Ethernet local-area network (LAN), a token ring, or a serial line. In an ISDN environment, a path additionally represents the channel over which data is transmitted. This section describes the paths that can be configured on ports under bandwidth management.

Static versus Dynamic Paths

Prior to software version 8.0, you were able to statically configure a path resource to a logical port through which data flowed to other destinations. A single phone number, for example, was assigned to a path, path-based dialing was attempted, and any failed call attempt resulted in the redialing of the same phone number.

Beginning with software version 8.0, it was possible to unbind static paths from their ports and save them in a dial pool to be shared by more than one port. The paths in the dial pool are called *dynamic paths*. A path in the dial pool can be *dynamically bound* to a port or PPP virtual port when the path is needed to transfer data on a dial-up line. After the path is bound to the port, port-based dialing occurs. When the traffic is no longer present and the line is idle for a specified period, the path is unbound from the port and returned to the dial pool. For an inbound call, a system caller ID is matched to the appropriate port and the path is bound to the port.

Software version 9.1 removes all limitations on port and path bindings, allowing a port to bind to multiple dial paths. For each port, bandwidth can be dynamically allocated by bundling the multiple dial paths into the virtual pipe. Bandwidth can be allocated when and where it is most needed.

Software version 9.1 also introduces WAN Extender virtual paths, which are considered by bandwidth management to be generic paths that can be allocated to a logical or virtual port. Virtual paths are available and not tied to a specific physical resource until they are bound to a port. For virtual paths assigned to dial pools, the binding of the virtual path to a port occurs when an incoming call is received or when an outgoing call is started.

Multidestination Dialing

You can use a dial pool to increase the reliability of your network configuration, achieve multidestination dialing by using dial phone number lists and modem pooling, and provide dynamic backup for leased or dial-up lines.

With multidestination dialing, you can allocate a small number of paths that are unbound from their ports to wait in the dial pool for an incoming call. You can create a PPP virtual port on the central router for each remote site and have all the virtual ports use the dial pool for path resources.

When the system receives an incoming call from a remote site, the dynamic path that answers is bound to a virtual port, which is standing by with the appropriate configuration information for the calling network. For the binding to occur, the remote site caller ID specified by the -SYS SysCallerID parameter is transmitted to the central router.

Because not all sites using a dial pool will be calling the central site at the same time, it is possible to share a small group of paths with a larger group of sites. Each site that can potentially call into the dial pool has its own virtual port predefined, so there can be more virtual ports configured to use the dial pool than there are dynamic paths assigned to the dial pool. However, if all the remote sites dial the central router at the same time and only a small number of paths exist in the dial pool, some of the call attempts may fail due to a lack of path resources. These calls can be redialed at a later time.

For a summary of the terms used in this section, refer to "Bandwidth Management Terms" on page 37-31. For more information about the dial pool, refer to "Resource Aggregation" on page 37-6.

Valid Port and Path Configurations

Table 37-2 lists the valid combinations of port and path binding configurations available beginning with software version 9.1.

Table 37-2 Valid Port and Path Configurations

Type of Path	Default Ports Prior Releases	Virtual Ports Prior Releases	Default Ports Release 9.1	Virtual Ports Release 9.1
Static leased line	1 or more	Not supported	1 or more	Not supported
Static dial path	1 or 2	Not supported	1 or more	Not supported
Dynamic dial physical path	1	1	1 or more	1 or more
Dynamic WAN Extender virtual path	Not supported	1	1 or more	1 or more

System Bandwidth Management

Bandwidth management provides two operating modes: system bandwidth management and manual bandwidth management. These modes are enabled with the -PORT DialInitState parameter; the DialOnDemand option enables system bandwidth management mode, and the ManualDial option enables manual bandwidth management mode. This section describes system bandwidth management. Refer to "Manual Bandwidth Management" on page 37-7 for information about the manual bandwidth management mode.

System bandwidth management provides you with automated bandwidth management features. You provide bandwidth allocation guidelines and the system automatically manages the virtual pipe for you by monitoring traffic rates on the line. When traffic increases, bandwidth management may automatically allocate additional bandwidth; when traffic decreases, it may automatically decrease the bandwidth.

Changes to bandwidth and line characteristics take effect immediately. The system also uses a phone number list that you define to automatically obtain additional bandwidth. If bandwidth requirements consume more than one path, the system picks additional paths to form the virtual pipe and meet the requirements.

Dial-on-Demand

Dial-on-demand (DOD) is a more economical way to use phone lines when communicating between bridge/routers. It is supported only under system bandwidth management mode and is enabled with the DialOnDemand option of the -PORT DialInitState parameter. DOD is triggered on when there is traffic on a port, and triggered off when the port is idle or experiences a decrease in traffic congestion. The careful monitoring of traffic provided by system bandwidth management allows more cost effective-use of DOD lines.

A connection is established when the system automatically dials a phone number specified with the -PORT DialNoList parameter, or the phone number configured in the DTR modem. The line stays up as long as traffic is present. When there is no more data, the call is terminated. It is automatically reestablished without any intervention when there is data to be sent across the line. Connections that are no longer in use are temporarily terminated until new demand occurs. The -PORT DialIdleTime parameter determines how long a connection must be idle before the call is terminated.

In general, DOD will limit background traffic to routed packets (DECnet, IP, and IPX-routed packets) and other network protocol packets (IPX RIPs and SAPs) that are absolutely necessary to maintain the functionality and integrity of the overall network. The software feature set provides you with the necessary parameters for controlling the traffic over that DOD link. For phone lines and the connections associated with those lines to operate properly in the DOD state, the network layer protocols running over those connections must use statically defined routes or Open Shortest Path First (OPSF) demand circuits (per RFC 1793). Currently IP, IPX, and DECnet are the only network layer protocols supported with DOD. For procedures and configuration examples of IP and IPX routing over a DOD link, refer to "Routing Configurations over DOD Links" on page 37-24. For information about routing DECnet over a DOD link, refer to Chapter 15.

Bandwidth-on-Demand

Bandwidth-on-demand (BOD) is triggered on when the system detects traffic congestion on a port configured for system management mode. With software version 8.3, the sensitivity of the trigger-up and trigger-down threshold of a BOD line was adjusted manually using the -PORT BODTHreshold parameter. Beginning with software version 9.1, you specify the bandwidth that a port should operate at normally, then define the maximum amount of bandwidth above this setting that the port can have. Together these settings define the maximum width of the virtual pipe.

The BOD allocation strategy provides a flexible approach for configuring WAN dial-up lines. For example, the normal operating bandwidth of a WAN with two 64 kbps ISDN lines could be configured together for a total bandwidth of 128kbps, or be configured at 64 kbps bandwidth with incremental increases up to 128 kbps, as traffic needs required. Depending upon traffic across your network, you can choose to configure one wide virtual pipe to handle the traffic, or configure a narrower virtual pipe that expands and contracts as traffic increases or decreases.

Bandwidth allocation is defined using the -PORT NORMalBandwidth, BODTHreshold, BODIncrLimit, and DialSamplPeriod parameters, which specify bandwidth settings and the conditions that trigger BOD. Bandwidth management monitors the outgoing rate of traffic and uses the settings to prevent dropped packets by changing the size of the virtual pipe (allocating or removing lines and bandwidth) as required by traffic demands.

You can also configure a line as a general purpose line that can be allocated for any purpose, including disaster recovery, using the UnReStricted option of the -PATH DialCONTRol parameter. When a line failure causes the port bandwidth to drop below the level specified with the -PORT NORMalBandwidth parameter, the DOD strategy and bandwidth management work to restore the specified bandwidth. If traffic conditions warrant additional bandwidth, then BOD increases the bandwidth accordingly.

Disaster Recovery

In previous releases of the software, you were able to configure one secondary path as a backup or as a disaster recovery line to a primary leased line path. Beginning with software version 9.1, this restriction is removed, and the definition of disaster recovery changed.

Disaster recovery is now the disaster recovery threshold, which is defined as the minimum of the normal bandwidth threshold, as defined by the -PORT NORMalBandwidth parameter, and the total amount of configured leased line bandwidth that is assigned to the port, excluding disabled paths.

When the total active bandwidth from the leased line paths falls below the disaster recovery threshold, bandwidth management tries to recover the port bandwidth to the target set with the -PORT NORMalBandwidth parameter using dial paths. In this event, a path configured for disaster recovery is given preference. You configure a line specifically for disaster recovery using the DisasterRcvry option of the - PATH DialCONTRol parameter.

Path Configuration Summary

Table 37-3 summarizes the path configurations available with system bandwidth management in software version 9.1 and prior software releases.

Table 37-3 Path Configurations Available with System Bandwidth Management

Strategy	Prior Releases	Release 9.1
Dial-on-demand	Single path for dial-on-demand	Single or multiple dial paths acting as a bundle
Bandwidth-on-demand	Single dial path for bandwidth-on-demand bandwidth aggregations	Single or multiple dial paths for bandwidth-on-demand bandwidth aggregation
Disaster recovery	Single dial path for disaster recovery on a manual dial port; no disaster recovery support for dial-on-demand port	Single or multiple dial paths for disaster recovery on all dial PPP ports

Resource Aggregation

This section describes how bandwidth management finds additional WAN resources to aggregate for the virtual pipe.

Dial Number List

The software can select a phone number from a list of destination phone numbers associated with a port. The bridge/router can automatically select alternate phone numbers as backup if a previously dialed phone number does not result in a connection. The software knows whether the dial-up line is a static or dynamic path. If it is a static path, the software obtains a phone number from the dial number list specified with the -PORT DialNoList parameter and brings up the path. If DTR dialing is being used, the software uses the phone number stored in the modem.

If the dial-up line is a dynamic path, the software searches a dial pool to obtain an additional path. The software obtains a path from the dial pool, and binds it to the port. The software obtains a phone number for the dial number list specified with the -PORT DialNoList parameter. By using the dial number list, the software can try other phone numbers if the first number is unavailable. When searching the dial number list, the software seeks a path that matches the baud rate set for the path and will choose the path that matches or most closely matches that rate. By using the dial pool, the software can select another dynamic path if the first path is not working. When traffic conditions return to normal, the dial-up path is unbound from the port and returned to the dial pool.

Prioritized Path Preferences

You can prioritize dynamic and static dial paths for use by specific ports using the -PORT PathPreference parameter. No other port can use these reserved paths, unless these paths are reserved by more than one port. The software tries the preferred list of paths first before using path resources in the rest of the dial pool, and will seek static paths first. By specifying your path preference, you can reserve path resources for your dial-up lines, and ensure that a path is always available.

When you specify the selection sequence for a static path, you actually specify the priority sequence of selection for use by a port. A static path is inserted at the end of the path preference list by default. A static dial path with either the V.25 bis or DTR dial mode can be inserted in the list, but DTR does not use the dial number in the DialNoList parameter to dial out; instead, it uses the dial number stored in the modem.

Leased line paths cannot be included in the path preference list because bandwidth management cannot bring a leased line up or down; leased line paths are brought up when the port is enabled.

Manual Bandwidth Management

Manual bandwidth management is enabled with the -PORT DialInitState parameter; the ManualDial option enables manual bandwidth management mode. Manual bandwidth management mode requires user intervention to control bandwidth on the line. You set a fixed amount of bandwidth using the -PORT NORMalBandwidth parameter, and then issue the DIal command to bring up the line. Bandwidth management tries to meet the bandwidth specification, but does not monitor traffic or make any dynamic changes based on traffic rates. Bandwidth changes are only made according to what you specify with the -PORT NORMalBandwidth parameter.

Manual Dial

Under manual bandwidth management, a line is brought up manually using the DIal command. The call remains connected until a timer expires or until you end the connection using the HangUp command. After the call has been disconnected, it can be reestablished only by issuing another DIal command.

The DIal command has a path mode and a port mode, and operation of this command differs depending upon the mode specified. The path-based DIal command is used mainly for testing and in event-based macros that automate line backup processes. Refer to the description of this command in Chapter 1 in *Reference for NETBuilder Family Software*.

The port-based DIal command manually dials on the specified port. The command accepts static or dynamic port numbers and an optional dial string. If a dial string is entered, the number must be listed in the -PORT DialNoList parameter. The call is placed on the available highest-priority phone number specified for the port using the -PORT DialNoList parameter. If the highest prioritized phone number is not available, the software tries to use the next phone number specified for the port, if more than one phone number is configured.

A telephone number must be inserted in the -PORT DialNoList parameter for dialing to occur, (although you can temporarily override the phone numbers in the DialNoList parameter by specifying a port and dial string with the port-based DIal command).To complete the call, the software automatically finds a path by first checking if a path is available in the path preference list. If one is, that path is used. If no path is available, the software determines whether the port can use the dial pool. If the port can use the dial pool, the software checks for an available path in the dial pool, binds it to the port, and makes the call.

Bandwidth management manages DIal command calls and makes bandwidth evaluations based on the -PORT NORMalBandwidth setting. All ports configured for DIal must have a positive bandwidth setting. Bandwidth management aggregates bandwidth resources as needed to meet the NORMalBandwidth setting, dialing for more resources if additional bandwidth is needed, or hanging up or substituting resources if less bandwidth is needed. You specify which phone number and path to select with the -PORT DialNoList and PathPreference parameters.

Manual Hangup

The port-based HangUp command manually brings down all dial path resources and is the default. Under manual bandwidth management mode, the port is brought down when HangUp is issued unless there are leased lines active.

You can hang up calls on dynamic paths or static paths. If the path is dynamic and is currently bound to a port, the HangUp command disconnects the call, unbinds the path, and places the path back into the dial pool.

Manual Bandwidth Management Disaster Recovery

Under manual bandwidth management, the software only maintains the normal operating bandwidth of a port; there is no monitoring and automatic allocation of additional resources if the line goes down. You can specifically configure disaster recovery for a leased line using the DisasterRcvry option of the -PORT DialCONTRol parameter. When disaster recovery is enabled, manual bandwidth management will restore the line bandwidth back to its original setting when a leased line fails and the port bandwidth falls below what has been specified for normal operation.

Bandwidth Management Status Displays

You can monitor the dial path status, including the state of the ports under bandwidth management control, using the -PORT DialSTatus parameter. The display from this parameter shows total port bandwidth, messages indicating congestion levels, and the intentions of the bandwidth manager for allocating additional resources.

Bandwidth Management Statistical Displays

You can control costs by monitoring connection charges. A dial MIB to support this monitoring facility is provided in software version 8.0 and later. To obtain DOD statistics, refer to Appendix H.

Configuring WAN Resources

The procedures in this section prepare your WAN resources for use with bandwidth management. See Figure 37-1 for an illustration of the configuration examples in this section.

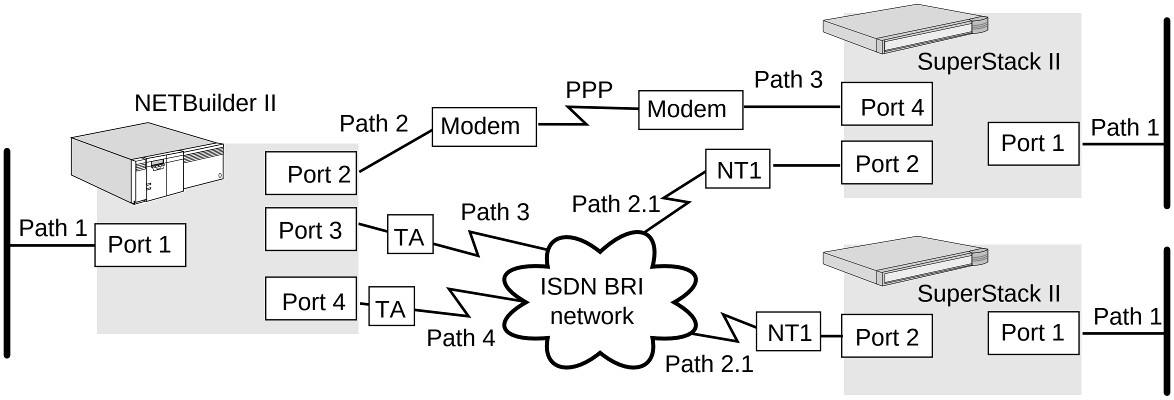


Figure 37-1 Basic Bandwidth Management Port and Path Configurations

Configuring Dial-Up Lines Using a Modem or TA

You can use digital ISDN and analog serial lines to establish connectivity with remote sites so that these sites can send updates to a central location. You can also configure serial lines when using Boundary Routing software.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.

Procedure

To configure a DTE serial line or ISDN line with a TA to use with bandwidth management, follow these steps:

You must complete this procedure at both ends of the link.

1 Set up the line type.

a By default, the dial path is set to Auto for the SuperStack II boundary router. You can check that the setting has not been changed using:

```
SHow !<path> -PATH LineType
```

b To change the value of the parameter to Dialup, use the following syntax:

```
SETDefault !<path> -PATH LineType = Dialup
```

Use this syntax to select Dialup as the line type for the NETBuilder II bridge/router.

2 Specify the baud rate for the device.

a Set the baud rate for a serial line using:

```
SETDefault !<path> -PATH Baud = <kbps> (1.2-52000)
```

The default baud rate for a serial line is 64 kbps. The auto startup feature automatically detects modem connections on the SuperStack II bridge/router; it does not sense the baud rate for ISDN paths with external TAs attached.

b Set the baud rate for an ISDN line connected with a TA using:

```
SETDefault !<connectorID> -PATH Baud = <kbps> (1.2-52000)
```

The auto startup feature automatically detects modem and TA connections on the SuperStack II bridge/router. Refer to "Configuring ISDN Lines" on page 37-10 for more information about configuring ISDN lines without a TA.

3 Select the connector type using:

```
SETDefault !<path> -PATH CONNector = V35 | RS232 | RS449 | G703 |
 HSSI | X21
```

4 Set the transmit clock for the bridge/router using:

```
SETDefault !<path> -PATH CLock = TestMode | External | Internal
```

The Internal value applies to model 32x and 52x SuperStack II bridge/routers only. The External value allows the bridge/router to derive the transmit clock from either the send or receive timing clock supplied by the digital service unit/channel service unit (DSU/CSU) or by the attached modem.

5 Select either the V.25bis standard or DTR dialing mode using:

```
SETDefault !<path> -PATH DialMode = V25bis | DTRdial
```

Select V.25 bis to configure a DTE serial line using a V.25 bis-compatible modem. Select DTR dial to configure a line using a modem that uses the DTR signal to initiate a call.

If you are using the V.25 bis standard, specify the telephone number of the remote site being dialed using:

```
ADD !<port> -PORT DialNoList “<phone-no>”
```

For more information, refer to “Configuring the Dial List” on page 37-14.

6 Specify the external device type attached to the DTE path using:

```
SETDefault !<path> -PATH ExDevType = [Modem | Bri | Sw56]
```

Port-based dialing that uses phone numbers from the dial-number list always looks at the setting of the ExDevType parameter to select an appropriate path for the phone number and phone technology. The default setting of the ExDevType parameter is Modem; the default setting for the Type attribute of the DialNoList parameter is also Modem.

7 Set the path characteristics for the line using:

```
SETDefault !<path> -PATH DialCONTrol = ([DYNamic | STAtic],
  [DisasterRcvry | NoDisasterRcvry | UnReSTricted])
```

The -PATH DialCONTrol parameter provides several options for setting the line.

The STAtic value allows the selected path to be statically bound to its corresponding port and is the default. The DYNamic setting unbinds a path from its corresponding port and adds the path to the dial pool. A static path is not part of the dial pool. Placing a path in the dial pool allows the path to be used by any dial port. For a dial path to become a dynamic dial path, the -PATH LineType parameter must be set to Dialup.

You can also choose to set the line specifically for disaster recovery, or as unrestricted to allow it to be used for any purpose including disaster recovery. The NoDisasterRcvry option prevents the line from being used for disaster recovery and is usually assigned to the slowest or least reliable line on the network.

For example, to configure the analog serial line on path 3 for no disaster recovery, enter:

```
SETDefault !2 -PATH LineType = Dialup
SETDefault !2 -PATH Baud = 28.8
SETDefault !2 -PATH CONNector = RS232
SETDefault !2 -PATH CLock = External
SETDefault !2 -PATH DialMode = DTRdial
SETDefault !2 -PATH ExDevType = Modem
SETDefault !2 -PATH DialCONTrol = NoDisasterRcvry
```

8 Enable the line and make sure all settings on the path take effect using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

Configuring ISDN Lines

You can use ISDN lines to establish connectivity with remote sites so that these sites can send updates to a central location. Refer to Figure 37-1 on page 37-8 for an illustration of the configuration examples in this section.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.
- Acquire services from a telecommunications carrier.

Procedure

To configure an ISDN line for use with bandwidth management, follow these steps:

1 To set the line type to Dialup for SuperStack II bridge/routers with an ISDN interface, enter:

```
SETDefault !2.1 -PATH LineType = Dialup
```

2 Set the switch type.

By default, the switch type is set to European Telecommunications Standards Institute (ETSI). If you need to change the switch type setting, use:

```
SETDefault !<connectorID> -PATH SwitchType = ETSI | NTT | KDD |
 NI1 | ATT5ESS | DMS100 | VN3 | AUSTEL
```

ETSI is the default and is only for users in the United Kingdom and Germany. Refer to the Chapter 37 in the Reference for NETBuilder Family Software to determine which switch type settings are supported and how international users should configure this parameter.

3 Specify a local telephone number using:

```
SETDefault !<connectorID.channelID> -PATH LocalDialNo = “<string>”
```

4 If you are planning to use an additional channel as a backup line and your telecommunications carrier provided only one telephone number for all channels, specify a subaddress using:

```
SETDefault !<connectorID.channelID> -PATH LocalSubAddr =
 “<string>”
```

When specifying the subaddress, you can specify up to 20 ASCII characters. Refer to Chapter 35 for information on why you would set up a subaddress.

You can also specify the telephone number of the remote site being dialed by using the -PORT DialNoList parameter. The phone number usually includes the dial prefix, country code, area code, and possibly a subaddress assigned to your ISDN interface. If you specify a subaddress, you must separate the phone number from the subaddress with a semicolon (;). With ISDN phone numbers, you can use hyphens (-) to separate the prefix, country code, and area code. For more information, refer to “Configuring the Dial List” on page 37-14.

5 Set the path characteristics for the line using:

```
SETDefault !<path> -PATH DialCONTrol = ([DYNamic | STAtic],
 [DisasterRcvry | NoDisasterRcvry | UnReSTricted])
```

The -PATH DialCONTrol parameter provides several options for setting the line.

The STAtic value allows the selected path to be statically bound to its corresponding port and is the default. The DYNamic setting unbinds a path from its corresponding port and adds the path to the dial pool. A static path is not part of the dial pool. Placing a path in the dial pool allows the path to be used by any dial port. For a dial path to become a dynamic dial path, the -PATH LineType parameter must be set to Dialup.

You can also set the line specifically for disaster recovery, or set it as unrestricted to allow it to be used for any purpose including disaster recovery. The NoDisasterRcvry option prevents the line from being used for disaster recovery and is usually assigned to the slowest or least reliable line on the network.

For example, to configure ISDN on path 2.1 as an unrestricted, dynamic line, enter:

```
SETDefault !2.1 -PATH LineType = Dialup
SETDefault !2.1 -PATH SwitchType = ATT5ESS
SETDefault !2.1 -PATH LocalDialNo = “1-213-555-1212”
SETDefault !2.1 -PATH LocalSubAddr = “100”
SETDefault !2.1 -PATH DialCONTrol = (DYNamic, UnReSTricted)
```

6 If you are configuring ISDN for North American BRI ISDN dial-up modes, specify the Service Profile Identifiers (SPIDs) using:

```
SETDefault !<connectorID> -PATH SPIDdn1 = “<string>”
```

Some North American ISDN switches require two SPIDs. In this case, you will need to add the SETDefault !<connectorID> -PATH SPIDdn2 = “<string>” parameter to your configuration. For DMS 100, the string must contain a Service Profile Identifier (SPID) and a directory number (DN) separated by a semicolon (;).

For example, to set the SPID to 4085551212 and the DN to 1234567, enter:

```
SETDefault !2.1 -PATH SPIDdn1 = “4085551212;1234567”
```

7 Enable the line and make sure all settings on the path take effect using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

8 If you changed the switch type or a SPID parameter, you must reboot the system for the changes take effect.

Configuring Leased LInes

You can use leased lines to establish connectivity with remote sites so that these sites can send updates to a central location.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.

Procedure

To configure a leased line to use with bandwidth management, follow these steps:

You must complete this procedure at both ends of the link.

1 Set the line type to Leased.

a By default, the dial path is set to Leased for all NETBuilder II systems. You can check that the setting has not been changed using:

```
SHow !<path> -PATH LineType
```

b To change the value of the parameter, use:

```
SETDefault !<path> -PATH LineType = Leased
```

2 Specify the baud rate for the device using:

```
SETDefault !<path> -PATH Baud = <kbps> (1.2-52000)
```

3 Select the connector type using:

```
SETDefault !<path> -PATH CONNector = V35 | RS232 | RS449 | G703 |
 HSSI | X21
```

4 Set the transmit clock for the bridge/router using:

```
SETDefault !<path> -PATH CLock = TestMode | External | Internal
```

The Internal value applies to model 32x and 52x SuperStack II bridge/routers only. The External value allows the bridge/router to derive the transmit clock from either the send or receive timing clock supplied by the digital service unit/channel service unit (DSU/CSU) or by the attached modem.

5 Enable the line and make sure all settings on the path take effect using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

Configuring System Bandwidth Management Mode (DOD)

This section describes the procedure to enable system bandwidth management mode.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.
- Enable your WAN communication resources according to “Configuring WAN Resources” on page 37-8.

Procedure

To enable system bandwidth management mode, set the initiator state to enable system bandwidth management mode using:

```
SETDefault !<port> -PORT DialInitState = DialOnDemand
```

This command also enables dial-on-demand.

Configuring Bandwidth-on-Demand

This section describes how to enable bandwidth-on-demand for the system bandwidth management mode. Refer to Figure 37-1 on page 37-8 for an illustration of the configuration example in this section.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.
- Enable your WAN communication resources according to “Configuring WAN Resources” on page 37-8.

Procedure

To enable bandwidth-on-demand (BOD), follow these steps:

1 Specify the amount of bandwidth the port should bring up when enabled using:

```
SETDefault !<port> -PORT NORMalBandwidth = <kbps>
```

2 Enable BOD and specify the amount of additional bandwidth that bandwidth management can allocate for a port using:

```
SETDefault !<port> -PORT BODIncrLimit = <kbps>
```

This syntax specifies incremental bandwidth levels that can be allocated for the port.

3 Specify the conditions that trigger additional path resources for BOD using:

```
SETDefault !<port> -PORT BODTHreshold = <%>(0–100)
```

4 Specify the amount of time bandwidth management should wait to take action to bring a port up or down using:

```
SETDefault !<port> -PORT DialSamplPeriod = <seconds>(0–300),
 (0–300)
```

For example, enter the following commands configure a DOD line with a normal port bandwidth of 64 kbps:

```
SETDefault !3 -PORT DialInitState = DialOnDemand
SETDefault !3 -PORT NORMalBandwidth = 64
SETDefault !3 -PORT BODIncrLimit = 64
SETDefault !3 -PORT BODTHreshold = 50
SETDefault !3 -PORT DialSamplPeriod = 30, 60
```

Traffic must exceed 32 kbps for 30 seconds before bandwidth management brings up additional bandwidth using BOD. The additional dial path is taken down when the rate of traffic is less than 32 kbps for longer than 60 seconds.

Configuring the Dial List

A dial list allows the software to select a phone number from a list of destination phone numbers associated with a port. Numbers in the dial list are selected sequentially. The software provides options for ordering the numbers in this list.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.
- Enable your WAN communication resources according to “Configuring WAN Resources” on page 37-8.

Procedure

To configure a list of telephone numbers to dial for remote WAN connections, and as possible resources for additional bandwidth allocation, follow these steps:

1 To allow your bridge/router to dial out, configure the dial number list using:

```
ADD !<port> -PORT DialNoList “<phone no>” [Baud = <rate>
 (1.2–16000)] [Type = Modem | Bri | Sw56 | WE | WEH0] | [Pos =
 <number>]
```

You can enter this command more than once to append a phone number or profile to the dial list.

The string entered for the WAN Extender profile is case-sensitive, can contain alphanumeric characters, and can be no longer than 52 characters.

With ISDN phone numbers, you can use hyphens (-) to separate the prefix from the country code from the phone number. For ISDN, the phone number includes a dial prefix, country code, and area code and possibly a subaddress. If you specify a subaddress, you must separate the phone number from the subaddress with a semicolon (;). You can configure up to 16 phone numbers per port.

For V.25 bis dialing, the phone number can include a dial prefix, country code, and area code.

The software uses the Baud and Type keywords to make a path match. It is important to enter the same device type as you entered in the -PATH ExDevType parameter; software will make a best match for the baud rate. Refer to "Configuring WAN Resources" on page 37-8 for use of these commands.

2 You can also specify the number of time the software attempts to redial the remote system if the call attempt fails using:

```
SETDefault !<port> -PORT DialRetryCount = <number> (0–20)
```

If dialing is based on a static port and path binding, the software first tries to make the call. If the attempt fails to bring the path up, the software tries the call again using the same or different path. The call attempts continue until the dial retry count is reached.

You can also append phone numbers to the end of the list, insert a phone number into a specific position in the list, edit an existing phone number, or delete an existing phone number. To add a phone number into a specific position in the list, refer to "Adding a Phone Number" on page 37-16.

Example 1 To enter a Los Angeles phone number for port 2 that consists of a long-distance dial prefix 1 (assume that the bridge/router being configured is located in Santa Clara), the phone number 213-456-7000, and the subaddress 101, enter:

```
ADD !2 -PORT DialNoList “1-213-456-7000;101” Baud = 56 Type = Bri
```

Example 2 The DialNoList parameter includes options for WAN Extender virtual paths. To add a dial number for virtual port V1 and instruct the system to go to port 4 with WAN Extender (WE) profile 5, enter:

```
ADD !V1 -PORT DialNoList "4 5" Type = WE
```

Example 3 To add a London phone number for port 2, at the end of a dial-up phone list, that consists of the international dialing code 011, the U.K. country code 44, and the phone number 213-456-7000, enter:

```
ADD !2 -PORT DialNoList “011 44 213 456 7000”
```

This entry ignores the Baud rate, Type, and Pos (position on the list).

For DTR dialing, the phone number is irrelevant, because the outgoing telephone number is stored in the modem.

Example 4 You can configure the dial number list to dial the same number repeatedly by adding multiple copies of the number. Prefix the phone number with a variable number of periods to distinguish the duplicate entries by entering:

```
ADD !V1 -PORT DialNoList “123 4567”
ADD !V1 -PORT DialNoList “.123 4567”
ADD !V1 -PORT DialNoList “..123 4567”
```

The bridge/router dials 123-4567 three times. This technique works for both ISDN and analog phone numbers.

Adding a Phone Number

To insert a phone number into a specific position in the dial number list, enter the Pos (Position) keyword with a non-zero number after the dial string.

For example, to insert a phone number for port 4 into position 2 of the dial number list that contains 10 phone numbers, enter:

```
ADD !4 -PORT DialNoList “510 555 7000” Pos = 2
```

The software inserts the new phone number into position 2. The phone number that was previously in position 2 is now in position 3. If the phone already exists in the dial number list, it will be moved to position 2. You can insert the same phone number twice by using blanks or other redundant characters. You also can include the Baud and Type keywords in any order when inserting phone numbers into the dial number list.

Editing an Existing Phone Number

To edit an existing phone number in the dial number list, you can change the position in the list, change the baud rate, and change the device type.

For example, if port 3 has already been assigned 612-345-3989 in position 2 with a baud rate of 64 kbps, you can change the baud rate by entering:

```
ADD !3 -PORT DialNoList “612 345 3989” Pos = 2 Baud = 14.4
```

Because the dial string is case-sensitive, make sure to match it exactly to successfully edit an existing string when characters other than numbers are used.

Deleting a Phone Number

To remove a phone number or profile from the dial number list, use:

```
DELete !<port> -PORT DialNoList “<phone no>”
```

The profile name is case-sensitive and must be matched exactly to be deleted.

Binding Paths to Ports

3Com software uses the concept of port and path bindings to pair a logical interface (port) to a physical network resource (path) such as an ISDN line. Software version 9.1 made it possible for a port to bind to multiple dial paths. For each port, the bandwidth can be dynamically allocated by bundling the multiple dial paths into the virtual pipe. This concept allows bandwidth to be allocated when and where it is most needed.

The software also allows you to create dynamic paths by unbinding static paths from their ports and saving them in a dial pool to be shared by more than one port. A path in the dial pool can be bound to a port when the path is needed for data transfer events associated with dial-up.

The procedures in this section illustrate how to create a dynamic path, then how to convert the dynamic path back to a static path.

Converting a Static Path to a Dynamic Path

Paths (except WAN Extender paths) are static by default. To convert a static path to a dynamic path, follow these steps:

1 Unbind a path from its corresponding port and add the port to the dial pool by entering:

```
SETDefault !1 -PATH DialCONTrol = DYNamic
```

2 Enable the path by entering:

```
SETDefault !1 -PATH CONTrol = Enabled
```

Changing a Dynamic Path to a Static Path

To change a dynamic path to a static path and remove it from the dial pool, follow these steps:

1 Convert a dynamic path to static by entering:

```
SETDefault !1 -PATH DialCONTrol = STAtic
```

2 Reassign a path or multiple paths to a port using

```
ADD !<port> -PORT PAths <path>
```

When a static dial path is added to a port, it is automatically inserted at the end of the path preference list; refer to the next section for further information about the path preference list.

3 Enable the path by entering:

```
SETDefault !1 -PATH CONTrol = Enabled
```

Configuring the Path Preference List

A path preference list reserves a path for use by a group of ports and sets the order of line use. Prioritization is accomplished by position in the path preference list. (Leased line paths cannot be included in the path preference list because bandwidth management cannot bring a leased line up or down; leased line paths are brought up when the port is enabled.) A path can be reserved by more than one port.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.
- Enable your WAN communication resources according to “Configuring WAN Resources” on page 37-8.

Procedure

To configure the path preference list, follow these steps:

1 Paths added to the path preference list must be a dial-up line. To specify path 1 as dial up, enter:

```
SETDefault !1 -PATH LineType = Dialup
```

2 Unbind a path from its assigned port and add it to the dial pool by entering:

```
SETDefault !1 -PATH DialCONTrol = DYNamic
```

The WAN Extender virtual paths eligible for path preferences are already dynamic; the line type for virtual paths cannot be changed.

3 Enable changes to the path by entering:

```
SETDefault !1 -PATH CONTrol = Enabled
```

In this command, use the <path> syntax if you are specifying a DTE serial path and use the <connectorID.channelID> syntax if you are specifying an ISDN path.

4 Enable the port or virtual port to use dial pool resources and map the remote system caller ID to a specific port.

a If you are using ports (as opposed to PPP virtual ports) with dynamic lines, use:

```
ADD !<port> -PORT PAths SCID“<SysCallerId>”
```

For example, to allow port 4 to use the dial pool for a path resource for outgoing calls and to map incoming calls with the caller ID of "London" to port 4, enter:

```
ADD !4 -PORT PAths SCID“London”
```

The string you enter for the caller ID is case-sensitive and can contain up to 31 characters.

b If you are using PPP virtual ports, use:

```
ADD !<port> -PORT VirtualPort {SCID“<SysCallerId>”}
```

For example, to create PPP virtual port V3 and allow it to use the dial pool for its path resources for outgoing calls and to map incoming calls with the caller ID of "NewYork" to virtual port V3, enter:

```
ADD !V3 -PORT VirtualPort SCID“NewYork”
```

The caller ID string is case-sensitive and can contain up to 31 characters.

Unlike Frame Relay and X.25 virtual ports, which are always associated with a particular path, PPP virtual ports can potentially use any path in the dial pool.

c Make sure that each remote site has been configured with a unique caller ID using:

```
SETDefault -SYS SysCallerID = “<string>”
```

The SysCallerID string is limited to 31 characters. The string should be administratively assigned and be unique across the network.

5 Reserve the paths and define their priority using:

```
ADD !<port> -PORT PathPreference [<path>] [,…] [Pos = <1- number>]
```

For example, to specify paths 2.1 and 2.2 for use by port 5, enter:

```
ADD !5 -PORT PathPreference 2.1, 2.2
```

6 Enable the previous port changes by entering:

```
SETDefault !1 -PORT CONTrol = Enabled
```

7 Check the current configuration of the path preference list by entering:

```
SHow -PATH DialPool
```

By default, the software adds dial paths to the end of the list if the position is not specified.

After paths are configured using the -PORT PathPreference parameter, no other ports can use the reserved paths except the designated ones. The software tries the preferred list of paths first before using path resources in the rest of the dial pool.

A dynamic path can appear in the path preference list for more than one port. A static path can only appear in the path preference list of the port to which it is bound.

You can append one or more dial-up paths to the end of the path preference list, insert one or more paths into a specific position in the list, or delete an existing path in the list. For more information, refer to the sections that follow.

Appending a Path

To append one or more dial paths to the end of the path preference list, use:

```
ADD !<port> -PORT PathPreference [<path>] [,…] [Pos = <1- number>]
```

For example, assume the path preference list for port 2 includes dial-up paths 3 and 2.1, and you want to append dial paths 5 and 6 to the end of the path preference list. Enter:

```
ADD !2 -PORT PathPreference 5, 6
```

After this command is executed, when port 2 needs a path resource, the software uses the preferred paths first. The order of their use is 3, 2.1, 5, and 6.

Adding a Path

To add one or more dial paths into a specific position in the path preference list, use the Pos (position) keyword with the desired position number. The paths are added into the list as follows:

- The software deletes any duplicate paths from the list.
- The software then adds the path list by inserting them starting at the specified position.

If you want to add more than one path, you must list the paths in the intended order.

For example, assume the path preference list for port 3 includes dial-up paths 3, 4, and 5, and you want to insert dial path 2.1 into position 2. Enter:

```
ADD !3 -PORT PathPreference 2.1 Pos = 2
```

After this command is executed, the path preference list has paths 3, 2.1, 4, and 5.

If you want to insert more than one dial path, you must list the paths in the intended order. For example, assume the path preference list for port 3 includes dial-up paths 3 and 5. To insert dial path 3 and 5 into position 2 and 3, enter:

```
ADD !3 -PORT PathPreference 3, 5 Pos = 2
```

After this command is executed, the path preference list is 2.1, 3, 5, and 4.

With this command, you can change the position of a path that already exists in the path preference list. For example, assume the path preference list for port 3 includes dial-up paths 4, 6, 7, and 5. To reposition path 6 into position 3, enter:

```
ADD !3 -PORT PathPreference 6 Pos = 3
```

The software inserts path 6 at position 3; path 7 that was originally in position 3 is now in position 2. If the position specified is larger than the existing list, the path is appended to the end of the list by default.

Deleting a Path

To remove one or more dial-up paths from the path preference list, use:

```
DELete !<port> -PORT PathPreference <path> [,...]
```

In this syntax, use <path> if you are specifying a DTE serial path. If you are specifying an ISDN path, substitute <connectorID.channelID> for <path>.

The paths to be deleted can be listed in any order. If you try to delete a path that does not exist in the list, an error message is displayed.

Configuring Manual Bandwidth Management Mode

With manual bandwidth management mode, you can control the connect sequence and bandwidth settings for a one-time call on a line. You use port-based dialing and the DIal command to manually dial on the specified port. The procedures in this section show how to configure manual bandwidth management mode and manually connect a line, and how to enable disaster recovery.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your local and wide area interfaces according to Chapter 1.
- Enable your WAN communication resources according to "Configuring WAN Resources" on page 37-8.

Procedure

To place a call under manual bandwidth management mode, follow these steps:

1 Enable system bandwidth management mode using:

```
SETDefault !<port> -PORT DialInitState = ManualDial
```

2 Specify the amount of bandwidth that bandwidth management should bring up when this port is enabled using:

```
SET !<port> -PORT NORMalBandwidth = <kbps>
```

3 Place the call using:

```
DIal !<port> [-PORT] ["<dial-string>"]
```

This command accepts a static or dynamic port number. If you enter a telephone number in the optional dial string, it must also be listed in the dial number list; however, you can enter a port and telephone number to temporarily override entries in the dial number list. Refer to "Configuring the Dial List" on page 37-14 for more information about setting up a dial list.

4 Add the telephone number to dial using:

```
ADD !<port> -PORT DialNoList "<phone no>" [Baud = <rate>
 (1.2–16000)]
[Type = Modem | Bri | Sw56 | WE | WEH0] | [Pos = <number>]
```

5 Disconnect the call using:

```
HangUp !<port> [-PORT]
```

HangUp brings down all dial path resources unless there are leased lines active.

Disaster Recovery Procedure

To configure disaster recovery using manual bandwidth management mode, follow these steps:

1 Enable manual bandwidth management by entering:

```
SETDefault !4 -PORT DialInitState = ManualDial
```

2 Enable disaster recovery on the port by entering.

```
SETDefault !4 -PORT DialCONTrol = DisasterRcvry
```

3 Specify the normal bandwidth for this port using:

```
SET !<port> -PORT NORMalBandwidth = <kbps>
```

When you enable disaster recovery, you configure the software to bring up bandwidth to meet the target when a leased line goes down and bandwidth on the port falls below that specified with the -PORT NORMalBandwidth parameter.

Verifying the Configuration

To verify the configuration, follow these steps:

1 Display dial-up configuration information using:

```
SHow [!<port> | !*] -PORT DialCONFig
```

This display shows the port state (up or down), port function (disaster recovery or BOD), the paths that are in use, the path state, and the dial string for active outgoing calls. The path in use can be a static path or a dynamic path from the dial pool. If the path is from the dial pool, the information is displayed similar to a static path.

The DialCONFig parameter display identifies a WAN Extender virtual path as Dialup or Leased on the Dial Ctrl list if the port to which the virtual path is assigned is up; if the port is not up it displays a hyphen (-).

2 Display path configurations by entering:

```
SHow -PATH CONFiguration
```

Verify that the paths are enabled and their status is up.

The CONFiguration parameter displays the Baud, Conn, and Line type for WAN Extender virtual paths only if the port to which they are connected is up. If the port is down, it displays a hyphen (-).

3 Display port configurations by entering:

```
SHow -PORT CONFiguration
```

Verify that the ports are enabled and their status is up. Also make sure that dynamic ports are selecting path resources from the dial pool.

The SHow -PORT CONFiguration parameter displays the SysCallerId (for example, SCID “Boston”) for dial-up and leased WAN Extender virtual paths if the port to which they are connected is up. It also shows the aggregate bandwidth of the port. if the port is down, it displays a hyphen (-).

Troubleshooting the Configuration

To troubleshoot the configuration, follow these steps:

1 Display the status of lines under bandwidth management control using:

```
SHow [!<port> | !*] -PORT DialSTatus
```

The display shows the state of the ports under bandwidth management. Displays include total port bandwidth utilization, and messages indicating congestion levels and the intentions of the bandwidth manager for allocating additional resources. If the port is to be used for an outgoing call, the dial string (phone number) is displayed.

The DialSTatus parameter displays path number, B channel, and network port for WAN Extender virtual paths only if the port to which they are connected is up. if the port is down, it displays a hyphen (-).

2 Display a time-stamped dial history for the specified port or for all ports using:

```
SHow [!<port> | !*] -PORT DialHistory
```

Configuration Examples

This section provides configuration examples that you can use to help you configure a wide-area network on your bridge/router using port bandwidth management.

Load Balancing over Multiple Dial-up Links

Load balancing equalizes traffic flow and makes sure that packets that may have been fragmented over the links arrive at their destination in the correct sequence. Load balancing is accomplished using the PPP Multilink Protocol (MLP) as described in RFC 1717. The -PPP MlpCONTrol parameter enables this protocol.

The example configuration depicted in Figure 37-2 uses system bandwidth management mode. Bandwidth is set to 64 kbps and BOD is triggered on when traffic exceeds 32 kbps; bandwidth management will add paths to increase bandwidth to the 128 kbps limit. The paths are taken down once traffic returns to 32 kbps or less for longer than 60 seconds.

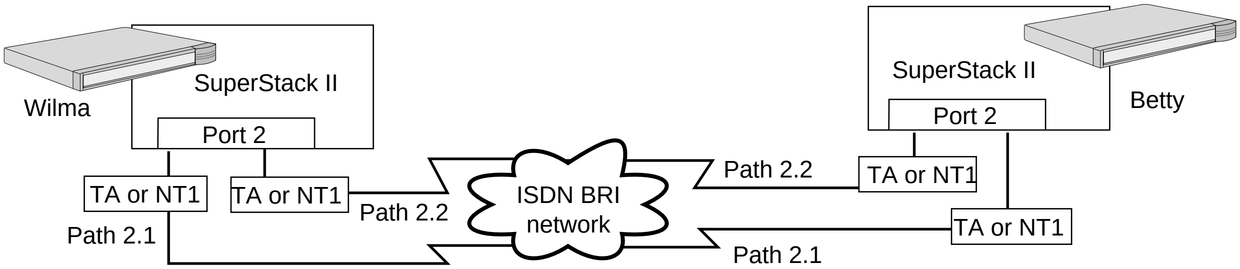


Figure 37-2 PPP MLP Load Balancing

The following example configuration enables load balancing over two ISDN BRI channels. The parameters must be configured on both bridge/routers.

```
SETDefault !2 -POrt name="ToBetty"
SETDefault !2.1 -PAth LineType=Dialup
SETDefault !2.2 -PAth LineType=Dialup
ADD !ToBetty -POrt DialNoList "4085551212" t=bri baud=64
ADD !ToBetty -POrt DialNoList "4085551313" t=bri baud=64
SETDefault !MLPPath1 -PAth LocalDialNo="1234567"
SETDefault !MLPPath2 -PAth LocalDialNo="2345678"
SETDefault !MLPPath1 -PAth DialCONTrol=(STAtic,UnReSTricted)
SETDefault !MLPPath2 -PAth DialCONTrol=(STAtic,UnReSTricted)
ADD !ToBetty -POrt PAth 2.1,2.2
SETDefault !2.1 -PAth CONTrol=Enabled
SETDefault !2.2 -PAth CONTrol=Enabled
SETDefault !ToBetty  -POrt DialInitState=DialOnDemand
SETDefault !ToBetty  -POrt NORMalBandwidth=64
SETDefault !ToBetty  -POrt BODIncrLimiT=64
SETDefault !ToBetty  -POrt BODTHreshold=50
SETDefault !ToBetty  -POrt DialSamplPeriod=0,30
SETDefault !ToBetty  -POrt DialIdleTime=300
SETDefault !ToBetty  -ppp  MlpCONTrol=Enabled
SETDefault !ToBetty  -POrt CONTrol=Enabled
```

You can verify that both paths are up and that MLP is enabled by entering:

```
SHow -PATH CONFiguration
SHow -PPP STATUS
SHow -PPP MlpSTATIstics
```

NETBuilder II WAN Extender Configuration Example

To configure the NETBuilder II bridge/router to use the WAN Extender virtual paths, you must set the owner of the port that corresponds to the bridge/router connection to WAN Extender. Set the baud rate on that port to 4096.

Virtual ports must be created to represent the logical attachment between the sites. Specify the mapping between the virtual port created as a representation of the logical attachment to the remote site and the WAN Extender profile that describes the basic physical connection in the dial number list. This must be done for each profile.

In Figure 37-3, two sites with NETBuilder II bridge/routers and WAN Extenders are to be configured with 256 kbps base bandwidth, and the remaining 19 B channels of the PRI line are to be made available for BOD.

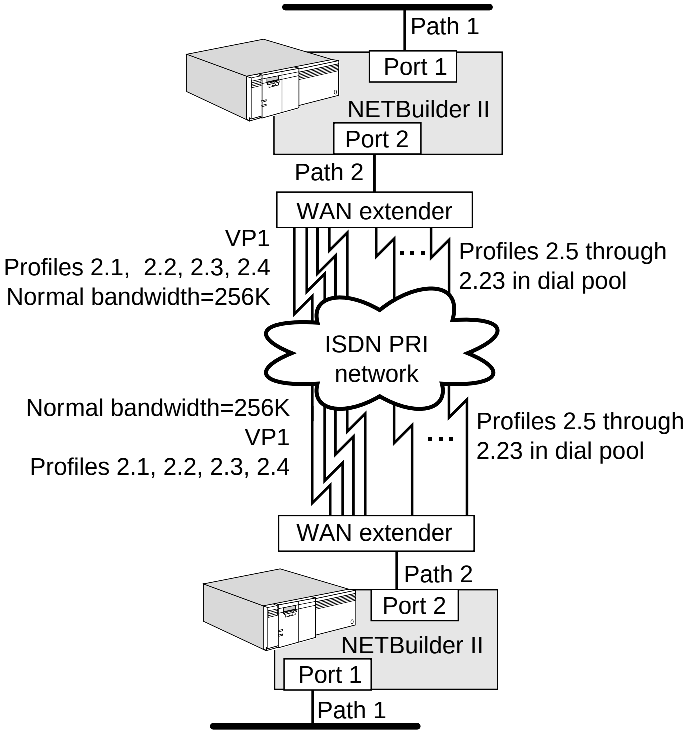


Figure 37-3 WAN Extender Configuration Example

The first requirement is that 23 profiles be set up on the WAN Extender to enable calls between the two bridge/routers. These profiles describe virtual paths in a dial pool for the NETBuilder II bridge/router to use for port 2.

The following sample configuration shows how to set the baud rate, create the virtual ports, and configure the WAN Extender profiles.

```
SETDefault !2 -POrt OWner = WanExtender
SETDefault !2 -PAth BAud = 4096
ADD !V1 -Port VirtualPort WanExtender “SystemCallerID”
ADD !V1 -Port DialNoList “2 1” Type=WE
ADD !V1 -Port DialNoList “2 2” Type=WE
ADD !V1 -Port DialNoList “2 3” Type=WE
ADD !V1 -Port DialNoList “2 4” Type=WE
    .
    .
    .
ADD !V1 -Port DialNoList “2 23”
```

The base bandwidth on the NETBuilder II bridge/routers are set to 256 kbps using the NORMalBandwidth parameter. The NETBuilder II bridge/router will dial up four B channels and keep them up at all times as a minimum bandwidth for the port. The upper bandwidth limit is set with the BODIncrLimit parameter. Setting this parameter to 1472 allows the full PRI line to be used if needed by the port. Setting the BODIncrLimit parameter to a level greater than zero enables BOD.

Two parameters control when the additional channels are to be dialed or hung up. The BODTHreshold parameter sets the trigger point stated as a percentage of the current active bandwidth. The DialSamplPeriod parameter sets how long the data rate has to remain over or under the BODTHreshold limit before a resource is dialed. In this case, the threshold is 75 percent and the sample period is 5 seconds.

These parameters configure a minimum 256 kbps pipe that is available all of the time. If the data rate through the pipe exceeds 192 kbps for 5 seconds, another 64 kbps channel will be added. If the data rate then exceeds 240 kbps for 5 seconds, another 64 kbps channel will be added and so on until all 23 channels are up for the port to use as long as the data rate exceeds 75 percent of the current bandwidth. If the data rate drops below 75 percent of the current bandwidth of 384 kbps (288 kbps) for 5 seconds, one of the B channels is dropped, and this continues as long as the data rate is less than 75 percent of the current bandwidth for 5-second intervals, down to the normal bandwidth of 256 kbps.

The following example configuration shows how to configure the bandwidth settings on the NETBuilder II bridge/routers.

```
SETDefault !V1 -POrt DialInitState = DialOnDemand
SETDefault !V1 -POrt NORMalBandwidth = 256
SETDefault !V1 -POrt BODIncrLimit = 1472
SETDefault !V1 -POrt BODTHreshold = 75%
SETDefault !V1 -POrt DialSamplPeriod = 5, 10
```

Routing Configurations over DOD Links

This section describes how to configure the network layer protocols supported with bandwidth management.

IP over a DOD Link

To configure IP over a DOD link, follow these steps. Figure 37-4 shows the IP network referenced in the following IP and RIPIP procedures.

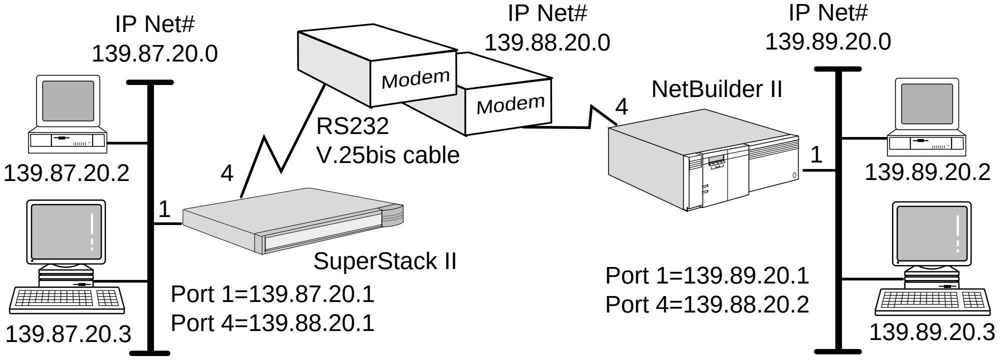


Figure 37-4 IP Network Design Example

1 Configure the network address for port 1 and port 4 on the SuperStack II bridge/router by entering:

```
SETDefault !1 -IP NETaddr = 139.87.20.1
SETDefault !4 -IP NETaddr = 139.88.20.1
```

2 Configure the network address for port 1 and port 4 on the NETBuilder II bridge/router by entering:

```
SETDefault !1 -IP NETaddr = 139.89.20.1
SETDefault !4 -IP NETaddr = 139.88.20.2
```

3 Enable IP routing by entering the following command on the SuperStack II and NETBuilder II bridge/routers:

```
SETDefault -IP CONTrol = ROute
```

4 Add a static route on the SuperStack II bridge/router by entering:

```
ADD -IP ROUte 139.89.0.0 139.88.20.2 1
```

5 Add a static route on the NETBuilder II system by entering:

```
ADD -IP ROUte 139.87.0.0 139.88.20.1 1
```

6 Check transport and network layer status by entering:

```
SHow -IP NETaddr
```

RIPIP over a DOD Link

To configure DOD on a RIPIP network, follow these steps:

1 On the SuperStack II bridge/router, configure the LAN ports to send and receive update packets by entering:

```
SETDefault !1 -RIPIP CONTrol = (TAlk, Listen)
```

2 On the SuperStack II bridge/router, configure the WAN ports to not run the RIPIP Protocol, but instead take advantage of the static routes you set up using IP by entering:

```
SETDefault !4 -RIPIP CONTrol = (NoTAlk, NoListen)
```

For more information on static routes, refer to "IP over a DOD Link" on page 37-24.

3 On the NETBuilder II system, configure the LAN ports to send and receive update packets by entering:

```
SETDefault !1 -RIPIP CONTrol = (TAlk, Listen)
```

4 On the NETBuilder II system, configure the WAN ports to not TAlk and not Listen by entering:

```
SETDefault !4 -RIPIP CONTrol = (NoTAlk, NoListen)
```

5 Advertise static policies by entering the following command on the SuperStack II and NETBuilder II bridge/routers:

```
ADD !1 -RIPIP StaticPolicy All
```

TCP for SNA Traffic over a DOD Link

To use the recommended TCP protocol settings to enable Systems Network Architecture (SNA) traffic, including data link switching (DLSw), to be sent over an ISDN DOD link, follow these steps:

1 On the NETBuilder II or SuperStack II bridge/routers, disable TCP keepalive packets by entering:

```
SETDefault -TCP CONTrol = NoKeepAlive
```

2 Set the TCP retransmit limit to either 3 or 4 using:

```
SETDefault -TCP RetransmitLimit = <retrys> (0-128)
```

The limit of either 3 or 4 is specifically recommended for SNA configurations over DOD.

3 Set the time in seconds before a DOD line is disconnected using:

```
SETDefault !<port> -PORT DialIdleTime = <seconds> (0–3600)
```

The default is three minutes (180 seconds).

4 Set the DOD retry count on the port to 9 using:

```
SETDefault !<port> DialRetryCount = 9
```

5 Set the DOD retry timer on the port to 15 seconds using:

```
SETDefault !<port> -PORT DialRetryTime = 15
```

IPX with Incremental Broadcasts over a DOD Link

Figure 37-5 shows the IPX network referenced in the following procedure. Refer to Chapter 13 for more information on IPX routing.

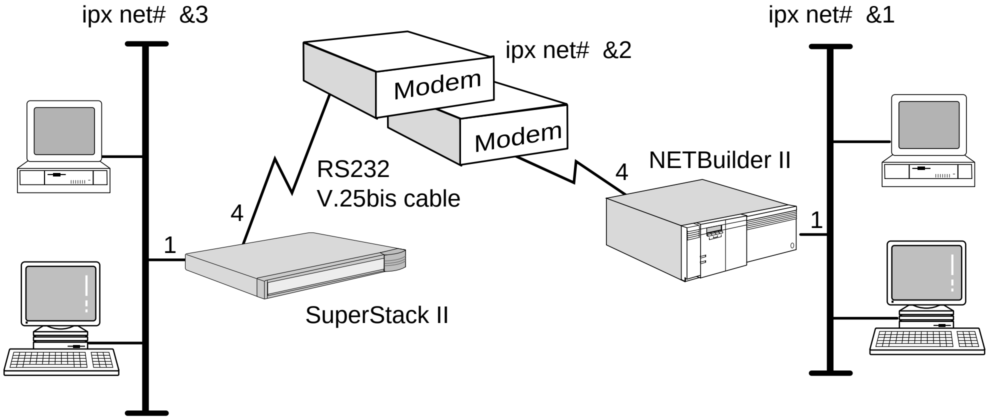


Figure 37-5 IPX Network Design Example

To configure DOD with incremental broadcasts on an IPX network, follow these steps:

1 Set the network number of port 1 and port 4 on the SuperStack II bridge/router by entering:

```
SETDefault !1 -IPX NETnumber = &3
SETDefault !4 -IPX NETnumber = &2
```

2 Set the network number for port 1 and port 4 on the NETBuilder II bridge/router by entering:

```
SETDefault !1 -IPX NETnumber = &1
SETDefault !4 -IPX NETnumber = &2
```

3 Enable IPX routing on both routers by entering:

```
SETDefault !1 -IPX CONTrol = ROute
SETDefault !4 -IPX CONTrol = ROute
```

4 Disable WAN broadcasts on LAN ports by entering:

```
SETDefault !1 -IPX CONTrol = NoIpxWan
```

5 Change NRIP updates on port 4 of both routers from periodic broadcast to incremental broadcast by entering:

```
SETDefault !4 -NRIP CONTrol = NoPEriodic
```

6 Change SAP updates on port 4 of both routers from periodic broadcast to incremental broadcast by entering:

```
SETDefault !4 -SAP CONTrol = NoPEriodic
```

Refer to Chapter 13 for more information on RIP/SAP updates.

IPX Protocol in a Boundary Routing Environment over a DOD Link

To help you configure DOD with the IPX Protocol in a Boundary Routing environment, two configuration examples are provided.

Example 1

Port 3 of the NETBuilder II bridge/router at the central site is linked with port 3 of the SuperStack II boundary router peripheral node. NetWare clients but no NetWare servers exist on the peripheral network. NetWare clients attach across the DOD link back to the NetWare servers on the central site.

To configure IPX in a Boundary Routing environment, follow these steps:

1 Enable Boundary Routing on port 3 of the NETBuilder II bridge/router by entering:

```
SETDefault !3 -BCN CONTrol = Enabled
```

2 Enable IPX routing on the NETBuilder II bridge/router by entering:

```
SETDefault !3 -IPX CONTrol = ROute
```

3 Assign the IPX network number on port 3 of the NETBuilder II bridge/router by entering:

```
SETDefault !3 -IPX NETnumber = &2001
```

4 Use incremental NRIP and SAP to reduce broadcast traffic on the DOD link by setting port 3 on the NETBuilder II bridge/router by entering:

```
SETDefault !3 -NRIP CONTrol = (NoPEriodic)
SETDefault !3 -SAP CONTrol = (NoPEriodic)
```

5 Verify that spoofing of NetWare Core Protocol (NCP) keep alive packets using the WatchDog mechanism is enabled on the NETBuilder II bridge/router by entering:

```
SHow !3 -IPX SPoofCONTrol
```

If spoofing has been disabled (NoNcpWatchDog), enable it by entering:

```
SETDefault !3 -IPX SPoofCONTrol = NcpWatchDog
```

Refer to “Configuring IPX Spoofing over a DOD Link” on page 13-27 for more information about NCP spoofing.

In this example, because there are no NetWare servers at the remote site, NetWare clients log on the servers at the central site. Consequently, there is periodic traffic of NCP keep alive packets between the servers and clients in order to maintain these NCP connections. For software releases before 8.0, it is recommended that the Delay Before the First WatchDog Packet parameter on the NetWare server be adjusted to a higher level to reduce traffic on the DOD link. For software version 8.0 and later, you can use the -IPX SPoofCONTrol parameter.

Example 2

Port 3 of the NETBuilder II bridge/router at the central site is linked with port 3 of the SuperStack II boundary router peripheral node. NetWare servers and NetWare clients exist on the peripheral network. On the peripheral network, NetWare clients log on to the remote servers as their primary servers and only attach across the DOD link to the central site servers periodically whenever their application needs dictate, for example, reading electronic mail.

To specify this configuration, follow these steps:

1 Enable Boundary Routing and smart filtering on port 3 of the NETBuilder II bridge/router by entering:

```
SETDefault !3 -BCN CONTrol = (Enabled, SmartFiltering)
```

2 Enable IPX routing on the NETBuilder II bridge/router by entering:

```
SETDefault !3 -IPX CONTrol = ROute
```

3 Assign the IPX network number on port 3 of the NETBuilder II bridge/router by entering:

```
SETDefault !3 -IPX NETnumber = &2001
```

4 Use periodic NRIP and SAP in conjunction with smart filtering by setting port 3 on the NETBuilder II bridge/router by entering:

```
SETDefault !3 -NRIP CONTrol = (Talk, Listen, PEriodic)
SETDefault !3 -SAP CONTrol = (Talk, Listen, PEriodic)
```

In this example, with servers on the remote sites, NetWare clients should log on to these servers as their main servers, and occasionally log on (attach) to the servers at the central site. To reduce the NCP keepalive packets across the DOD link, you can set user guidelines to request that users only maintain their login to a central site server when their application needs it.

Summary of Bandwidth Manager Commands and Parameters

Table 37-4 summarizes the commands and parameters that are used with the port bandwidth management.

Table 37-4 Bandwidth Management Tasks and Commands

Task	Command or Parameter	Description	Applies to
Configure a path	-PATH LineType	Sets the type of line being used.	DTE, ISDN, WE
	-PATH CONNector	Specifies the type of connector for a serial interface.	DTE
	-PATH DialMode	Configures V.25 bis standard dialing or dialing from a DTR modem.	DTE
	-PATH ExDevType	Specifies the device type attached to the DTE path.	DTE, ISDN
	-PATH SwitchType	Specifies the type of ISDN switch to which an ISDN path is connected.	ISDN
	-PATH LocalDialNo	Associates a phone number to your ISDN path.	ISDN on model 42x and 52x SuperStack II
	-PATH LocalSubAddr	Specifies a subaddress to the phone number you specified for your ISDN path.	ISDN on model 42x and 52x SuperStack II

(continued)

Table 37-4 Bandwidth Management Tasks and Commands (continued)

Task	Command or Parameter	Description	Applies to
	-PATH SPIDdn1 and SPIDdn2	Specifies the Service Profile Identifiers (SPIDs) and directory numbers (DNs) for North American BRI ISDN dial-up modes.	ISDN on model 42x and 52x SuperStack II
	-PATH RateAdaption	Specifies a method that determines the data rate to be used on an ISDN path.	ISDN on model 42x and 52x SuperStack II
Enable manual bandwidth management mode (manual dial mode)	-PORT DialInitState	ManualDial option allows you to manually dial call. You specify bandwidth settings rather than let bandwidth management monitor the line and adjust settings as needed.	DTE, ISDN, WE
Connect	DIal	Manually connects a dial-up path or port.	DTE, ISDN, WE
	-PORT AutoDial	Connects the dial-up path assigned to a port as soon as the path is enabled.	DTE, ISDN, WE
Disconnect	HangUp	Manually disconnects a dial-up path or port.	DTE, ISDN, WE
	-PATH DialCarrierTime	Defines the number of seconds the system must wait for carrier signals on the line that has connected.	DTE, ISDN, WE
	-PORT DialIdleTime	Sets the idle timer in seconds for a dial-up line before the line is disconnected if it is not in use.	DTE, ISDN, WE
Retry a dial-up connection	-PORT DialRetryCount	Specifies the number of times to retry the call if the call attempt fails.	DTE, ISDN, WE
	-PORT DialRetryTime	Sets the initial value in seconds to wait before attempting to reconnect after a connection has failed because the carrier was not detected or for any other reason that the path did not come up.	DTE, ISDN, WE
Configure port attributes for answer-only line	-PORT DialRcvrState	Sets the call receiver dial control state.	DTE, ISDN, WE
	-PORT DialInitState	NoDialOut option prevents outgoing calls when -PORT DialRcvrState is set to Answer.	DTE, ISDN, WE
Enable system bandwidth management mode (dial-on-demand)	-PORT DialInitState	DialOnDemand option enables system bandwidth management for bandwidth-on-demand and dial-on-demand modes, and monitors the line; additional lines are brought up or down based on traffic demand.	DTE, ISDN, WE
Configure for bandwidth-on-demand	-PATH DialCONTrol	Assigns the dial path unrestricted use as an additional resource for adding bandwidth.	DTE, ISDN
	-PORT NORMalBandwidth	Specifies the port bandwidth setting.	DTE, ISDN, WE
	-PORT BODTHreshold	Configures the threshold that triggers the BOD line up and down.	DTE, ISDN, WE
	-PORT BODIncrLimit	Configures the maximum incremental bandwidth that can be allocated using BOD.	DTE, ISDN, WE
	-PORT DialSamplPeriod	Sets the time to sample threshold conditions before taking an action to bring a path up or down, based on packet traffic load for BOD.	DTE, ISDN, WE
Configure disaster recovery	-PORT DialCONTrol	Restricts the dial path for use as disaster recovery only.	DTE, ISDN, WE
Configure a port to use dynamic dial path resources	-PORT PAths	Assigns dial pool resources to a specified port and identifies the remote system caller ID. For dial-up lines of any kind, the remote caller ID is a text string (like a city name).	DTE, ISDN, WE
	-PORT VirtualPort	Creates a virtual port that uses path resources from the dial pool and identifies the remote system caller ID. For dial-up of any kind, the remote caller ID is a text string. A telephone number or text string can be used as the remote site ID for ISDN remote sites only.	DTE, ISDN, WE

(continued)

Table 37-4 Bandwidth Management Tasks and Commands (continued)

Task	Command or Parameter	Description	Applies to
Configure path attributes for a dial-up path	-PATH DialCONTrol	Sets the path attributes for the static and dynamic dial-up paths.	DTE, ISDN
	-PORT PathPreference	Configures the dial path usage preference.	DTE, ISDN
Display port and path status	-PORT DialSTatus	Provides the status of the dial-up service and the state of bandwidth management for the specified dial ports.	DTE, ISDN, WE
Display port dial history	-PORT DialHistory	Provides the dial history for the specified port.	DTE, ISDN, WE
Display the dial pool status	-PATH DialPool	Provides the status and configuration of the paths in the dial pool.	DTE, ISDN, WE

Bandwidth Management Concepts

This section explains the concept of bandwidth management and lists the resources the bandwidth management feature manages. Before proceeding, you need to be familiar with the concepts of ports and paths as described in Chapter 1 of this guide. A glossary of terms used in this chapter is provided at the end of the chapter.

Virtual Pipe

Software version 9.1 introduced the concept of port *bandwidth management*, which is a process that applies static bandwidth, dynamic bandwidth, or a combination of these to provide a port with the bandwidth it needs to meet current requirements. At each port, a set of serial path resources are configured to provide a bandwidth bundle called a *virtual pipe*.

Bandwidth

Static bandwidth is provided by a configuration of one or more leased lines or dial paths to a port. The static resources are dedicated to a single port. Leased lines can provide continuous dedicated bandwidth to the port. Static dial paths can also provide incremental bandwidth, or bandwidth that becomes available only when a decision is made to dial them up.

Dynamic bandwidth is provided by dial-up line paths, which are allocated from a *dial pool*. Incremental bandwidth is provided by dial paths. The port can use either analog or digital lines that are allocated to it, and additional dial path resources can be added incrementally.

Bandwidth management operates on a port-by-port basis. It monitors line use based on rate of traffic and increases or decreases bandwidth based on limits that you specify. Network protocols that use the port are unaware of the underlying physical links, which bandwidth management bundles together into the virtual pipe to meet the port bandwidth requirements.

Bandwidth Aggregation

The main function of bandwidth management is to determine the aggregate bandwidth that will be provided to the set of protocols passing through the port. However, a WAN operates most efficiently when it can allow for variations in the type and amount of traffic passing through it. In addition to bandwidth management, the software provides the protocol reservation feature, which allocates portions of the virtual pipe to specified traffic such as the Internet Protocol (IP) or AppleTalk. As traffic passes through the pipe, the Point-to-Point (PPP) Multilink Protocol (MLP) can also be enabled to distribute packets more evenly over the virtual pipe. Figure 37-6 illustrates these concepts.

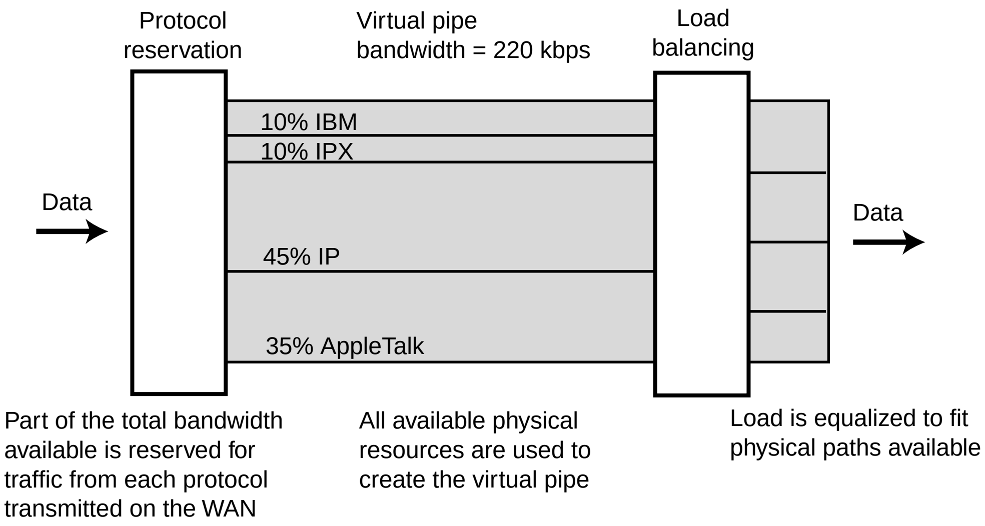

Protocol reservation
Virtual pipe bandwidth = 220 kbps
Load balancing
10% IBM
10% IPX
Data
45% IP
35% AppleTalk
Data
Part of the total bandwidth available is reserved for traffic from each protocol transmitted on the WAN
All available physical resources are used to create the virtual pipe
Load is equalized to fit physical paths available

Figure 37-6 Use of Resources through the Virtual Pipe

You reserve bandwidth for the protocols traversing the WAN using the -PORT ADD ProtocolRsrv parameter; refer to Chapter 4 in this guide for further explanation.

Load balancing equalizes traffic flow and makes sure that packets that may have been fragmented over the links arrive at their destination in the correct sequence. Load balancing is accomplished using the PPP Multilink Protocol as described in RFC 1717 and is enabled using the -PPP MlpCONTrol parameter.

Bandwidth Management Terms

The following terms are used in this chapter to explain concepts such as dial pools and the ports that can use them, and the bandwidth management strategies.

bandwidth management

A process that applies static bandwidth, dynamic bandwidth, or a combination of these to provide a port with the bandwidth it needs to meet current requirements. See also *virtual pipe*.

dial pool

The pool of dial paths that can be dynamically bound to any properly configured port.

disaster recovery threshold

The minimum of the normal bandwidth threshold and the total amount of configured leased line bandwidth that is assigned to the port. See also *normal bandwidth threshold*.

dynamic binding

The association of a path in the dial pool to a port when it is needed.

dynamic path

A path that can be used by more than one port. You create a dynamic path by unbinding it from its port. A dynamic path is stored in the dial pool. Characteristics of dynamic paths are as follows:

- Initially, dynamic paths are not bound to any port, but dynamically bind to make an outgoing call. To receive an incoming call, a dynamic path receives the call while still in the dial pool, and then is bound to a port.

- Dynamic paths can bind to different ports without user action, but only one port can bind at a time.
- Once the path becomes inactive, it unbinds from the port and becomes available for other ports.

normal bandwidth threshold
Bandwidth threshold defined by the -PORT NORMalBandwidth, BODTHreshold, BODIncrLimit, and DialSamplPeriod parameters.

port
A port is a logical interface used by the software to represent a connection to a network.

When the DOD path is up, the bridge/router routes the packets as expected in the normal NCP and SPX1 connection processes.

static binding
The association or binding of a path to a port as defined at system initialization time or by user configuration.

static path
A path is assigned (bound) to one port and can be used only by one port; a static path cannot be shared. By default, all paths are static at system initialization time.

virtual path
A path used by the NETBuilder II bridge/router to represent multiple logical paths multiplexed over a single interface. ISDN B, ISDN PRI, and DS0 channels delivered by channelized T1/E1 or switched-56 are presented as distinct virtual paths. A virtual path can be used as a static or dynamic resource.

virtual pipe
A term that describes a port of variable bandwidth.

virtual port
A port that is not associated with a physical interface. Virtual ports allow configuration of multiple destinations through a single interface.

38 CONFIGURING PROTOCOL RESERVATION

The protocol reservation feature enables you to assign a percentage of bandwidth to designated packets transmitting out of a WAN port that meet certain conditions.

This chapter describes how to configure protocol reservation for the following bridged- and routed-protocol packet types:

- IP-routed packets
- IPX-routed packets and all bridged packets
- NETBuilder-supported IBM traffic types, including DLSw (endpoint), LLC2 (for both SNA and NetBIOS) and APPN-routed packets

The reservation of bandwidth for packets transmitting over X.25 is not supported by protocol reservation. Refer to Chapter 45 for bandwidth management solutions for packets using X.25.

Protocol reservation only affects traffic being transmitted from the local bridge/router (the transmit direction). You cannot configure protocol reservation for traffic being received by the local bridge/router (the receive direction). If you want protocol reservation for traffic in both directions, then you must configure protocol reservation on both bridge/routers (the local and remote) that are sending traffic to each other.

Protocol reservation can be set for all WAN ports on the bridge/router or for specified WAN ports. However, configuring protocol reservation for specific ports is not recommended because it can affect network performance.

For conceptual information, refer to "How Protocol Reservation Works" on page 38-18.

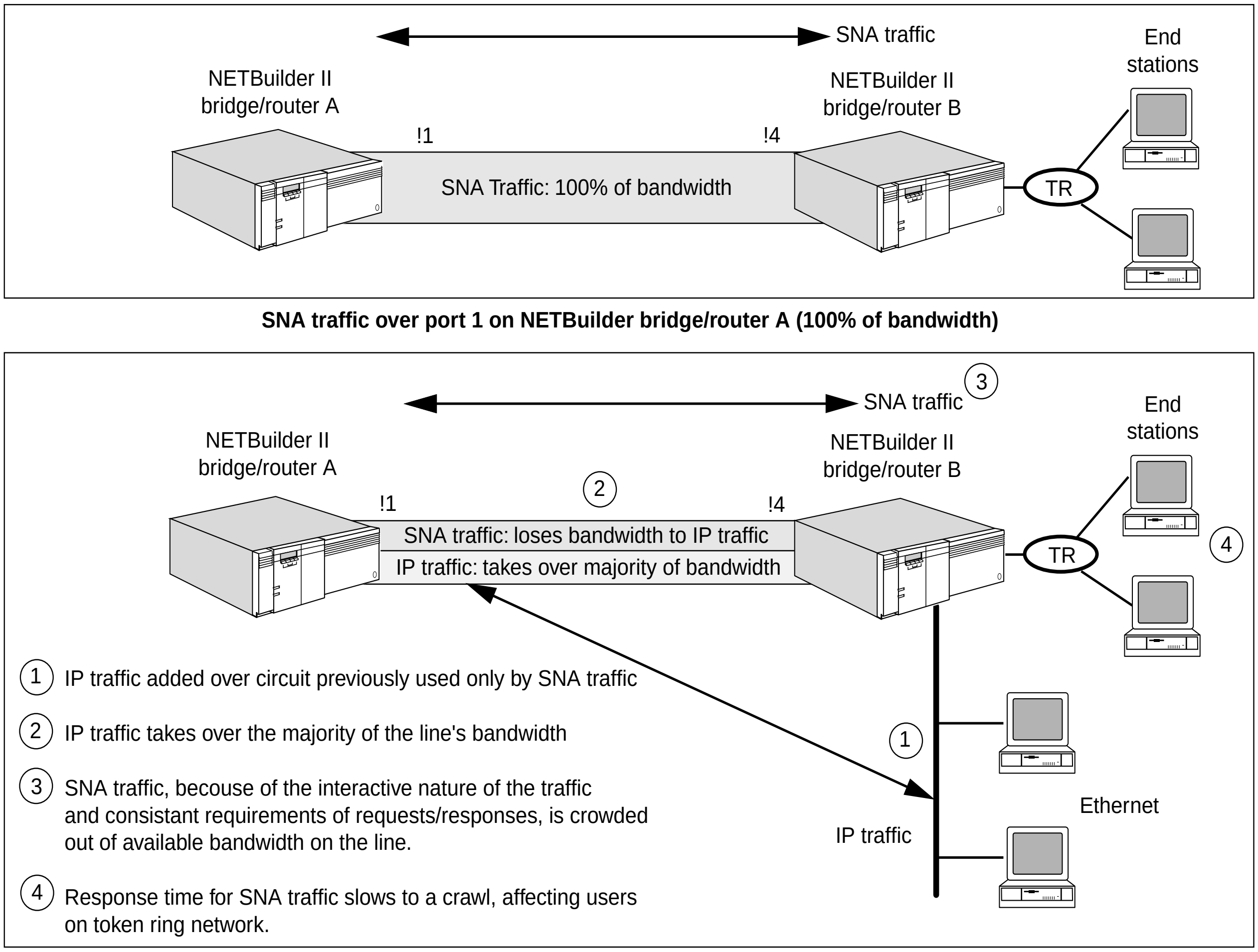
Why Use Protocol Reservation

Protocol reservation enables you to reserve bandwidth for lower bandwidth usage, interactive, response-time-sensitive, or transaction-oriented network applications, which are normally crowded out by heavy bandwidth usage applications such as file transfer or mail.

For example, in a multiprotocol environment that includes IBM protocol traffic (such as response-time-sensitive and mission-critical SNA packets) mixed in with other protocol traffic (such as IP or IPX), SNA devices throttle back the data transmission rate to the end station when they sense available bandwidth decreasing. If other network protocols increase this bandwidth consumption, SNA devices will throttle back the data transmission rate more, which slows the response time of SNA packets even more.

To avoid this situation, use protocol reservation to provide a percentage of bandwidth for the SNA packets and to restrict the percentage of bandwidth to the other more aggressive protocol packets. This will ensure that the small, response-time-sensitive SNA packets can pass through the port in a timely manner.

Figure 38-1 shows this situation. In the first diagram, only interactive SNA traffic is travelling over the port, using up the available bandwidth that provides the end user adequate response time. However, if you decide to also send IP packet traffic over that same port (as shown in the second diagram) then the IP packets continually use as much bandwidth as possible until the SNA traffic is “crowded out” of the bandwidth, which greatly reduces the response time of the SNA devices on the network. This crowding out is due to the connectionless nature of how IP works versus the connection-oriented nature of SNA interactive traffic.


SNA traffic
End stations
NETBuilder II bridge/router A
NETBuilder II bridge/router B
!1
!4
SNA Traffic: 100% of bandwidth
TR

SNA traffic over port 1 on NETBuilder bridge/router A (100% of bandwidth)

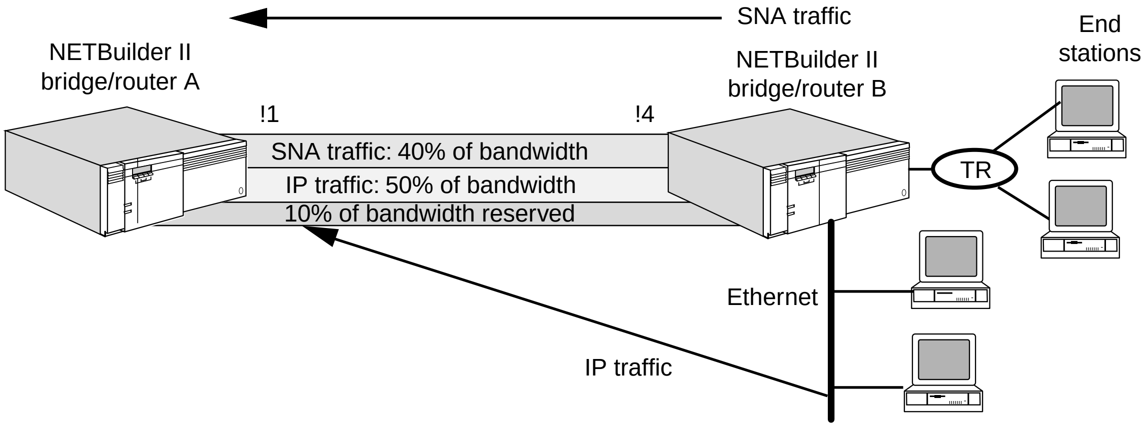
3
SNA traffic
End stations
NETBuilder II bridge/router A
NETBuilder II bridge/router B
2
!1
!4
SNA traffic: loses bandwidth to IP traffic
IP traffic: takes over majority of bandwidth
TR
4
1
Ethernet
IP traffic
1 IP traffic added over circuit previously used only by SNA traffic
2 IP traffic takes over the majority of the line's bandwidth
3 SNA traffic, becouse of the interactive nature of the traffic and consistant requirements of requests/responses, is crowded out of available bandwidth on the line.
4 Response time for SNA traffic slows to a crawl, affecting users on token ring network.

SNA Traffic and IP packets over port 1 on NETBuilder bridge/router A
(protocols contending for bandwidth resources)

Figure 38-1 IP and SNA Traffic Contention (Without Protocol Reservation)

To deal with this situation, you can use protocol reservation to reserve a percentage of the port bandwidth to each protocol. Figure 38-2 shows this same example, only with IP allocated 50 percent of bandwidth and SNA traffic allocated 40 percent of traffic (the other 10 percent is the default, for other traffic).

The allocation of the bandwidth configured with protocol reservation occurs only when the different packet types actually contend for the bandwidth of the configured port.


SNA traffic
NETBuilder II bridge/router A
NETBuilder II bridge/router B
End stations
!1
!4
SNA traffic: 40% of bandwidth
IP traffic: 50% of bandwidth
10% of bandwidth reserved
TR
Ethernet
IP traffic

SNA traffic and IP packets dividing percentage of bandwidth based on protocol reservation percentages

Figure 38-2 IP and SNA Traffic Contention (With Protocol Reservation)

Protocol reservation can be used to allocate recommended bandwidth for other protocols besides IP and SNA traffic. You also have wide flexibility in determining which protocols you want to reserve bandwidth to, and how much. For more information about how protocol reservation works, refer to “How Protocol Reservation Works” on page 38-18.

Protocol Reservation Procedural Overview

This section provides an overview of how to configure protocol reservation for different traffic types. Because procedures for each of the traffic types varies, read this section to determine the proper procedure for your configuration.

For specific step-by-step configuration procedures, refer to “Configuring for Bridged Traffic or IP- or IPX-Routed Traffic” on page 38-6, or “Configuring for IBM Traffic” on page 38-11.

Protocol reservation can be configured using a variety of procedures, depending on the type of packet traffic you are configured. Table 38-1 lists the traffic types that can be configured for protocol reservation and the procedure used to configure each. More detailed information about the procedures for each traffic type follows the table.

Table 38-1 Packet Types and Configuration Procedures for Protocol Reservation

Traffic Types	Configuration Procedure	Mask	Tag	See Configuration Examples in This Chapter
All Bridged traffic including IP, IPX, AppleTalk, XNS, SNA, NetBIOS	FIlter Service*	Built-in masks or user-defined masks SNA and NetBIOS need user-defined bridged masks	User-defined	“Configuring for Bridged Traffic” on page 38-6

(continued)

Table 38-1 Packet Types and Configuration Procedures for Protocol Reservation (continued)

Traffic Types	Configuration Procedure	Mask	Tag	See Configuration Examples in This Chapter
IP-Routed traffic such as DLSw (*within* DLSw tunnel), FTP, IP, IPDATA, ICMP, SMTP, TCP, TELNET, and UDP	-IP FilterAddrs parameter†	Built-in protocol masks	User-defined	"Configuring for IP-Routed Packets" on page 38-7
IPX-routed traffic	FIlter Service*	Built-in masks or user-defined masks	User-defined	"Configuring for IPX-Routed Traffic" on page 38-9
DLSw traffic for a port on a bridge/router that is the endpoint of the DLSw tunnel	-PORT PROTocolRsrv parameter‡	Not applicable	Built-in DLSW tag	"Configuring for DLSw Traffic at the Tunnel Endpoint" on page 38-12
DLSw traffic to a specific DLSw peer that is the endpoint of the DLSw tunnel	-PORT PROTocolRsrv parameter**	Not applicable	Built-in DLSWPEER tag	"Configuring for DLSw Traffic at the Tunnel Endpoint" on page 38-12
LLC2 traffic, which carries SNA and NetBIOS packets	-FIlter Service*	Built-in LLC2 masks: SNA, or NetBIOS	User-defined	"Configuring for LLC2 Traffic for SNA Boundary Routing" on page 38-13
APPN-routed traffic	-FIlter Service*	Built-in LLC2 mask: APPN	User-defined	"Configuring for APPN-Routed Traffic" on page 38-14

* Refer to Chapter 4 in this guide and to Chapter 23 in *Reference for NETBuilder Family Software*.

† Refer to "Configuring Packet Filtering" in Chapter 6 for IP filtering examples, and to Chapter 29 in *Reference for NETBuilder Family Software* for parameter syntax.

‡ Refer to Chapter 43 in *Reference for NETBuilder Family Software*.

**Refer to Chapter 43 in *Reference for NETBuilder Family Software*.

When you configure a FIlter POLicy for use with protocol reservation, the FIlter POLicy should not specify a port number. Not specifying a port number will ensure that the protocol reservation valve will control bandwidth as defined. The examples in this chapter follow this recommendation and will operate correctly regardless of the configuration of the WAN port (Frame Relay with or without virtual ports, PPP, or WAN Extender).

Protocol reservation uses bandwidth allocation rules to determine how to allocate bandwidth of one traffic type compared to the bandwidth of other traffic types. For more information, refer to "Bandwidth Allocation Process Rules" on page 38-19.

When you enter the -PORT PROTocolRsrv command, you must specify a physical WAN port, not a virtual port. This rule applies for all bridge/router port configurations with the exception of WAN Extender ports, where you must enter the -PORT PROTocolRsrv command and specify a virtual port.

More specifically, protocol reservation is configured using the following procedures for each traffic type:

- Procedure for mnemonic filtering

  To configure protocol reservation using the mnemonic filtering procedure, perform the following major tasks:

  - Using various FIlter Service parameters, assign a built-in or user-defined mask, create a filter policy, and designate the type of packet filtering that is being performed.

    Refer to Chapter 4 for mnemonic filtering descriptions and lists of built-in masks and instructions on how to create user-defined masks. Refer to Chapter 23 in *Reference for NETBuilder Family Software* for syntax and descriptions of FIlter Service parameters.

  - Set the -PORT QueueCONTrol parameter to PROTocolRsrv and use the -PORT PROTocolRsrv parameter to assign bandwidth percentage and a tag for the packet traffic type.

    Refer to Chapter 43 in *Reference for NETBuilder Family Software* for syntax and descriptions of the -PORT PROTocolRsrv parameter.

- Procedure for IP filtering

  To configure protocol reservation using the IP filtering procedure, perform the following major tasks:

  - Use IP Service parameters to create a filter and enable filtering.

    Refer to "Configuring Packet Filtering" in Chapter 6 for IP filtering descriptions and examples. Refer to Chapter 29 in *Reference for NETBuilder Family Software* for syntax and descriptions of the IP Service parameters.

  - Set the -PORT QueueCONTrol parameter to PROTocolRsrv, and use the -PORT PROTocolRsrv parameter to assign bandwidth percentage and a tag for the packet traffic type.

    Refer to Chapter 43 in *Reference for NETBuilder Family Software* for syntax and descriptions of the -PORT PROTocolRsrv parameter.

- Procedure for DLSw

  To configure a port for protocol reservation using the DLSw procedure (used for all DLSW tunnel endpoint packets or packets designated for a DLSW peer for an end of the DLSw tunnel — traffic that will not be routed forward), perform the following major tasks:

  - Set the -PORT QueueCONTrol parameter to PROTocolRsrv.
  - Select either the DLSw tag or the DLSWPEER tag (and enter the peer's IP address) from the -PORT PROTocolRsrv parameter options, and enter the percentage of bandwidth to be designated for the DLSw or DLSwPEER packet type.

    Refer to Chapter 43 in *Reference for NETBuilder Family Software* for the syntax and descriptions of the -PORT QueueCONTrol and -PORT PROTocolRsrv parameters.

Using Protocol Reservation with Frame Relay Virtual Ports

When you configure protocol reservation for traffic being sent over Frame Relay virtual ports, you must configure protocol reservation on the physical port. You can set up the filter to tag packets on virtual ports, but if you configure the filters using this method, you must configure filters for *all* virtual ports assigned to the physical port. If you have a large number of virtual ports and you configure protocol reservation filters for each virtual port, system performance will be negatively impacted. 3Com recommends that you configure the protocol reservation filters to apply to the bridge/router instead of individual ports, then configure the protocol reservation percentages to apply to individual ports.

Configuring for Bridged Traffic or IP- or IPX-Routed Traffic

This section describes how to configure protocol reservation for IP-routed packets, bridged traffic, and IPX-routed packets. This section provides the following procedures:

- "Configuring for Bridged Traffic" (see next section)
- "Configuring for IP-Routed Packets" (on page 38-7)
- "Configuring for IPX-Routed Traffic" (on page 38-9)

The procedures in this section only describe how to configure a single traffic type at one time. For configuration examples on how to configure mixed environments, refer to "Protocol Reservation Configuration Examples" on page 38-15.

Configuring for Bridged Traffic

This section provides an example on how to use the mnemonic filtering procedure to configure protocol reservation for all bridged protocol packets such as IP, IPX, and AppleTalk.

In this example, a bridge/router is bridging IPX, IP, and AppleTalk traffic. The user wants to reserve 40 percent of the bandwidth for IPX traffic, 35 percent for IP traffic, and 20 percent for AppleTalk traffic transmitting from WAN port 4.

Five percent of the bandwidth is automatically set aside as a default for untagged traffic. Figure 38-3 illustrates this example.

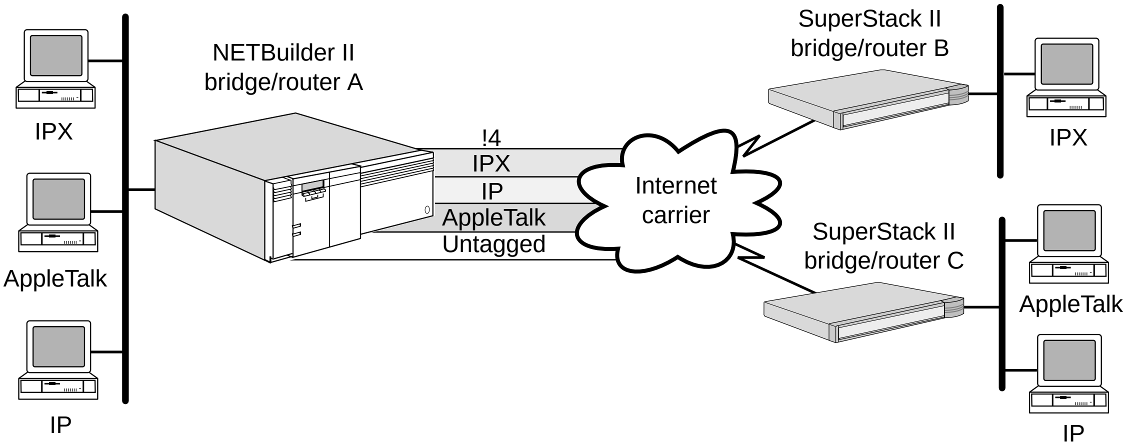


Figure 38-3 Hardware Configuration for Bridged Packets Example

To allocate the required bandwidth for these bridged protocols, follow these steps on NETBuilder II bridge/router A:

1 Add a filter policy for each protocol with built-in IPX, IP, and AppleTalk filter masks by entering:

```
ADD -FIlter POLicy POLICY1 PROTocolRsrv ANY_IPX IPX
ADD -FIlter POLicy POLICY2 PROTocolRsrv ANY_IP IP
ADD -FIlter POLicy POLICY3 PROTocolRsrv ANY_APPLE ATALK
```

When you configure filters to control protocol reservation, the filters will affect all ports on the bridge/router. You can configure filters for specific ports, but this configuration is not recommended for use with protocol reservation.

2 Apply the bridge filtering policies by entering:

```
SETDefault -FIlter SELection = BRidge
```

3 Enable the FIlter Service by entering:

```
SETDefault -FIlter CONTrol = Enable
```

4 Define the percentage of bandwidth to be reserved for each protocol and enter name tags that match those entered in the -FIlter POLicy commands in step 1 by entering:

```
ADD !4 -PORT PROTocolRsrv ANY_IPX 40
ADD !4 -PORT PROTocolRsrv ANY_IP 35
ADD !4 -PORT PROTocolRsrv ANY_APPLE 20
```

If configuring protocol reservation on a WAN Extender port, enter the PROTocolRsrv command specifying a virtual port instead of a physical port.

5 Specify the PROTocolRsrv option for the -PORT Service QueueCONTrol parameter by entering:

```
SETDefault !4 -PORT QueueCONTrol = PROTocolRsrv
```

For more information and examples on how to use the mnemonic filtering procedure to set up protocol reservation, refer to "PROTocolRsrv <Tag>" in Chapter 4.

Configuring for IP-Routed Packets

This section provides an example on how to use the IP filtering procedure to configure protocol reservation for IP-routed packets.

Prerequisites

Before beginning this procedure, complete the following tasks on the NETBuilder II bridge/router:

- Add an IP filter that assigns 20 percent of reserved bandwidth for all Telnet sessions, 30 percent of reserved bandwidth for all FTP packets, and 25 percent for all other IP packets transmitted through port 2.
- Set the -IP FilterDefAction parameter so that all packets that do not meet the filtering conditions are forwarded. Figure 38-4 shows the hardware configuration for this example.

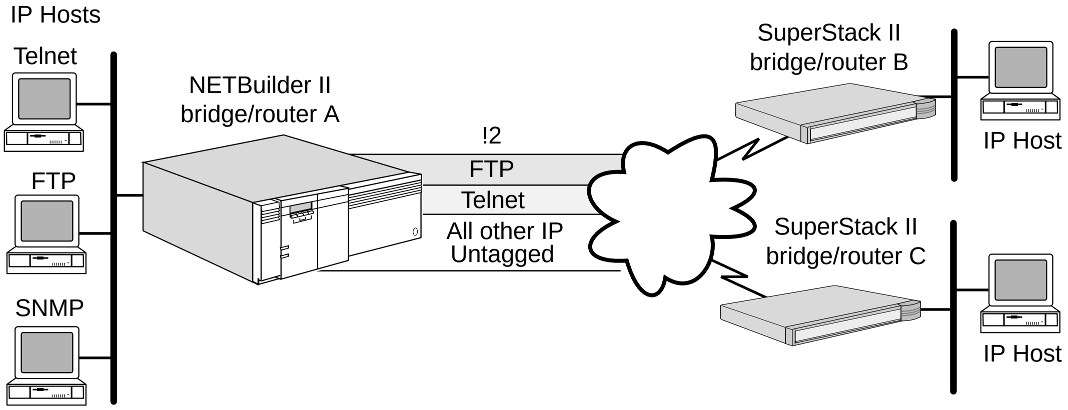

IP Hosts
Telnet
FTP
SNMP
NETBuilder II bridge/router A
!2
FTP
Telnet
All other IP Untagged
SuperStack II bridge/router B
IP Host
SuperStack II bridge/router C
IP Host

Figure 38-4 Hardware Configuration for IP-Routed Packets Example

Procedure

To configure these filtering operations, follow these steps on NETBuilder II bridge/router A:

1 Set up IP routing according to the information in Chapter 6.

2 Add IP filters that do the following for packets:

- Assign 20 percent of reserved bandwidth for all Telnet packets and designate a tag name of "Telnet" to identify the packets.
- Assign 30 percent of reserved bandwidth for all FTP packets and designate a tag name of "FTP" to identify the packets.
- Assign 25 percent of reserved bandwidth for all other IP packets and designate a tag name of "ALLOther-IP" to identify the packets.

Add these filters by entering:

```
ADD -IP FilterAddrs ALL ALL PROTocolRsrv = TELNETTAG Telnet
ADD -IP FilterAddrs ALL ALL PROTocolRsrv = FTPTAG FTP
ADD -IP FilterAddrs ALL ALL PROTocolRsrv = ALLOther-IP IP
```

When you configure filters to control protocol reservation, the filters will affect all ports on the bridge/router. You can configure filters for specific ports, but this configuration is not recommended for use with protocol reservation.

3 Add an IP filter default action that forwards any packets that do not satisfy the filter requirements by entering the following command:

```
SETDefault -IP FilterDefAction = Forward
```

4 Enable the IP filtering feature by entering:

```
SETDefault -IP CONTrol = Filtering
```

5 Assign 20 percent of bandwidth to the PROTocolRsrv name tag "Telnet," 30 percent of the bandwidth to the PROTocolRsrv name tag "FTP," and 25 percent of bandwidth to the PROTocolRsrv name tag "ALLOther-IP" for port 2 by entering:

```
ADD !2 -PORT PROTocolRsrv TELNETTAG 20
ADD !2 -PORT PROTocolRsrv FTPTAG 30
ADD !2 -PORT PROTocolRsrv ALLOther-IP 25
```

If configuring protocol reservation on a WAN Extender port, enter the PROTocolRsrv command specifying a virtual port instead of a physical port.

6 Set PROTocolRsrv as the QueueCONTrol option for port 2 by entering:

```
SETDefault !2 -PORT QueueCONTrol = PROTocolRsrv
```

After you have entered these commands, any packet sent out by the system through port 2 that has the name tag "Telnet" is allocated 20 percent of the bandwidth, packets with the name tag "FTP" are allocated 30 percent of the bandwidth, and IP packets with the name tag "ALLOther-IP" are allocated 25 percent.

Five percent of the bandwidth is allocated for all untagged traffic, and the remaining 20 percent of the bandwidth is added to the default to be used by the configured protocols or by the untagged traffic on a first-come first-serve basis.

For more information and examples on how to use the IP filtering procedure to set up protocol reservation, refer to "Configuring Packet Filtering" in Chapter 6.

How Protocol Reservation Allocates Different IP Protocol Types

Using IP filtering, how you define the tags for IP packets or other protocols in the TCP/IP protocol suite determines how much percentage bandwidth is used for each. If you configure a percentage of bandwidth for a specific protocol, such as UDP, TCP, or Telnet, then those packets will be removed from the percentage allocated to IP. However, if you define a percentage of bandwidth for IP only, then all the IP-related protocols such as UDP, TCP, and Telnet will be included within that percentage.

Figure 38-5 is an example of how this allocation works. In the example, 60 percent of the bandwidth is allocated to IP.

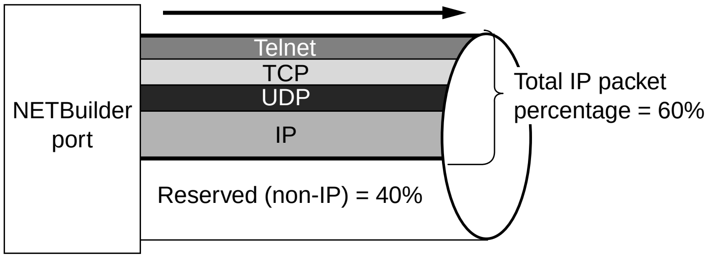


Figure 38-5 IP Protocols Allocation (UDP Included)

Figure 38-6 shows the same situation but with 60 percentage of bandwidth allocated to IP and 20 percent allocated to UDP. While UDP traffic is no longer included in the 60 percent of bandwidth allocated to IP, TCP and Telnet are still allocated as a subset of the IP bandwidth percentage. The bandwidth allocated to UDP in this case is exclusive of bandwidth allocated to IP.

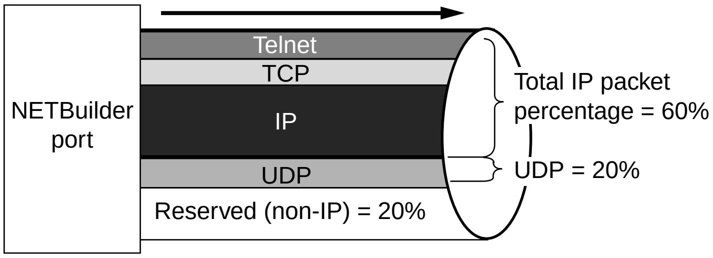


Figure 38-6 IP Protocols Allocation (UDP Excluded)

Configuring for IPX-Routed Traffic

The following example describes how to configure protocol reservation for IPX-routed traffic (IPXRIP packets) transmitted from WAN port 4 on a central node NETBuilder bridge/router to an end node NETBuilder bridge/router.

In this example, WAN port 4 on bridge router A is configured for protocol reservation to reserve the following bandwidth percentages for the following packet types (refer to Figure 38-7):

- 45 percent of the bandwidth for IPXRIP-routed packets
- 50 percent of the bandwidth for IP-routed packets
- 5 percent as a default for AppleTalk-routed packets

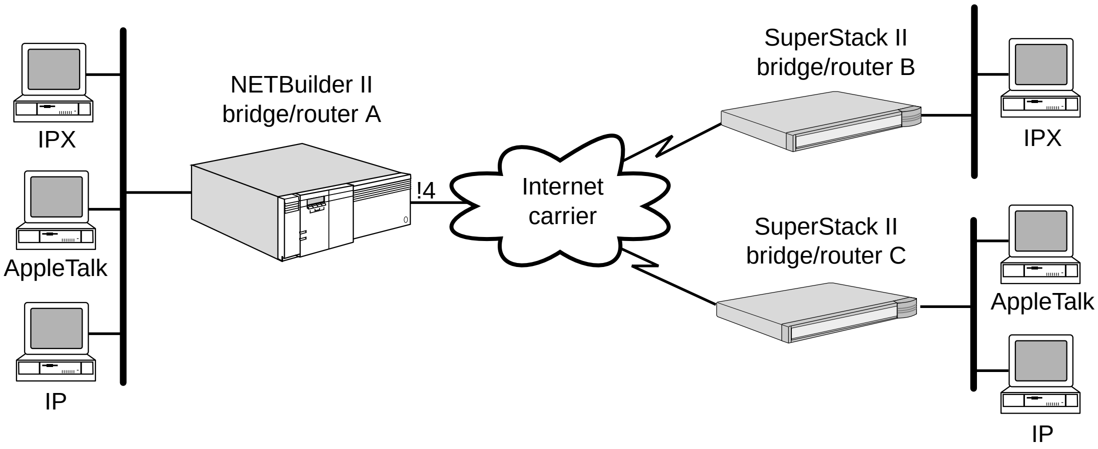


Figure 38-7 Hardware Configuration for IPX-Routed Traffic

To allocate the required bandwidth for these protocols, follow these steps on NETBuilder II bridge/router A:

1 Set up IPX routing according to the information in Chapter 13, set up IP routing according to Chapter 6, and AppleTalk routing according to Chapter 14.

2 Add a filter policy named "IPXPolicy" with PROTocolRsrv as the action option, with the name tag "IPXtag" and with the built-in mask "IPXRIP" by entering:

```
ADD -FIlter POLicy IPXPOLICY PROTocolRsrv IPXTAG IPXRIP
```

When you configure filters to control protocol reservation, the filters will affect all ports on the bridge/router. You can configure filters for specific ports, but this configuration is not recommended for use with protocol reservation.

3 Apply the filtering policy by entering:

```
SETDefault -FIlter SELection = IPX
```

4 Set the Filter CONTrol parameter to Enable by entering:

```
SETDefault -FIlter CONTrol = Enable
```

5 Add an IP filter that assigns all IP-routed traffic the name tag "IPtag" by entering the following command:

```
ADD -IP FilterAddrs ALL ALL PROTocolRsrv = IPTAG IP
```

6 Add an IP filter default action that forwards any packets that do not satisfy the filter requirements by entering:

```
SETDefault -IP FilterDefAction = Forward
```

7 Enable the IP filtering feature by entering:

```
SETDefault -IP CONTrol = Filtering
```

8 Configure port 4 with the "IPXtag" PROTocolRsrv name tag entered in the filter policy with 45 percent of reserved bandwidth, and with the "IPtag" entered in the IP filter with 50 percent of the bandwidth by entering:

```
ADD !4 -PORT PROTocolRsrv IPXTAG 45
ADD !4 -PORT PROTocolRsrv IPTAG 50
```

If configuring protocol reservation on a WAN Extender port, enter the PROTocolRsrv command specifying a virtual port instead of a physical port.

9 Set PROTocolRsrv as the -PORT QueueCONTrol parameter option for port 4 by entering:

```
SETDefault !4 -PORT QueueCONTrol = PROTocolRsrv
```

After this configuration, 45 percent of the port 4 bandwidth is reserved for IPXRIP packets, 50 percent for IP-routed traffic, and 5 percent is the default untagged traffic, which is AppleTalk in this example.

Configuring for IBM Traffic

This section describes how to configure protocol reservation for IBM traffic types. This section provides instructions for the following procedures:

- "Configuring for DLSw Traffic at the Tunnel Endpoint" (on page 38-12)
- "Configuring for LLC2 Traffic for SNA Boundary Routing" (on page 38-13)
- "Configuring for APPN-Routed Traffic" (on page 38-14)

The procedures in this section only describe how to configure a single traffic type at one time. For configuration examples on how to configure mixed traffic environments, refer to "Protocol Reservation Configuration Examples" on page 38-15.

The following IBM traffic supported by the NETBuilder bridge/routers can be configured for protocol reservation: DLSw, APPN-routed, LLC2 locally terminated (by DLSw or LLC2 tunneling), and SNA and NetBIOS, bridged traffic.

DLSw traffic is used to encapsulate SNA or NetBIOS traffic that is transmitting over a WAN. The DLSw traffic is itself encapsulated in IP traffic frames.

How DLSw traffic is configured for protocol reservation depends on whether the DLSw traffic is being configured for a bridge/router that is the end of the DLSw tunnel or if the DLSw traffic is to be forwarded on through the tunnel to another bridge/router.

Configuring protocol reservation for DLSW traffic for a bridge/router that is the end of the DLSw tunnel is accomplished using the DLSw or DLSwPeer built-in tags. You configure the -PORT PROTocolRsrv parameter and enter DLSW as the name tag option and enter the percentage of bandwidth to be reserved. For more information, refer to "Configuring for DLSw Traffic at the Tunnel Endpoint" on page 38-12.

Configuring a bridge/router for protocol reservation that is forwarding DLSW traffic through the DLSW tunnel is accomplished using the IP filtering procedure that uses IP Service parameters. Refer to "Configuring for IP-Routed Packets" on page 38-7 for instructions on how to use the IP filtering procedure; substitute the DLSw built-in tag for the TELNET or FTP built-in tag in the example.

The APPN built-in mask is used for IBM APPN-routed packets. Refer to "Configuring for APPN-Routed Traffic" on page 38-14.

The following built-in masks are provided for IBM LLC2 traffic at a WAN port where DLSw or LLC2 tunneling is locally terminating the LLC2 connection, for example, at the WAN port connecting a NETBuilder boundary router central site with a NETBuilder leaf node:

- SNA – Used as criteria to select SNA traffic packets.
- NetBIOS – Used as criteria to select NetBIOS traffic packets.

These masks are used as criteria to select the packets to be the recipients of the reserved bandwidth. The packets are identified by name tags that are entered with the -PORT PROTocolRsrv parameter. Refer to "Configuring for LLC2 Traffic for SNA Boundary Routing" on page 38-13.

Configuring for DLSw Traffic at the Tunnel Endpoint

The following example describes how to configure protocol reservation to assign a percentage of a WAN port's bandwidth for DLSw traffic transmitting from a port on a bridge/router that is the endpoint of a DLSw tunnel. The DLSw packets (which are encapsulated within IP packets) can carry SNA packets, NetBIOS packets, or both.

This example allocates bandwidth of a DLSw tunnel endpoint on port 4 as follows:

- 70 percent of the bandwidth for all DLSw traffic
- 25 percent of the bandwidth for DLSw traffic destined for the DLSw peer at IP address 200.200.1.1

Figure 38-8 shows the hardware configuration for this example.

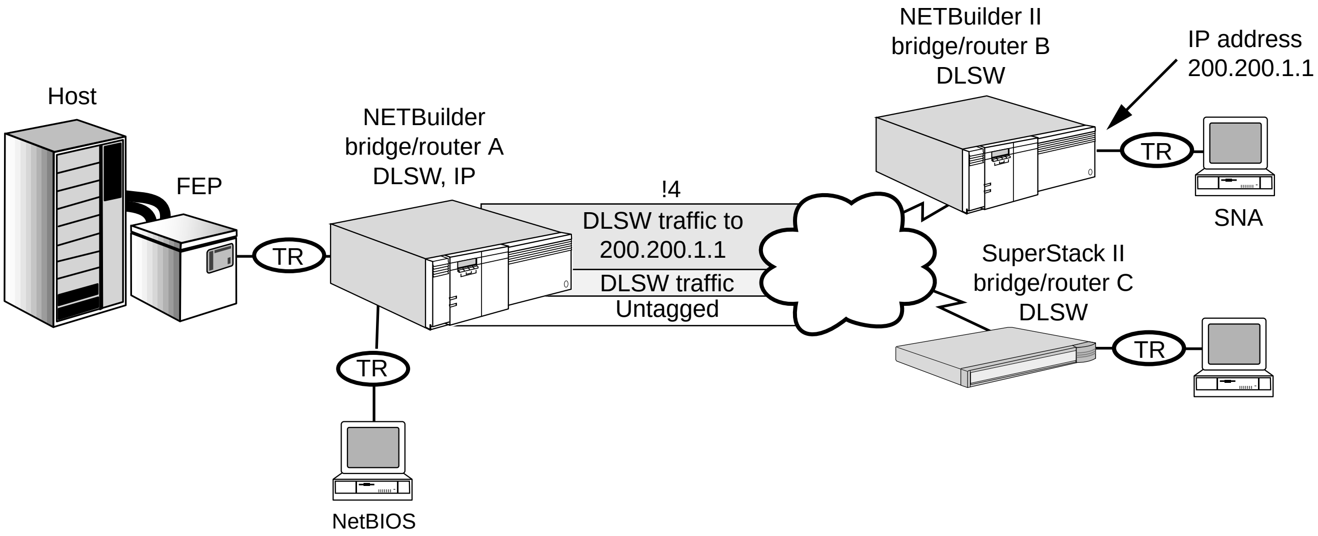


Figure 38-8 DLSw Tunnel Endpoint Hardware Configuration

To allocate the required bandwidth for the DLSw traffic in this example, follow these steps:

1. Set up a DLSw tunnel according to the information in Chapter 24.
2. Assign 70 percent of bandwidth for DLSw traffic and 25 percent of the bandwidth for traffic destined for the DLSw peer at IP address 200.200.1.1 on port 4 by entering (DLSw and DLSwPeer have a built-in tags so you do not need to enter a tag):

```
ADD !4 -PORT PROTocolRsrv DLSw 70
ADD !4 -PORT PROTocolRsrv DLSwPeer 200.200.1.1 25
```

If configuring protocol reservation on a WAN Extender port, enter the PROTocolRsrv command specifying a virtual port instead of a physical port.

3 Set PROTocolRsrv as the -PORT QueueCONTrol parameter option for port 4 by entering:

```
SETDefault !4 -PORT QueueCONTrol = PROTocolRsrv
```

After you have completed this configuration, 70 percent of the bandwidth is reserved for all DLSw traffic transmitting out of port 4, and 25 percent of the bandwidth is reserved for the DLSw peer with IP address 200.200.1.1. The 5 percent of bandwidth is the default to be used for untagged traffic transmitting from port 4.

Configuring for LLC2 Traffic for SNA Boundary Routing

The following example describes how to set up protocol reservation for LLC2 traffic (carrying SNA or NetBIOS packets, or both) transmitting from a WAN port on a NETBuilder bridge/router serving as a central node to another NETBuilder bridge/router serving as an end node.

In this example, WAN port 4 on the central node bridge/router is configured for protocol reservation to reserve the following bandwidth percentages for the following packet types (see Figure 38-9):

- 50 percent of the bandwidth for SNA-bridged packets
- 45 percent of the bandwidth for NetBIOS-bridged packets

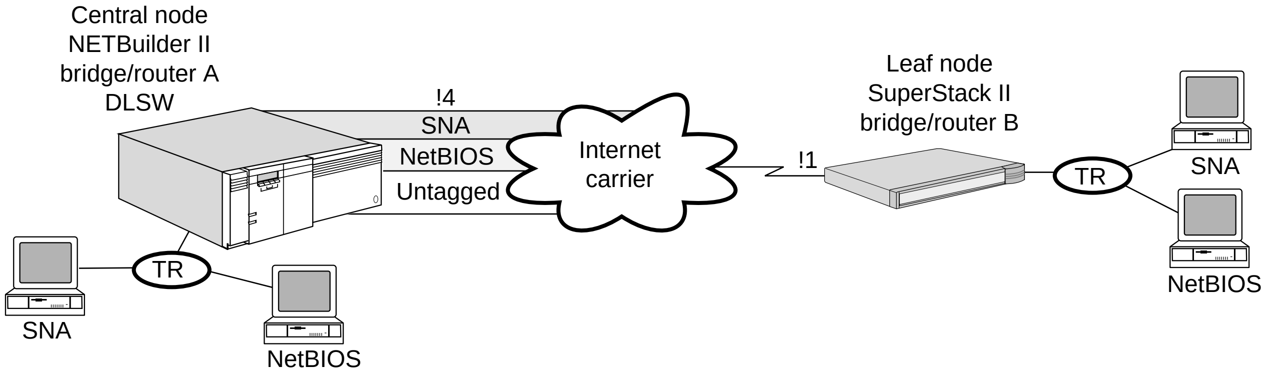


Figure 38-9 LLC2 Example Hardware Configuration

To configure protocol reservation for SNA and NetBIOS packets encapsulated in LLC2 traffic, follow these steps:

1 Configure bridging according to Chapter 3.

2 Assign the following filter policies:

- A policy named “SNAPolicy,” with the built-in mask SNA, and with PROTocolRsrv as the action option and the name tag “SNAtag” with no port number specified.
- A policy named “NetBIOSPol,” with the built-in mask NetBIOS, and with PROTocolRsrv as the action option and the name tag “NetBIOStag.

Assign these policies by entering:

```
ADD -FIlter POLicy SNAPOLICY PROTocolRsrv SNATAG SNA
ADD -FIlter POLicy NETBIOSPOL PROTocolRsrv NETBIOSTAG NetBIOS
```

When you configure filters to control protocol reservation, the filters will affect all ports on the bridge/router. You can configure filters for specific ports, but this configuration is not recommended for use with protocol reservation.

3 Apply the filtering policies by entering:

```
SETDefault -FIlter SELection = LLC2
```

4 Set the Filter CONTrol parameter to Enable by entering:

```
SETDefault -FIlter CONTrol = Enable
```

5 Using the -PORT PROTocolRsrv parameter, configure the WAN port 4 with the name tags assigned in step 2 for the following bandwidth percentages:

- SNAtag 25 percent
- NetBIOStag 20 percent

Assign these percentages by entering:

```
ADD !4 -PORT PROTocolRsrv SNATAG 50
ADD !4 -PORT PROTocolRsrv NETBIOSTAG 45
```

If configuring protocol reservation on a WAN Extender port, enter the PROTocolRsrv command specifying a virtual port instead of a physical port.

6 Set the -PORT QueueCONTrol parameter to PROTocolRsrv for port 4 by entering:

```
SETDefault !4 -PORT QueueCONTrol = PROTocolRsrv
```

Configuring for APPN-Routed Traffic

The following example describes how to configure protocol reservation so that APPN-routed traffic transmitted from WAN port 4 on a central node NETBuilder bridge/router to an end node NETBuilder bridge/router is assigned 10 percent of the bandwidth (refer to Figure 38-10).

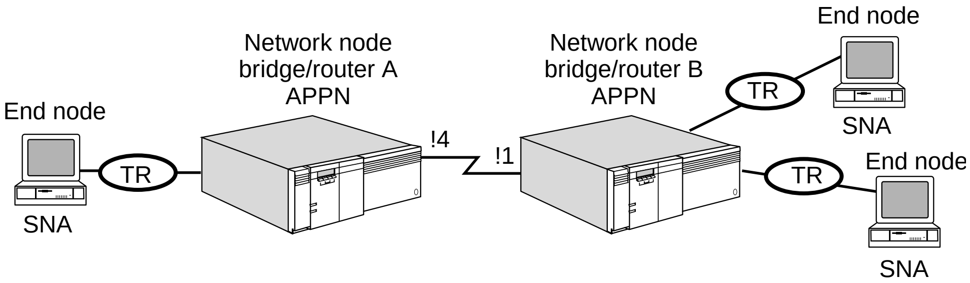


Figure 38-10 APPN Routed Hardware Configuration

To configure port 4 on NETBuilder bridge/router A for these settings, follow these steps:

1 Set up APPN according to the information in Chapter 10.

2 Add a filter policy named “APPNPolicy” with PROTocolRsrv as the action option, with the name tag “APPNtag” and with the built-in mask “APPN” by entering:

```
ADD -FIlter POLicy APPNPolicy PROTocolRsrv APPNTAG APPN
```

When you configure filters to control protocol reservation, the filters will affect all ports on the bridge/router. You can configure filters for specific ports, but this configuration is not recommended for use with protocol reservation.

3 Apply the filtering policy by entering (LLC2 is used as the filter type for SNA, NetBIOS, and APPN traffic):

```
SETDefault -FIlter SELection = LLC2
```

4 Set the Filter CONTrol parameter to Enable by entering:

```
SETDefault -FIlter CONTrol = Enable
```

5 Configure port 4, with the "APPNtag" PROTocolRsrv name tag entered in the filter policy in step 2, with 75 percent of reserved bandwidth by entering:

```
ADD !4 -PORT PROTocolRsrv APPNTAG 75
```

If configuring protocol reservation on a WAN Extender port, enter the PROTocolRsrv command specifying a virtual port instead of a physical port.

6 Set PROTocolRsrv as the -PORT QueueCONTrol parameter option for port 4 by entering:

```
SETDefault !4 -PORT QueueCONTrol = PROTocolRsrv
```

Protocol Reservation Configuration Examples

This section provides protocol reservation configuration examples that use the configuration procedures described earlier in this chapter.

Example 1: Mixed Bridged Traffic

In this example, you are configuring protocol reservation on WAN port 1 that supports SNA- and NetBIOS- bridged traffic through several DLSw tunnels and IP-bridged traffic at the same time.

The hardware configuration is as follows (see Figure 38-11):

- Bridge/Router A, B, C, and D run DLSw.
- Bridge/Router A has DLSw tunnels with bridge/router B, C, and D.
- Bridge/Router A and bridge/router D run IP.

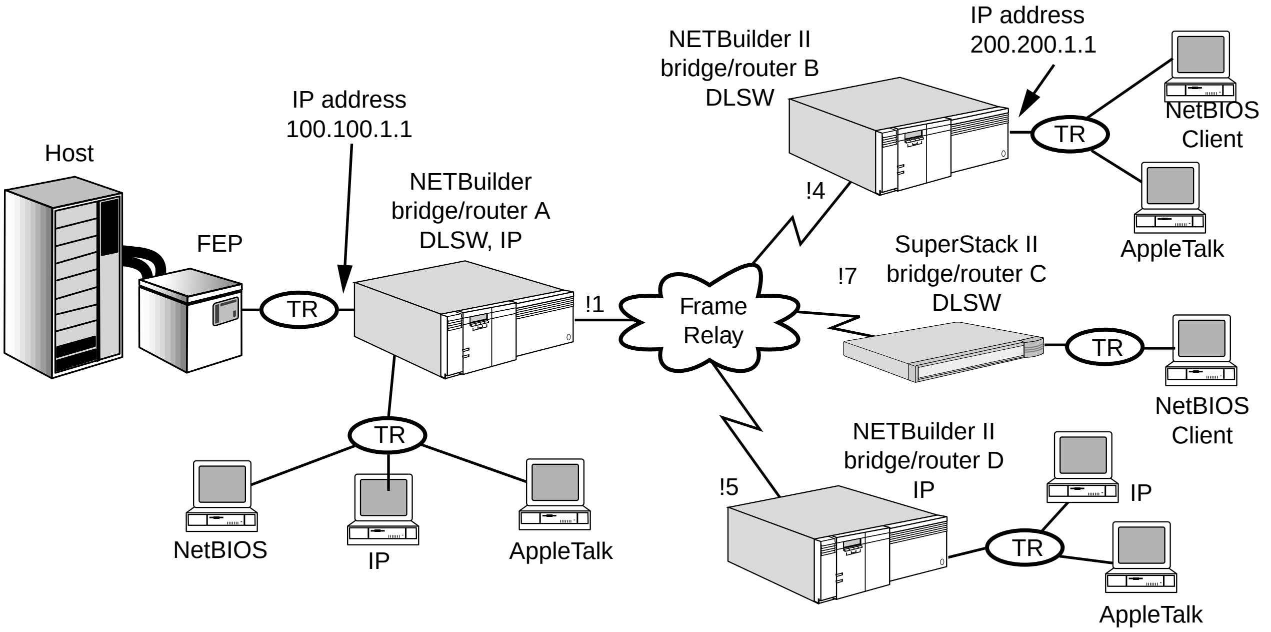


Figure 38-11 Mixed-Bridged Traffic Hardware Configuration

Table 38-2 lists the port bandwidth for different bridged traffic types for the ports on each bridge/router.

Table 38-2 Traffic Type and Reserved Bandwidth for Example 1

Traffic Type	Reserved Bandwidth
NETBuilder Bridge/Router A port 1:	
Default (AppleTalk traffic)	5 percent
DLSw traffic to DLSwPeer Bridge/Router B with IP address 200.200.1.1 (SNA)	20 percent
All other DLSw traffic (NetBIOS)	50 percent
Traffic to IP hosts	25 percent
NETBuilder Bridge/Router B port 4:	
DLSw traffic to DLSwPeer Bridge/Router A with IP address 100.100.1.1 (SNA)	45 percent
DLSw traffic to all other DLSw nodes	45 percent
NETBuilder Bridge/Router C port 7:	
DLSw traffic to DLSwPeer Bridge/Router A with IP address 100.100.1.1 (NetBIOS)	95 percent
NETBuilder Bridge/Router D port 5:	
IP traffic	45 percent
AppleTalk traffic	45 percent

Table 38-3 lists all the commands required to configure protocol reservation on each of the bridge/routers shown in the figure. Before entering the commands in the table, configure bridging (refer to Chapter 3), and DLSw (refer to Chapter 24).

Table 38-3 Required Commands (Example 1)

Commands Entered on Bridge/Router A	Commands Entered on Bridge/Routers B, C and D
ADD -FIlter POLicy IP POLICY PROTocolRsrv IPtag IP	Bridge/Router B:
SETDefault -FIlter SELection = BRidged	**ADD !4 -PORT PROTocolRsrv DLSwPeer 100.100.1.1 45**
SETDefault -FIlter CONTrol = Enable	**ADD !4 -PORT PROTocolRsrv DLSw 45**
ADD !1 -PORT PROTocolRsrv IPTAG 25	**SETDefault !4 - PORT QueueCONTrol = PROTocolRsrv**
ADD !1 -PORT PROTocolRsrv DLSwPeer 200.200.1.1 20	Bridge/Router C:
ADD !1 -PORT PROTocolRsrv DLSw 50	**ADD !7 -PORT PROTocolRsrv DLSwPeer 100.100.1.1 95**
SETDefault !1 - PORT QueueCONTrol = PROTocolRsrv	**SETDefault !7 - PORT QueueCONTrol = PROTocolRsrv**
	Bridge/Router D:
	ADD -FIlter POLicy IP Policy1 PROTocolRsrv IPTAG IP
	SETDefault -FIlter SELection= BRidged
	SETDefault -FIlter CONTrol = Enable
	ADD !5 -PORT PROTocolRsrv IPTAG 45
	ADD -FIlter POLicy Policy2 PROTocolRsrv any_Apple ATALK
	ADD !5 -PORT PROTocolRsrv any_Apple 45
	SETDefault !5 - PORT QueueCONTrol = PROTocolRsrv

Example 2: Mixed-Routed Packets

In this example, you are configuring protocol reservation on a WAN port for APPN-routed and IP-routed traffic at the same time. AppleTalk-routed packets use the 5 percent default bandwidth. Figure 38-12 shows the hardware configuration for this example.

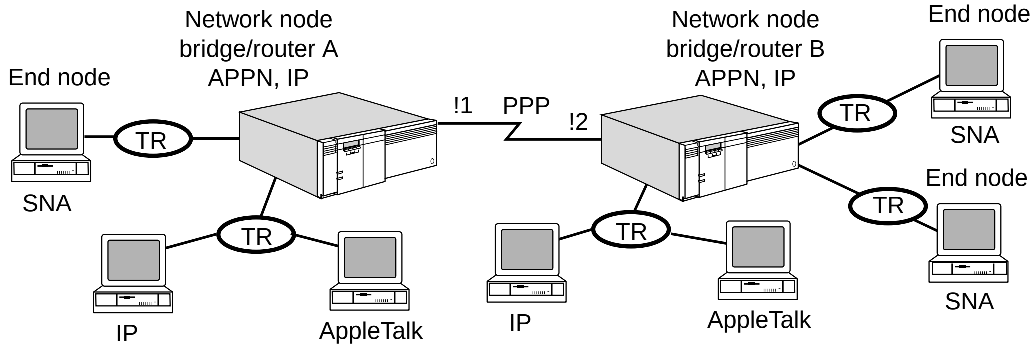


Figure 38-12 Mixed Routed Packets Hardware Configuration

The goal in this example is to divide the bandwidth of the port for different types of transmitting traffic from the port as shown in Table 38-4.

Table 38-4 lists the port bandwidth for different routed traffic types for the ports on each bridge/router.

Table 38-4 Traffic Type and Reserved Bandwidth for Example 2

Traffic Type	Reserved Bandwidth
NETBuilder Bridge/Router A port 1:	
Default (AppleTalk traffic)	5 percent
APPN-routed traffic	70 percent
IP-routed traffic	25 percent
NETBuilder Bridge/Router B port 2:	
Default (AppleTalk traffic)	5 percent
APPN-routed traffic	70 percent
IP-routed traffic	25 percent

Table 38-5 lists all the commands required to configure protocol reservation on each of the bridge/routers shown in the figure. Before entering the commands in the table, configure IP routing (refer to Chapter 6), APPN routing (refer to Chapter 10), and AppleTalk routing (refer to Chapter 14).

Table 38-5 Required Commands (Example 2)

Commands Entered on Bridge/Router A	Commands Entered on Bridge/Router B
`ADD -IP FilterAddrs ALL ALL PROTocolRsrv IPTAG IP`	`ADD -IP FilterAddrs ALL ALL PROTocolRsrv IPTAG IP`
`SETDefault -IP FilterDefAction = Forward`	`SETDefault -IP FilterDefAction = Forward`
`SETDefault -IP CONTrol = Filtering`	`SETDefault -IP CONTrol = Filtering`
`ADD -FIlter POLicy APPNPolicy PROTocolRsrv APPNTAG APPN`	`ADD -FIlter POLicy APPNPolicy PROTocolRsrv APPNtag APPN`
`SETDefault -FIlter SELection= LLC2`	`SETDefault -FIlter SELection= LLC2`
`ADD !1 -PORT PROTocolRsrv APPNTAG 70`	`ADD !2 -PORT PROTocolRsrv APPNtag 70`
`ADD !1 -PORT PROTocolRsrv IPTAG 25`	`ADD !2 -PORT PROTocolRsrv IPTAG 25`
`SETDefault !1 -PORT QueueCONTrol = PROTocolRsrv`	`SETDefault !2 - PORT QueueCONTrol = PROTocolRsrv`

Example 3: Virtual Ports

In this example, when you configure protocol reservation for Frame Relay virtual ports, you configure the protocol reservation on the parent port, not on the virtual port. No special configuration for virtual ports is necessary if you do not specify port numbers when you set up the filter masks.

Figure 38-13 shows a configuration with virtual ports. In this configuration, on NETBuilder bridge/router A, you configure the specific filter masks without specifying a port number. You then configure the filter percentages using the -PORT PROTocolRsrv parameter on port 2, the parent port to virtual ports 1 and 2.

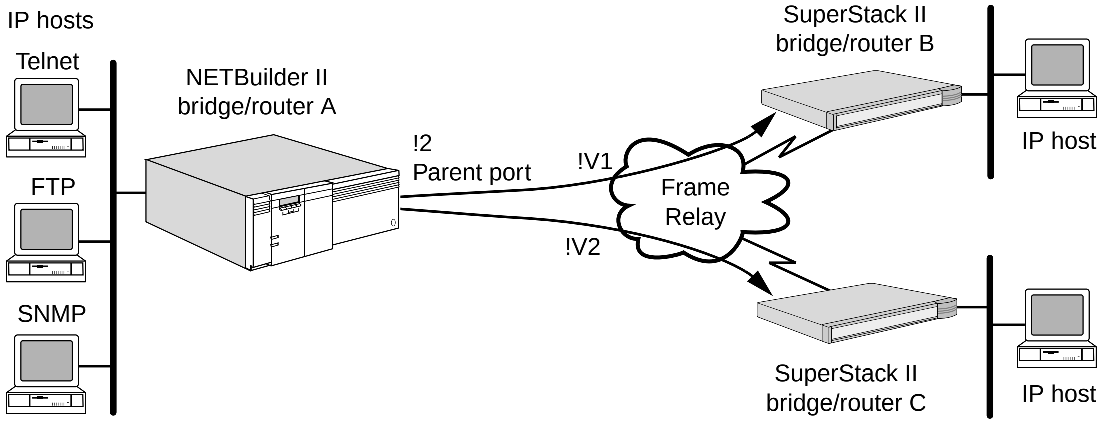


Figure 38-13 Protocol Reservation on Virtual Ports

How Protocol Reservation Works

Protocol reservation allows you to assign a percentage of bandwidth to designated packets transmitting through a specified WAN logical port (no virtual ports) and meeting certain conditions. The conditions can include protocol type, packet length, and packets destined for a specified address, among others.

Protocol reservation allows you to reserve bandwidth for lower bandwidth usage, interactive, response-time-sensitive, or transaction-oriented network application packets. These type of packets are normally crowded out by heavy bandwidth usage applications such as file transfer or mail.

For example, in a multiprotocol environment that includes IBM protocol traffic (such as response-time-sensitive SNA packets) mixed in with other protocol traffic (such as IP or IPX), SNA devices throttle back the data transmission rate to the end station when they sense available bandwidth decreasing. If other network protocols increase this bandwidth consumption, SNA devices will throttle back the data transmission rate more, which slows the response time of SNA packets even more.

To avoid this situation, use protocol reservation to provide a percentage of bandwidth for the SNA packets and to restrict the percentage of bandwidth to the other more aggressive protocol packets to ensure that the small, response-time-sensitive SNA packets to pass through port in a timely manner.

How Protocol Reservation Controls Bandwidth for Traffic Types

Protocol reservation provides a “valve” that sits above the transmit queue and controls the amount of bandwidth reserved for specific types of data (identified by the user through filtering schemes). During times of stress when bandwidth of the WAN link is consumed beyond a threshold, the protocol reservation valve engages and works to normalize ratios of traffic types down to the configured percentages. If the bandwidth is not utilized to threshold, the NETBuilder bridge/router does not attempt to achieve the configured percentages so all packets can be serviced.

The traffic is identified by the user through the use of mnemonic or manually configured filters. OSI, DECnet, Vines, AppleTalk, and XNS routed traffic is treated as “default”; specific percentages cannot be assigned to them (percentages can be allocated if these protocols are bridged). When the protocol reservation valve is engaged each packet is checked against the configured filter(s), which introduces some latency when compared with traffic flow over an uncongested link. However reserving minimum percentages for a protocol helps to prevent session loss, which may occur during traffic bursts.

Some protocols reduce their transmit rate when congestion is sensed, and thus may not use all of their allocated bandwidth. Protocol reservation automatically allocates any unused bandwidth to other protocols, and the desired effect may not be achieved. If you are using a protocol that reduces its transmit rate and does not utilize its configured bandwidth, you may want to use the priority queueing feature, which allows you to control the order in which packets are serviced between high, medium, and low priority queues (refer to Chapter 41).

Protocol reservation supports PPP virtual ports or WAN Extender virtual ports. For Frame Relay and SMDS, protocol reservation is supported only on the parent ports at the physical port level (specific percentages are not applied to individual virtual ports, except for WAN Extender ports).

Tuning

The protocol reservation valve normalizes bandwidth to configured percentages over time intervals. If a large packet is encountered, the packet must be passed to the driver transmit queue in its entirety; it is not fragmented into smaller sizes. Other packets will be passed to the queue behind it. Percentages are maintained over time but it is still possible for some traffic to experience latency in extremely busy environments if larger packets fill up the driver transmit queue. Do not allocate more bandwidth for a protocol than you can use. If a protocol cannot reach the percentage of bandwidth allocated to it, then the bandwidth not used by the protocol will be used by other protocols.

Bandwidth Allocation Process Rules

Protocol reservation uses bandwidth allocation process rules for allocating bandwidth to WAN ports.

Bandwidth Normalization

The protocol reservation features can be used for IP routing, IPX-routing, all bridging protocol traffic, and all NETBuilder-provided IBM traffic. To allow any traffic that is not "tagged" to go through, at least 5 percent of the bandwidth is reserved for this untagged traffic. This reservation is called the *default queue*. Untagged traffic includes non-bridged AppleTalk, XNS, OSI, DecNet, and VINES protocol packets.

If the total configured bandwidth percentages for the port exceed 95 percent, the values are balanced by the system so that the default queue still has its default allotment of approximately 5 percent of the available bandwidth. The rest of the bandwidth is distributed among the entries configured for the port in a ratio to the percentages that were configured for each.

This process of distributing the ratio is called *normalization*. Since the distribution only uses whole numbers for a percentage, the fraction remainders of each protocol are added to the default queue. As a result, the default queue sometimes can have a percentage greater than 5.

Table 38-6 is an example of the traffic type, configured bandwidth, and then the normalized numbers that occur when the configured assignments of bandwidth exceed 95 percent.

Table 38-6 Traffic Type, Configured Bandwidth, and Normalized Bandwidth

Traffic Type	Configured Bandwidth	Normalized Bandwidth
Default (AppleTalk traffic)	5 percent	7 percent
SNA traffic	95 percent	37 percent
IP traffic	95 percent	37 percent
IPX traffic	55 percent	19 percent

Distribution of Non-Allocated Bandwidth

If the total configured bandwidth percentages are less then 95 percent, the non-allocated bandwidth is added to the default to be given to the configured protocols or for untagged traffic on a first-come first-served basis.

For example, if you configure protocol reservation for a WAN port with the following bandwidth allocations:

- 50 percent of the bandwidth for SNA traffic
- 20 percent of the bandwidth for NetBIOS traffic
- 5 percent automatically set aside as default bandwidth for untagged traffic

The remaining 25 percent of the bandwidth is added to the default to be used for SNA traffic, NetBIOS traffic, or for untagged traffic, whatever traffic needs it first.

39 CONFIGURING DATA COMPRESSION

Data compression is an optional feature that may be used to enhance the effective throughput on Point-to-Point Protocol (PPP), X.25, and Frame Relay connections.

If you are using a modem that already performs compression, 3Com suggests that you do not configure data compression.

Three data compression types exist: tinygram, history-based, and per-packet. Tinygram compression is a packet-level compression that can be configured for PPP links only. History-based and per-packet compression are link-level compression types. History-based compression may be configured for PPP and X.25. Per-packet based compression may be configured for PPP, X.25, and Frame Relay. All three types of compression operate and are configured independently.

This chapter describes how to configure these compression types and how data compression works and when to use each type.

For conceptual information, refer to "How Data Compression Works" on page 39-4.

Configuring Data Compression

The following procedures describe how to configure tinygram, history, and per-packet compression.

Configuring Tinygram Compression

The -PATH TinyGramcomp parameter allows you to compress all bridged Ethernet packets that are 64 bytes and are padded with trailing zeros. When the packet is sent on a serial line, the receiving side reinserts the zeros before forwarding the packet to an Ethernet LAN. This compression method is effective only on serial lines and is normally used in the Digital Equipment Corporation/local area transport (DEC/LAT) terminal-to-host environments.

This procedure must be completed at both ends of the link. To enable tinygram compression, follow these steps:

1 Enable tinygram compression on a specific path using:

```
SETDefault !<path> -PATH TinyGramcomp = Enabled
```

2 Verify the PATH configuration using:

```
SHowDefault !<path> -PATH TinyGramcomp
```

3 Activate tinygram compression using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

Configuring Link-Level Compression

Link-level compression can be configured as either history-based or packet-based. History-based link-level compression requires a reliable link for proper operation. PPP and X.25 support both history and per-packet compression. When history-based compression is enabled over a PPP link, Link Access Procedure, Balanced (LAPB) must be run to provide the reliable link. Frame Relay does not provide a reliable link and therefore does not support history-based compression.

Per-packet compression is supported for PPP, X.25, and Frame Relay links.

On X.25 configurations, the data compression feature in version 8.0 software is not backward-compatible with pre-8.0 software versions. When interoperating with pre-8.0 software, do not use the data compression options.

Enabling History-based or Per-packet Compression

This procedure must be completed at both ends of the link. To enable history-based or per-packet link-level compression, follow these steps:

1 Enable link-level history based compression on a particular port using:

```
SETDefault !<port> -PORT COMPressType = HIStory
```

Frame Relay does not support history-based data compression.

2 To select the per-packet link-level compression algorithm, use:

```
SETDefault !<port> -PORT COMPressType = PerPacket
```

3 For the configured compression types to take effect, re-enable the port using:

```
SETDefault !<port> -PORT CONTrol = Enabled
```

Before you re-enable the port with this command, you must perform any optional configuration steps such as those described in “Enabling LAPB for a PPP Link” on page 39-2,“Frame Relay Configuration Options” on page 39-2, or “X.25 Configuration Options” on page 39-3.

Optional Configurations

The following sections describe optional configurations you may perform.

Enabling LAPB for a PPP Link

If you are planning to run the history-based algorithm on a PPP link, set up LAPB by follow these steps:

1 Enable LAPB at both ends of the link using:

```
SETDefault !<path> -LAPB CONTrol = Enable
```

2 Configure one end of the serial link as data terminal equipment (DTE) using:

```
SETDefault !<path> -LAPB InterfaceType = DTE
```

3 Configure the other end of the serial link as data communications equipment (DCE) using:

```
SETDefault !<path> -LAPB InterfaceType = DCE
```

Frame Relay Configuration Options

When you set up data compression for Frame Relay links, data compression can be specified for each individual data link connection identifier (DLCI). The -FR COMPressType parameter overrides the -PORT COMPressType parameter.

Over Frame Relay connections, data compression is not negotiated. Data compression must be configured appropriately at both ends of the link.

- To enable per-packet data compression on a specific DLCI in an Frame Relay link, use:

  ```
  SETDefault !<port> -FR COMPressType = <dlci> PerPacket
  ```

- To disable data compression on a specified DLCI, use:

  ```
  SETDefault !<port> -FR COMPressType = <dlci> NONE
  ```

- To use the compression type configured by the -PORT COMPressType parameter, use:

  ```
  SETDefault !<port> -FR COMPressType = <dlci> DEFault
  ```

The DEFault value for the -FR COMPressType parameter in this command requires that the DLCI use the compression type configured on the PORT Service.

X.25 Configuration Options

When you set up data compression for X.25 links, data compression can be specified for individual X.25 profiles. The -PROFile X25COMPressType parameter overrides the -PORT COMPressType parameter.

- To enable per-packet data compression on a specific profile on an X.25 link, use:

  ```
  SETDefault !<profile ID> -PROFile X25COMPressType = PerPacket
  ```

- To enable history data compression on a specific profile on an X.25 link, use:

  ```
  SETDefault !<profile ID> -PROFile X25COMPressType = HIStory
  ```

- To disable data compression on a specified profile, use:

  ```
  SETDefault !<profile ID> -PROFile X25COMPressType = NONE
  ```

- To use the compression type configured for the X.25 link in the PROFile Service, use:

  ```
  SETDefault !<profile ID> -PROFile X25COMPressType = DEFault
  ```

The DEFault value for the -PROFile X25COMPressType parameter in this command requires that the selected profile use the compression type configured on the PORT Service.

Verifying Link-Level Compression Effectiveness

In multiprotocol and mixed-application environments, it may be difficult to achieve a consistent high level of compression. The effectiveness of compression is measured by the ratio of the uncompressed data to the compressed data, also known as the compression ratio. The software can display the number of raw and compressed bytes, which you can use to measure the effectiveness of link-level compression for your environment.

After you have configured link-level compression, you may need to determine its effectiveness for your network environment. To decide whether link-level compression is beneficial on your network, follow these steps:

1 Display the link compression status using:

```
SHow !<port> -PORT LinkCompStat
```

The system will display statistics accumulated since link compression was configured. The following is a sample display:

```
--------Compression Statistics for Port = 4 LCN = 1--------
Owner        = X25
CompType    = PerPacket
TX_Raw      = 742
TX_Comp     = 200
TX_Ratio    = 3.71
RX_Raw      = 452
RX_Comp     = 162
RX_Ratio    = 2.79
TX_Fail     = 0
RX_Err      = 0
```

If you want to recalculate link compression performance, old statistics can be removed using:

```
FLush !<port> -PORT LinkCompStat
```

2. Compare the number of raw bytes to compressed bytes for both the transmit and receive sides; check all ports that are currently using compression.

   Compression is very CPU-intensive. You may want to disable compression if you do not get a favorable compression ratio, which depends on the nature of the data, or if the overall system performance suffers because of CPU overloading.

How Data Compression Works

Data compression performs additional processing on the contents of each packet to look for repetitive patterns. Consequently, it is most effective when there is sufficient CPU cycles available to handle the additional processing. Data compression is most effective on slow lines.

If you are using a modem that already performs compression, 3Com suggests that you do not configure data compression.

Tinygram Compression

Tinygram compression is packet-level compression. Tinygram compression is performed on packets with a length of less than 64 bytes. An increase in effective throughput is achieved by suppressing the transmission of trailing nulls, or hexadecimal zeros, in packets that are encapsulated in the Ethernet frame format. This type of compression is called tinygram compression and is also referred to as local area transport (LAT) compression (since LAT packets are typically small in size and are padded up to 64 bytes with trailing nulls). The receiving end of a compressed packet can easily recreate the original packet by adding the trailing nulls. Because the CPU cycles involved in the stripping and adding of the trailing nulls are significantly less than the time it takes to transfer those nulls across a slow-speed line, the effective throughput of the system is increased.

Link-Level Compression

Link-level compression is performed over all packets sent on a specified link. The effective throughput is increased by sending fewer bytes across the link, as with tinygram compression. The algorithm used for link-level compression looks for repetitive data patterns in packets and replaces them with shorter length codes.

The software supports the following types of link-level compression algorithms:

- History-based
- Per-packet

The algorithm used for history-based link-level compression looks for repetitive data patterns across multiple packets and replaces them with shorter length codes. The sending and receiving ends both build up a history buffer, and encode and decode the data in the packet according to that buffer. The history buffer will have the last 2 KB of data. For proper encoding and decoding, the history buffer at each end must always be synchronized. Because the history information is transferred along with compressed data, the sending side must be assured that the receiving side reliably gets the data. As a result, history-based compression can operate *only* over a reliable data link. History-based compression requires that the LAPB Service be configured and operational over all the links on which this type of compression is desired. By default, the history-based link-level compression is selected.

The algorithm used for per-packet link-level compression looks for repetitive patterns within a packet and replaces them with shorter length codes. With per-packet compression, the sending and receiving ends do not preserve the history between packets. As a result, per-packet compression does not need to operate over a reliable data link, and the LAPB Service does not need to be configured over all links on which this type of compression is desired.

Because history-based compression looks for repetitive data across multiple packets, it is more effective in shrinking a packet size, which includes the line throughput. When considering history-based compression, the memory required to maintain a history buffer (approximately 26 KB of memory per interface) must be considered, particularly if it is enabled on several links. Because a history buffer is not maintained in per-packet mode, the memory requirement is considerably less (24 KB of memory per a fully populated NETBuilder II system) than for history mode.

When To Use Tinygram Compression

The decision about whether to use tinygram compression depends on the characteristics of your system. Some general recommendations, which are based on line speeds of 64 kbps, are provided here.

Consider using tinygram compression in the following situation:

- In environments with small data packets using null character padding (for example, in LAT environments)

Avoid using tinygram compression in the following situations:

- In environments where packets are generally transmitted with enough data to create 64-byte packets requiring no padding
- In environments where small data packets use random data for padding (for example, in some Telnet environments)

When To Use Link-Level Compression

The decision about whether to use link-level compression depends on the characteristics of your system. Some general recommendations, which are based on line speeds of 64 kbps, are provided here.

Consider using link-level compression in the following situations:

- In environments with repetitive patterns in the bit stream being transferred (for example, with file transfers or electronic mail)
- In environments where slow lines (64 kbps or lower) are being used

Avoid using link-level compression in the following situations:

- In environments with patterns in the bit stream that are *not* repetitive (for example, in image files)
- In environments where high-speed lines are being used
- When the overall throughput of a system is already below normal

If you decide to use link-level compression, you must further decide which type to use: history-based or per-packet. Some general recommendations, which are based on line speeds of 64 kbps, are provided here.

Consider using history-based link-level compression in the following situations:

- In remote sites with 1–2 LAN ports and 1–2 WAN ports or in central sites with minimal LAN traffic and 3–6 WAN ports
- In sites where the wide area links are considered reliable

Consider using per-packet link-level compression in the following situations:

- In central sites with more than 6 WAN ports
- In any site where there is significant LAN-to-LAN traffic

Use the preceding recommendations as general guidelines. In cases where link compression is used, verify the effectiveness of the algorithm with the -PORT LinkCompStat parameter. If you do not see a significant difference between the raw and compressed bytes, the serial line throughput increase may not be enough to offset the overhead of applying the algorithm. If overall performance degradation is experienced, you should reevaluate the continued use of link compression.

40 SCHEDULING AND EVENT-BASED MACRO EXECUTION

This chapter describes how to schedule repetitive events to occur on specific days or dates using the SCHeduling Service and how to set up an automatic back-up for a port using Event-based Macros Execution (EBME).

For conceptual information, refer to "How the Scheduler Works" on page 40-4 and "How EBME Works" on page 40-5.

Creating Schedules

This section describes how to create schedules.

Defining a Daily Schedule

You must define at least one daily schedule before you can create an active schedule.

Create a daily schedule and define its first event in one step using:

```
ADD -SCH EVent <daily schedule> <hh:mm> <command-string>
```

You can add additional events to the daily schedule using the same syntax, where <daily schedule> is the name of a daily schedule for which at least one step has already been defined.

Creating an Active Schedule

After the daily schedule is complete, you must assign it to a calendar date or to a day of the week using:

```
ADD -SCH ActiveSCHedule <mm/dd | SUN | MON | TUE | WED | THU |
  FRI | SAT > <daily schedule>
```

The scheduler is only active when two conditions are met:

- The -SCH CONTrol parameter must be set to Enabled.
- The -SCH CONTrol parameter must be set to RealTimeClock.

The CONTrol parameter also allows you to choose the hardware clock or software clock and select or deselect the logging function. For a complete summary of settings for the CONTrol parameter, refer to Chapter 50 in *Reference for NETBuilder Family Software*.

Enable the scheduler by entering:

```
SETDefault -SCH CONTrol = Enabled
```

Executing Macros Using the Scheduler

The scheduler is a batch-oriented utility. Always test macros thoroughly before submitting them to the scheduler. The scheduler submits the macro to the system without evaluating it and does not report success or failure (even if logging is enabled, the scheduler reports only that the macro was submitted).

Be sure that any macros submitted to the scheduler contain:

- No infinite loops.
- No input variables.
- No illegal commands.
- Minimal output message.

CAUTION: *The only way to stop a macro that contains an infinite loop is to reboot the system.*

Scheduling WAN Connections

You can create a daily schedule to establish a dial-up connection and hang up the line at a specified times, and assign this daily schedule to one or more calendar dates or days of the week.

To use scheduled dial-up, follow these steps:

1 To create a daily schedule that establishes a dial-up connection, use:

```
ADD -SCH EVent <daily schedule> <hh:mm> DIal !<port> "<telephone
  number>"
```

For example, to define the daily schedule, "Daily," that establishes a connection to the telephone number 555-1212 on port 3 at 11 a.m., enter:

```
ADD -SC EVent daily 11:00 DIal !3 "5551212"
```

2 To add to the daily schedule an event that hangs up the connection, use:

```
ADD -SCH EVent <daily schedule> <hh:mm> HangUp !<port>
```

For example, to add to the daily schedule, "Daily," a hangup event at 3 p.m., enter:

```
ADD -SCH EVent daily 15:00 HangUp !3
```

3 To assign the daily schedule, "Daily," to each weekday, enter:

```
ADD -SCH ActiveSCHedule MON DAILY
```

Repeat this step four times, substituting for "MON" the remaining weekday designators: "TUE," "WED," "THU," and "FRI."

Executing Event-based Commands/Macros

EBME provides automatic back-up if a connection fails between two sites when a primary link goes down. EBME also provides loopback detection and recovery. This is a port-based service.

For conceptual information, refer to "How EBME Works" on page 40-5.

EBME provides you with the following features:

- User-defined command or macro configuration.
- A backup action that is executed when the status of a port changes. EBME can back up any kind of port including static, virtual, parent, and dial ports.
- An action that is executed when a port is in a loopback condition. EBME can disable the port and then re-enable the port after a defined delay.

- A user-configurable debounce timer that provides a delay before the command or macro is executed. This timer is provided to prevent the software from reacting to transient changes in the port status.
- A log to track system command or macro execution.
- User control to enable or disable this service.

EBME is also a batch-oriented utility. Always test macros thoroughly before submitting them to EBME. EBME submits the macro to the system without evaluating it and does not report success or failure (even if logging is enabled, the scheduler reports only that the macro was submitted).

Be sure that any macros submitted to the EBME contain:

- No infinite loops.
- No input variables.
- No illegal commands.
- Minimal output message.

CAUTION: *The only way to stop a macro that contains an infinite loop is to reboot the system.*

Setting Up a Backup Port

To configure EBME to bring up a port when a primary link fails, follow these steps:

1 Configure the command or macro for the port being backed up by entering:

```
ADD !2 -SCH EbmeEVent PortDown 30 DO port2down_macro
```

When port 2 fails, this command causes the EBME Service to execute the port2down_macro file and sets the debounce timer to 30 seconds. The port2down_macro brings up the backup port you specify in the macro.

2 Enable the EBME Service by entering:

```
SETDefault -SCH EbmeCONTrol = Enable
```

3 Set the Log option to record the commands or macros that are executed in the system log buffer by entering:

```
SETDefault -SCH EbmeCONTrol = Log
```

Hanging Up a Port

To configure EBME to hang up a backup port when the primary link becomes active again, enter:

```
ADD !2 -SCH EbmeEVent PortUp HangUp !3
```

When port 2 comes up, this command causes the EBME Service to execute the HangUp command for port 3.

Recovering from Port Loopback

The Spanning Tree Protocol (STP) is designed to prevent bridges from forming a loop in active paths. However, STP does not prevent data from being looped back into the port that emitted it. This situation may cause an incorrect station hop and/or broadcast storm problems.

EBME can be used to detect a port loopback condition. A user-defined command or macro can be executed when a loopback condition occurs.

To configure EBME to recover from a port loopback condition, follow these steps:

1 To configure EBME to recover from a loopback condition on port 2, define the port recovery macro, port2LBmacro, by entering:

```
define port2LBmacro =
 setd !2 -port CONT=Disable
 pause 300
 setd !2 -port CONT=Enable
 )
```

This macro checks every five minutes (300 seconds) to see if the loopback condition is still happening. When the loopback is gone, the port is returned to the forwarding state.

2 Enable the Spanning Tree Protocol by entering:

```
SETDefault -STP CONTrol = Enable
```

3 Configure the macro for the port by entering:

```
ADD !2 -SCH EbmeEvent LoopBack DO port2LBmacro
```

4 Enable the EBME Service entering:

```
SETDefault -SCH EbmeCONTrol = Enable
```

***CAUTION:** Keep in mind that the event command string is executed sequentially. A Pause command delays another event from being executed even if the other event occurs concurrently. 3Com recommends that the Pause command be used in loopback events only.*

How the Scheduler Works

The SCHeduling Service allows you to schedule repetitive events to be executed on a specified calendar date each year or on a specified day of each week. Scheduled events can consist of commands or macros. Used with the dial-up feature, the scheduler allows you to perform many useful tasks, including:

- Matching traffic prioritization with work schedules.
- Updating configuration and test-booting off-hours.
- Allowing for time zone and work habit differences, and giving several remote sites scheduled access to a single WAN port at the central site.
- Creating a more secure Internet with remote sites having scheduled access and only central site originating calls.
- Synchronizing the mail server and WAN link to control cost of electronic mail distribution.

The scheduler allows you to define up to 12 *daily schedules* and to assign each daily schedule to one or more calendar dates or days of the week. Each daily schedule contains one or more events, each of which consists of a time of day and a command or macro to be executed.

An *active schedule* is the combination of a calendar day or day of the week with a daily schedule. On any given day, only one active schedule can be in use.

If an active schedule exists for today’s calendar date (for example, March 31) and an active schedule exists for today’s day of the week (for example, Wednesday), the active schedule for the calendar date takes precedence.

How EBME Works

Event-based Command/Macro Execution is a platform routing protocol and provides a backup link when the primary link between two sites fails. It also provides port loopback detection EBME supports a port-level backup to a primary link failure and loopback detection and recovery.

You can use EBME to configure commands or macros for execution when the status of a given port changes from UP to DOWN, from DOWN to UP or LOOPBACK condition. You can configure commands and macros to contain the instructions to bring back the primary connection if an UP to DOWN event occurs on the port. You can also configure commands and macros to be executed when a loopback event occurs.

EBME consists of event generators, a port-based event responder, and an event and action database. The EBME Service provides the interface for these components and for the bridge/router software.

When the service is enabled to log all events, the sequence of events to execute the action you configured is as follows:

- When the port is configured, with some EBME events, its status is monitored by EBME.
- The port status, either up, down or loopback is recognized by the port-based event responder from the event generator. Each port may have no more than three events, up, down or loopback.
- The port up/down event becomes valid when the debounce timer runs out.
- When the event responder determines that the event has occurred, it queues the event and selects the appropriate action from the event and action database.
- The action is scheduled to take place.
- When the action has completed, a message is written to the log file stating that the action is complete.

Only the first 80 characters of the output from the UI command are printed, while only the macro name and a complete or incomplete message are printed.

41 PRIORITIZING MULTIPROTOCOL DATA

This chapter describes how to use the data prioritization feature to assign a priority (high, medium, or low) to most packets that are forwarded over a wide area network using Point-to-Point Protocol (PPP), Phone Line Gateway (PLG), Frame Relay, or Switched Multimegabit Data Service (SMDS).

For conceptual information, refer to "How Data Prioritization Works" on page 41-6.

Advantages of Prioritizing Data

You can receive the following benefits by using the data prioritization feature:

- Control data traffic on heavily used wide area networks.
- Allow the following types of packets to have a higher priority over other data traffic on a wide area network:
  - Network-critical traffic, for example, bridge spanning tree packets (system-configured)
  - Mission-critical traffic, for example, Logical Link Control, type 2 (LLC2) tunnel packets (user-configured)
  - Time-critical traffic, for example, Telnet packets (system-configured)
  - Specific protocol packets, for example, Advanced Peer-to-Peer Networking (APPN) or Internet Protocol (IP)-routed packets (user-configured)
- Allow sessions to be prioritized according to the session characteristics.
- Avoid LLC, Systems Network Architecture (SNA), and NetBIOS session failures due to timeouts.

Setting Up Data Prioritization

This section describes how to set up the data prioritization feature.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Determine what types of packets you want to prioritize and what priority you want to assign to each type.

  For example, you may want to assign IP-routed packets a high priority, Internetwork Packet Exchange (IPX) packets a medium priority, and AppleTalk packets a low priority. You also may want to prioritize types of packets within the IP protocol itself.
- Read through this chapter to familiarize yourself with the different ways you can assign a priority to a type of packet. Determine which option you want to use for each type of packet you want to prioritize.

For example, you may want to assign a high priority to IP-routed packets, a medium priority to IPX packets by setting up a mask and policy, and a low priority to all other packets, including AppleTalk packets.

- Determine the interleave factor, which is defined as the ratio of packets that you want forwarded from high- to medium-priority queues and the ratio of packets you want forwarded from medium- to low-priority queues. For more information on the interleave factor and queue arbitration, refer to "How Data Prioritization Works" on page 41-6.

Procedure To set up the data prioritization feature, follow these steps. The example of assigning a high priority to IP-routed packets, a medium priority to IPX packets, and a low priority to all other packets including AppleTalk packets will be used throughout this procedure.

1 If you want to assign a priority to APPN, LLC2 tunnel, or IP-routed packets, use one of the following lines of syntax:

```
SETDefault -APPN QueuePriority = <H | M | L | DEFault>
SETDefault -LLC2 TUNnelPRiority = <H | M | L | DEFault>
SETDefault -IP QueuePriority = <H | M | L | DEFault>
```

For example, to assign a high priority to IP-routed packets, enter:

```
SETDefault -IP QueuePriority = H
```

If you retain the default setting of DEFault for any of the above commands, the system uses the setting of the -PORT DefaultPriority parameter. For instructions on configuring the -PORT DefaultPriority parameter, refer to step 3.

2 If you want to assign a priority to packets other than LLC2 tunnel or IP-routed packets, follow these steps:

a Set up a mask that determines the types of packets that should be prioritized.

You can set up either a built-in or a user-defined mask. Since using a built-in mask requires less configuration, 3Com recommends this option. To determine if a built-in mask exists for the type of packet you want to prioritize, refer to Chapter 4. If a built-in mask that suits your purposes exists, go on to step 2b.

In the example used throughout this procedure, you want to assign a medium priority to IPX packets. Since a built-in mask for IPX packets exists, you do not need to create a mask.

If a built-in mask that suits your needs does not exist, configure one using:

```
ADD -FIlter MASK <maskname> <location> [<operation>] <pattern>
```

For example, to configure a mask that prioritizes packets with a value greater than %45 at the first byte of data, enter:

```
ADD -FIlter MASK some_data dl.data+%0>%45
```

For more examples of configuring various masks, refer to Chapter 4.

b Set up a policy that determines the priority each type of packet should be assigned using:

```
ADD -FIlter POLicy <policyname> <action> <masks> [<context>]
```

For example, to prioritize IPX packets at a medium priority using a built-in mask, enter:

```
ADD -FIlter POLicy prioritize_ipx PRIoritize M IPX
```

3 The default priority for packets that you do not specifically assign a priority to (as in steps 1 and 2) is medium. If you want to change this priority, use:

```
SETDefault -PORT DefaultPriority = <H | M | L>
```

For example, to change the default priority for all packets other than IP and IPX, including AppleTalk packets, enter:

```
SETDefault -PORT DefaultPriority = L
```

4 The default ratio of packets forwarded from high- to medium-priority queues is 3 and the ratio of packets forwarded from medium- to low-priority queues is 2. If you want to reconfigure these ratios, use:

```
SETDefault !<port> -PORT QueueInterLeave = <ratio1> <ratio2>
  (1–10)
```

For example, to change the default values to 6 and 3 on port 3, enter:

```
SETDefault !3 -PORT QueueInterLeave = 6 3
```

For more information on forwarding ratios and queue arbitration, refer to “How Data Prioritization Works” on page 41-6.

5 By default, the serial line driver will accept as many medium- and low-priority packets as it can possibly handle. If you anticipate that high-priority packets, such as SNA packets, may be slowed down by many medium- and low-priority packets, (especially large medium- and low-priority packets) in the queue, adjust the number of medium- and low-priority packets forwarded to the queue using:

```
SETDefault !<port> -PORT QueueThrottle = <1–40>
```

3Com recommends setting the value between 1 and 20 if you are using a slow-speed line (64K or below) and between 21 and 40 if you are using a high-speed line (for example, T1).

For more information on the parameters used in this procedure, refer to Chapter 23, Chapter 29, Chapter 34, and Chapter 43 in *Reference for NETBuilder Family Software*.

Prioritizing LLC2-, SNA-, and NetBIOS-Bridged Packets

This section provides information on assigning a priority to the following types of packets using the data prioritization feature:

- LLC2-bridged packets from two groups of end stations (one group simulates 3270 interactive traffic; the other, SNA file transfers)

  The setting of the -LLC2 TUNnelPRiority parameter on bridge/routers located at both ends of the tunnel should be the same. For example, if the setting of the -LLC2 TUNnelPRiority parameter on bridge/router 1 located on one end of the tunnel is high, then the setting of this parameter on bridge/router 2 located on the other end of the tunnel should also be high.

- SNA-bridged packets
- NetBIOS-bridged packets

To assign a priority to these types of packets, you need to set up a filter, which includes a mask and a filter policy. The mask specifies the type of packet that should be prioritized; the filter policy determines the priority that the specified packet should be assigned.

The following sections provide more information on assigning priorities to these types of packets.

CAUTION: *Do not prioritize connection-oriented packets such as LLC2 (SNA, NetBIOS, etc.) to the low queue because the low queue can be flushed to favor high and medium packets. With connection-oriented packets such as LLC2, REJects and possible session disconnects will be generated.*

Prioritizing LLC2-Bridged Packets From Two Groups of End Stations

Suppose you want to assign LLC2-bridged packets from end station group 1 (3270 interactive traffic) a high priority and LLC2-bridged packets from end station group 2 (SNA file transfers) a medium priority. You also want to assign all other packets a low priority.

Since you want to prioritize a certain type of packet received from two groups of end stations, you need to define each group by identifying each end station that belongs to a group using the -FIlter StationGroup parameter. For example, if groups 1 and 2 are composed of three end stations each, enter:

```
ADD -FIlter StationGroup group_1 %0800020000a1
ADD -FIlter StationGroup group_1 %0800020000a2
ADD -FIlter StationGroup group_1 %0800020000a3
ADD -FIlter StationGroup group_2 %0800020000b1
ADD -FIlter StationGroup group_2 %0800020000b2
ADD -FIlter StationGroup group_2 %0800020000b3
```

Next you need to set up a mask and a filter policy for the LLC2-bridged packets received from end systems in groups 1 and 2. To set up a mask called “inter” that looks for bridged LLC2 packets from end systems in group 1 and a mask called “ft” that looks for bridged LLC2 packets from end systems in group 2, enter:

```
ADD -FIlter MASK inter dl.sa = group_1
ADD -FIlter MASK ft dl.sa = group_2
```

To set up a filter policy called “interhigh” that assigns a high priority to packets specified in the mask called “inter” and a filter policy called “ftlow” that assigns a low priority to packets specified in the mask called “ft,” enter:

```
ADD -FIlter POLicy interhigh PRIoritize H inter
ADD -FIlter POLicy ftmed PRIoritize M ft
SETDefault -PORT DefaultPriority = L
```

Because you changed the port default priority to low, all other packets are assigned a low priority.

Prioritizing SNA- and NetBIOS-Bridged Packets

Suppose you want to assign SNA-bridged packets a high priority and NetBIOS-bridged packets a medium priority. You also want to assign all other packets a low priority.

You need to set up a mask and a filter policy for the SNA- and NetBIOS-bridged packets. To set up a mask called “sna” that looks for bridged SNA packets and a mask called “netbios” that looks for bridged NetBIOS packets, enter:

```
ADD -FIlter MASK sna dl.lsap = %4
ADD -FIlter MASK netbios dl.lsap = %f0
```

To set up a filter policy called “snahigh” that assigns a high priority to packets specified in the mask called “sna” and a filter policy called “nbmed” that assigns a medium priority to packets specified in the mask called “netbios,” enter:

```
ADD -FIlter POLicy snahigh PRIoritize H sna
ADD -FIlter POLicy nbmed PRIoritize M netbios
SETDefault -PORT DefaultPriority = L
```

Because you changed the port default priority to low, all other packets are assigned a low priority.

Assigning a Priority to Different IP Packets

You can use the IP filter facilities in the IP Service to prioritize IP traffic over the traffic of other protocols or to prioritize various types of IP traffic. By using the -IP FilterAddrs parameter, you can specify a packet filtering policy. Use the Qpriority, X25Profile, and DodDiscard actions of the FilterAddrs parameter for data prioritization among IP packets.

For example, suppose you want to assign a high priority to Telnet packets going to host 129.0.0.1. The socket number used by the Telnet protocol is 23. Enter:

```
ADD -IP FilterAddrs ALL> 129.0.0.1 QPriority High 23
```

For more information about the FilterAddrs parameter, refer to Chapter 29 in *Reference for NETBuilder Family Software.* For more examples using the Qpriority, X25Profile, and DodDiscard actions of the FilterAddrs parameter, refer to Chapter 6.

Data Prioritization Parameters

Table 41-1 briefly summarizes the available prioritization parameters and the services where they exist.

Table 41-1 Data Prioritization Parameters

Service	Parameter	Description
FIlter	MASK	Sets up a mask that determines which packets should be prioritized.
	POLicy	Sets up a policy that determines the priority a particular type of packet is assigned.
APPN IP PORT	QueuePriority	The -APPN and -IP QueuePriority parameters set the priority of APPN and IP-routed packets, respectively; if this parameter is set to DEFault, the system uses the setting of the -PORT DefaultPriority parameter. The -PORT QueuePriority parameter displays the settings of the following parameters: ■ -APPN QueuePriority ■ -IP QueuePriority ■ -LLC2 TUNnelPRiority ■ -PORT DefaultPriority
IP	FilterAddrs	The QPriority option for the FilterAddrs parameter specifies a queue priority value of high, medium, or low. A numerical value specifies an X25 profile ID to be used. For more information on prioritizing packets over the X25 Service, refer to Chapter 45 in *Reference for NETBuilder Family Software* and to Chapter 45 in this guide.

Table 41-1 Data Prioritization Parameters (continued)

Service	Parameter	Description
LLC2	TUNnelPRiority	Sets the priority of LLC2 packets tunneled over an IP internetwork; if this parameter is set to DEFault, the system uses the setting of the -PORT DefaultPriority parameter. The priority of LLC2 tunnel packets is maintained across 3Com bridge/routers that the packets traverse through the use of the type of service (TOS) field in the IP header.
(continued)		
PORT	DefaultPriority	Sets the default priority of packets if one of the following conditions apply: ■ The -APPN QueuePriority parameter, the -IP QueuePriority parameter, or the -LLC2 TUNnelPRiority parameter is set to DEFault. ■ A mask and prioritization policy is not configured for a particular type of packet.
	QueueInterLeave	Sets and displays the interleave factor, which is defined as the forwarding ratio of high- to medium-priority packets and of medium- to low-priority packets.
	QueuePATtern	Displays the interleave factor configured by the -PORT QueueInterLeave parameter translated by the system into a high, medium, and low pattern. A ratio based on the high, medium, and low pattern also displays.
	QueueThrottle	Controls the number of medium- and low-priority packets that are forwarded to the driver each time packets from the priority queue are forwarded on to the wide area network.
PROFile	X25PacketSiZE X25PROFileType X25VCLimit X25VCQueueSize X25VCThruputClass X25VCTimer X25WindowSiZe	These parameters help you prioritize traffic if you are using the X25 Service. For more complete information on these parameters, refer to Chapter 45 in this guide and Chapter 45 in *Reference for NETBuilder Family Software*.

For more information on these parameters, refer to Chapter 23, Chapter 29, Chapter 34, and Chapter 43 in *Reference for NETBuilder Family Software*.

How Data Prioritization Works

The bridge/router software implements a prioritization scheme that assigns a priority to each packet then forwards the packet to an urgent-, high-, medium-, or low- priority queue. The packets are then forwarded from the queues onto a wide area network using PPP, PLG, Frame Relay, or Switched Multimegabit Data Service (SMDS) in an order controlled by a queue arbitration algorithm.

The system assigns a priority to some packets. Table 41-2 lists these packet types and the priority assigned to them by the system. You cannot reconfigure the priority of these packets.

Table 41-2 Packets with System-Assigned Priorities

Packet Type	Priority Level
Bridge Spanning Tree	Urgent
DECnet routing update	High
OSPF	High
Telnet	High

The system assigns a priority to Telnet packets that originate from the box; it does not assign a priority to routed Telnet packets.

You can assign a priority to all other types of packets that are not listed in Table 41-2. You can assign priorities to different types of packets in the following ways:

- For APPN, LLC2 tunnel and IP-routed packets, you can assign a high, medium, or low priority using the SETDefault -APPN QueuePriority, SETDefault -LLC2 TUNnelPRiority, or SETDefault -IP QueuePriority commands, respectively.

- For all packets other than LLC2 tunnel and IP-routed packets, for example, AppleTalk packets, you can set up a mask that determines what packets should be prioritized and a policy that determines what priority the packets should be assigned.

Any packet not specifically assigned a priority receives its priority from the setting of the -PORT DefaultPriority parameter. The default setting of this parameter is medium.

When multiple ports are attached to one path, no one particular port receives a higher priority over another port. All ports attached to one path receive the same priority.

For complete information on assigning priorities to packets, refer to "Setting Up Data Prioritization" on page 41-1. For more information on the parameters described in this section, refer to Chapter 5, Chapter 29, Chapter 34, and Chapter 43 in *Reference for NETBuilder Family Software*.

How Packets Are Assigned a Priority

All packets are assigned either an urgent, high, medium, or low priority before they are transmitted over a wide area interface.

The system assigns a priority to a packet using the following process:

1. The system assigns a priority to certain packet types (refer to Table 41-2). You cannot change the priority assigned to these packets.
2. The system assigns a priority to APPN, IP-routed, and LLC2 tunnel packets with the settings of the -APPN QueuePriority, -IP QueuePriority, and -LLC2 TUNnelPRiority parameters. You can change the default settings of these parameters.
3. The system assigns a priority to all packets other than APPN, IP-routed, and LLC2 tunnel packets through a prioritization filter.

   You set up a prioritization filter by configuring masks that determine the types of packets that should be prioritized using the ADD -FIlter MASK command. A configuration policy that determines the priority each type of packet should be assigned using the ADD -FIlter POLicy command.
4. The system assigns a priority to all other packets that do not receive their priority through any of the previously discussed methods via the setting of the -PORT DefaultPriority parameter. You can change the default setting of this parameter.

   After the system assigns a priority to a packet using one of steps described above, the packet is forwarded to a queue for transmission to a wide area network. For example, if a packet is assigned a priority using the method described in step 2, the packet does not undergo the methods described in steps 3 and 4. The system handles packet priority assignment in this way to reduce the number of packets that are sent through the prioritization filter (step 3) because filtering can impact system performance.

Queues

After the system assigns a priority to a packet automatically or through configured or default parameter settings, the packet is forwarded to one of four types of queues: urgent, high, medium, or low. Figure 41-1 shows how particular types of packets are assigned priorities and how they are forwarded to the various queues. This figure assumes that a user has configured:

- A medium priority for IP-routed packets.
- A high priority for LLC2 tunnel packets.
- A filter that assigns a medium priority to IPX packets.
- A filter that assigns a low priority to AppleTalk packets.
- The value of low for the -PORT DefaultPriority parameter.

The other packets are assigned their priorities by the system.

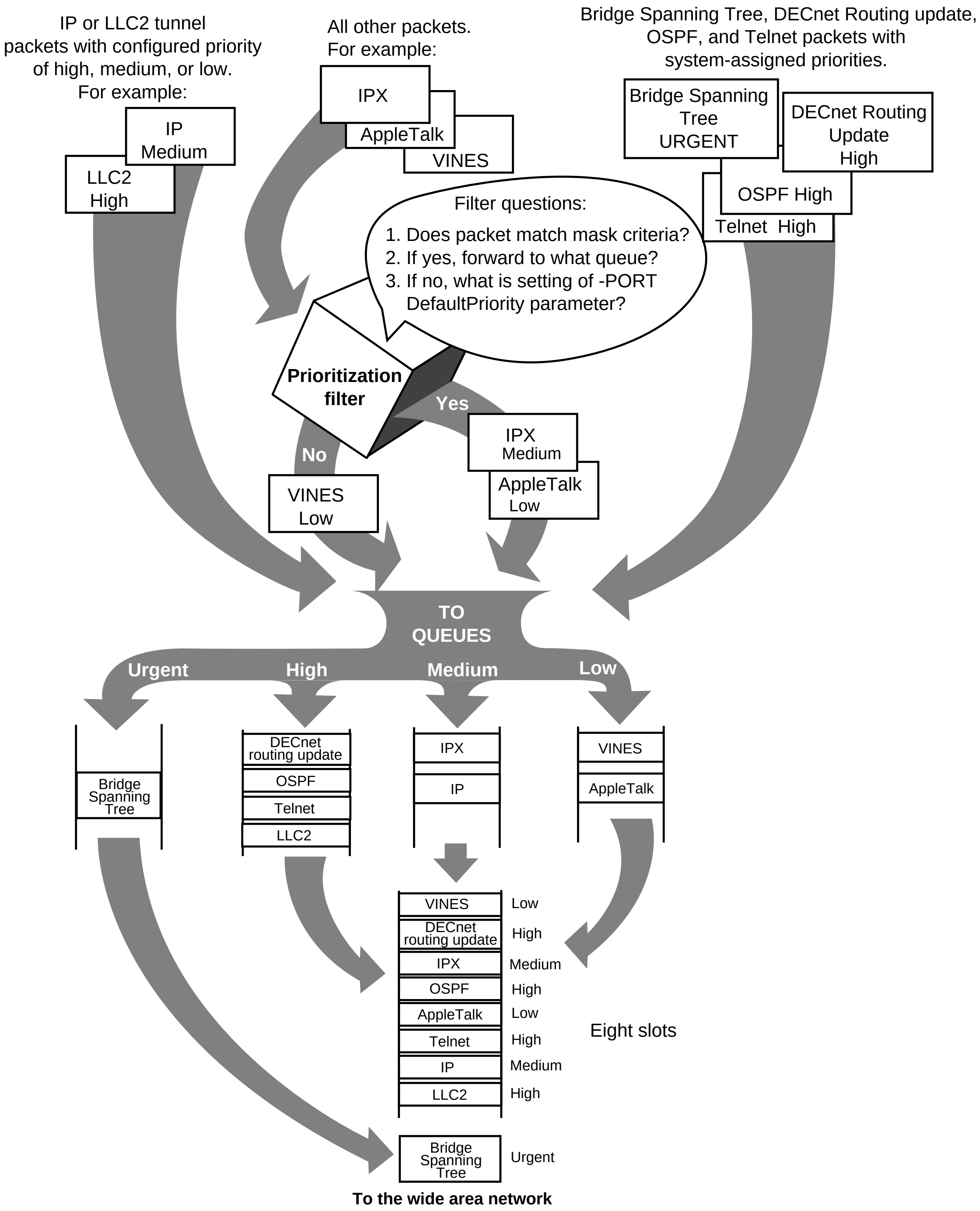

IP or LLC2 tunnel packets with configured priority of high, medium, or low. For example:
IP Medium
LLC2 High
All other packets. For example:
IPX
AppleTalk
VINES
Bridge Spanning Tree, DECnet Routing update, OSPF, and Telnet packets with system-assigned priorities.
Bridge Spanning Tree URGENT
DECnet Routing Update High
OSPF High
Telnet High
Filter questions:
1. Does packet match mask criteria?
2. If yes, forward to what queue?
3. If no, what is setting of -PORT DefaultPriority parameter?
Prioritization filter
Yes
No
IPX Medium
AppleTalk Low
VINES Low
TO QUEUES
Urgent
High
Medium
Low
Bridge Spanning Tree
DECnet routing update
OSPF
Telnet
LLC2
IPX
IP
VINES
AppleTalk
VINES Low
DECnet routing update High
IPX Medium
OSPF High
AppleTalk Low
Telnet High
IP Medium
LLC2 High
Eight slots
Bridge Spanning Tree Urgent
To the wide area network

Figure 41-1 Prioritizing and Forwarding Packets to Queues

The system forwards all packets in the urgent-priority queue before packets in the high-, medium-, and low-priority queues. You cannot assign the urgent priority to a packet. The system automatically assigns an urgent priority to bridging spanning tree packets only.

After all packets in the urgent-priority queue are forwarded, the system forwards packets from the high-, medium-, and low-priority queues according to a queue arbitration algorithm. For more information on this algorithm, refer to the next section. The system automatically assigns a high priority to DECnet routing update, open shortest path first (OSPF), and Telnet packets and forwards these types of packets to the high-priority queue.

Queue Arbitration Algorithm

Instead of forwarding all packets from high-priority queues, then all packets from medium-priority queues, and so on, the system uses a queue arbitration algorithm, which ensures that the high-, medium-, and low-priority queues are serviced according to an interleave factor.

You can configure the interleave factor by entering:

```
SETDefault -PORT QueueInterLeave
```

This command allows you to set up the forwarding ratio of high- to medium-priority packets and of medium- to low-priority packets.

The algorithm implements an 8-slot queue that is composed of a variable number of high-, medium-, and low-priority slots as shown in Figure 41-1. Urgent packets bypass this queue and are immediately forwarded.

The number of high-, medium-, and low-priority slots in the queue are based on the setting of the -PORT QueueInterLeave parameter. The system chooses the closest of five possible 8-slot patterns. Table 41-3 lists sample values of the -PORT QueueInterLeave parameter and the corresponding 8-slot pattern selected by the system.

Table 41-3 -PORT QueueInterLeave Parameter Values and 8-Slot Patterns

Value of -PORT QueueInterLeave	Corresponding 8-Slot Pattern
6, 1	HHMHHLHH
3, 2*	HHMHMHLH
2, 2	HMHMHLHM
2, 1	HMHLHMHL
1, 1	HMHLHMLM

* Default value.

The actual transmission rate of high, medium, and low packets on a WAN link may not exactly match the 8-slot pattern because of the packet receive rate and receive pattern. For example, only medium and low packets may arrive in one window of time, and if no high packets are available to send at this time, then the medium and low packets are sent out on the serial line. The low and medium transmission queue can get flushed to favor high-priority packets. For example, in a situation where the packet receive rate is much higher than the WAN link can handle, many packets will be dropped, and with the default queue arbitration pattern of 5 high, 2 medium, and 1 low, the actual transmission ratio may be 6 high, 1 medium, and 1 low.

To display the 8-slot pattern selected by the system, enter:

```
SHow -PORT QueuePATtern
```

A ratio based on the pattern also is displayed. For example, for the default value of the -PORT QueueInterLeave parameter (3, 2), the following display appears:

```
HHMHMHLH (5:2:1)
```

The contents of this display are based on the setting of the -PORT QueueInterLeave parameter and are the results of a translation algorithm. The contents of this display are calculated by the system and cannot be reconfigured.

This display indicates that the first and second packets are sent from the high-priority queue, the third packet is sent from the medium-priority queue, and so on. Once the eighth packet is sent, the algorithm wraps to the beginning of the pattern again and the first and second packets are sent from the high-priority queue, and so on.

If a packet is sent from the high-priority queue but that particular queue is empty, a packet from the medium-priority queue is sent instead; if the medium-priority queue is also empty, a packet from the low-priority queue is sent instead. If a packet is sent from the medium-priority queue but that particular queue is empty, a packet from the high-priority queue is sent instead. If a packet is sent from the low-priority queue but that particular queue is empty, a high-priority packet is sent instead.

More information on the commands discussed in this section is provided in Chapter 43 of *Reference for NETBuilder Family Software*.

42 WIDE AREA NETWORKING USING FRAME RELAY

This chapter describes how to configure your bridge/router to establish serial line connectivity through Frame Relay. It also describes how this wide area protocol works and gives guidelines for operating and managing it.

The Asynchronous Transfer Mode data exchange interface (ATM DXI) on the bridge/router operates as part of the Frame Relay service. Most of the procedures in this chapter for configuring Frame Relay can also be used to configure the ATM DXI. For more information about ATM DXI, refer to Chapter 43.

For conceptual information, refer to "How Frame Relay Works" on page 42-22.

Setting Up the Frame Relay Service

This section describes how to configure your bridge/router to transmit and receive data over a Frame Relay interface.

You must follow the steps in this section whether you are configuring for bridging or for routing. After you have completed these steps, proceed to "Setting Up Basic Bridging over Frame Relay" on page 42-3 for bridging configuration information or to "Setting Up Basic Routing over Frame Relay" on page 42-4 for routing configuration information.

For detailed descriptions of all commands, refer to *Reference for NETBuilder Family Software.*

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your wide area bridge/router ports and paths according to Chapter 1.
- Determine if your Frame Relay network supports the Local Management Interface (LMI) Protocol. For information about LMI, refer to "Local Management Interface Protocol" on page 42-27.
- Determine if you have a partially meshed or nonmeshed topology.

  If you do, and plan to enable the Internet Protocol (IP), the Routing Information Protocol (RIP), the Internet Packet Exchange (IPX), or AppleTalk routing, you need to make certain that the next-hop split horizon feature is enabled. If you have a partially meshed or nonmeshed topology, and you plan to enable Open Shortest Path First (OSPF), make sure that you set -OSPF CONTrol to NonMesh to enable the point-to-multipoint interface. If you plan to enable bridging, Xerox Network Systems (XNS), VINES, or DECnet IV routing, make sure that you have created virtual ports for each remote network that is attached to a Frame Relay cloud. For information on meshed, partially meshed, and nonmeshed topologies, next-hop split horizon, and virtual ports, refer to "How Frame Relay Works" on page 42-22. For instructions on setting up virtual ports, refer to Chapter 1.

The Frame Relay software in the NETBuilder 6.0 software or later release is not interoperable with Frame Relay software in version 5.0 or earlier.

Procedure To transmit and receive data over a Frame Relay network, follow these steps:

1 Enable the Frame Relay service by setting the owner of the serial interface to Frame Relay using:

```
SETDefault !<port> -PORT OWNer = FrameRelay
```

If PORT OWNer is set to Auto, Frame Relay is detected and configured automatically and this step may not be necessary.

For networks running RIP with the port up and the -RIPIP CONTrol parameter set to TAlk, the CONTrol parameter DynamicNbr option is automatically enabled. With the DynamicNbr enabled, neighbors are automatically added. If DynamicNbr is not enabled, neighbors must be added manually.

2 If your Frame Relay network supports the LMI Protocol, make sure that the appropriate LMI Protocol is enabled. If your Frame Relay network does not support the LMI Protocol, disable this protocol.

The NETBuilder software includes three types of LMI: Consortium LMI, Annex-D LMI, and NTT LMI. Configure the software with the type of LMI that the switching equipment supports. Consortium LMI is configured by specifying LMI; Annex-D LMI is configured by specifying ANsiLMI; NTT LMI is configured by specifying the value NTTLMI.

The type of LMI is determined dynamically if the port is configured for auto detect. To manually enable the specific LMI or to completely disable the LMI Protocol, use:

```
SETDefault !<port> -FR CONTrol = [NoLMI | LMI | ANsiLMI | NTTLMI]
```

3 If you set the value of the -FR CONTrol parameter to NTTLMI, you must set up your bridge/router to control the throughput of a DLCI if the Frame Relay network becomes congested.

Use:

```
SETDefault !<port> -FR DLCIR = <dlci> <cir>
```

Specify DLCIs and committed information rate (CIR) values provided by your Frame Relay service provider. Refer to Chapter 25 in *Reference for NETBuilder Family Software* for more information on this parameter.

CAUTION: *Failure to specify CIR values of the DLCIs causes unpredictable results.*

Verifying the Configuration

To verify the Frame Relay configuration, enter:

```
SHow -FR CONFiguration
```

The router displays current Frame Relay configuration information.

Setting Up Basic Bridging over Frame Relay

This section describes how to configure transparent and source route bridging over Frame Relay.

Configuring Transparent Bridging

This section provides information for configuring transparent bridging over Frame Relay.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 3.
- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.
- If your Frame Relay network does not support the LMI Protocol and you want to bridge to specific DLCI neighbors, obtain DLCI neighbor addresses to add to the DLCI table. For information on DLCI addresses, refer to "How Frame Relay Works" on page 42-22.

Procedure

To configure transparent bridging over Frame Relay, follow these steps:

1 If your Frame Relay network does not support the LMI Protocol or if you want to bridge to specific DLCI neighbors, you must add DLCI neighbors to the static DLCI neighbor table using:

```
ADD !<port> -BRidge DlciNeighbor = <dlci>
```

Even if the LMI Protocol is enabled, you can add DLCI neighbors to the static DLCI neighbor table to bridge to specific DLCI neighbors. Static DLCI neighbors take precedence over neighbors learned dynamically with the LMI Protocol.

If LMI protocol is running consortium LMI, the valid range for subscriber numbers is 16 to 1022. For other LMI protocols, the range is 16 to 991.

2 Verify that transparent bridging has been enabled for the appropriate wide area port or virtual port by entering:

```
SHow -BRidge TransparentBRidge
```

By default, transparent bridging is enabled on all ports. If transparent bridging has been disabled for the wide area port, you can enable it using:

```
SETDefault !<port> -BRidge TransparentBRidge = TransparentBRidge
```

3 Verify that bridging is enabled by entering:

```
SHow -BRidge CONFiguration
```

If bridging has been disabled, enable it for the system by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

Configuring Source Route Bridging

This section provides information for configuring source route bridging over Frame Relay.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 5.
- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.

- Assign a unique ring number for each remote network.
- Assign a bridge number for the bridge.

Procedure

To configure source route bridging over Frame Relay, follow these steps:

1 Assign each wide area port of each bridge/router that is attached to the Frame Relay network the ring number (hexadecimal) of the network it accesses.

Use:

```
SETDefault !<port> -SR RingNumber = <number> (1-4095) | 0x<number>
  (1-FFF)
```

You can enter the ring number in decimal or hexadecimal format. Precede the hexadecimal number with 0x.

2 Verify that source route bridging is enabled on the wide area port using:

```
SHow !<port> -SR SrcRouBridge
```

If source route bridging is disabled, you need to enable it for your wide area port by using:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

3 If you want to run source route and transparent bridging on a NETBuilder II bridge/router, skip this step and go on to step 4. If you want to run source route bridging only on a NETBuilder II bridge/router, disable transparent bridging on the wide area port using:

```
SETDefault !<port> -BRidge TransparentBRidge = NoTransparentBRidge
```

This step does not apply to model 32x and 52x SuperStack II bridge/routers. Transparent bridging is not supported on these models.

4 Verify that bridging is enabled by entering:

```
SHow -BRidge CONFiguration
```

If bridging has been disabled, enable it for the system by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

Setting Up Basic Routing over Frame Relay

This section describes how to configure your router to transmit and receive data over a Frame Relay interface. Procedures for the following routing protocols are provided:

- AppleTalk
- APPN
- DECnet
- IP
- IPX
- OSI
- VINES
- XNS

A router can be configured to simultaneously route multiple protocols over Frame Relay to one or more remote network connections. For example, in Figure 42-1, the local network supports both XNS and TCP/IP traffic and routes information through a single Frame Relay connection to both types of remote networks.

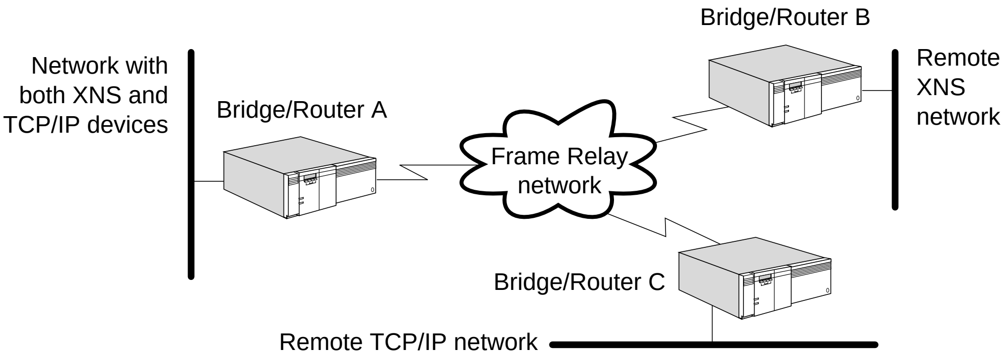


Figure 42-1 Routing Multiple Protocols over Frame Relay Network

Configuring AppleTalk

This section provides information for configuring AppleTalk routing for communication over a Frame Relay network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 14.
- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.
- Determine whether to operate the Frame Relay network as either a non-AppleTalk or AppleTalk network. In both cases, Routing Table Maintenance Protocol (RTMP) packet broadcasts are sent as directed broadcasts every 10 seconds (this is the default) to each neighboring router configured on a port.

  For a non-AppleTalk network configuration, obtain the Frame Relay DLCI addresses representing the virtual circuits to the routers at the remote networks so that you can configure static mapping.

  For an AppleTalk network configuration, obtain the tentative network number and tentative node ID for each of the remote router ports connected to the Frame Relay network. Also obtain the Frame Relay DLCI addresses representing the virtual circuits to the routers at the remote networks so that you can configure static mapping.

For Frame Relay ports, split horizon decisions are made at the next router link level instead of at the port level when more than one neighbor link is associated with a port. Next-hop split horizon allows for support of partially meshed and nonmeshed topologies by allowing a router to use a Frame Relay port as a virtual hub, sending route information to each router out of the port learned from all other routers out of the same port. If the decisions were made at the port level, as is the case for AppleTalk on LANs and Switched Multimegabit Data Service (SMDS), no routing information learned from any router out of the port would be sent to any router out of the same port.

Non-AppleTalk Configuration

To configure AppleTalk routing over a Frame Relay network configured as a non-AppleTalk network, see Figure 42-2 and follow these steps:

1 Configure all the ports on bridge/routers connected to the Frame Relay network to be connected to a non-AppleTalk network.

On bridge/routers A, B, and C, enter:

```
SETDefault !3 -AppleTalk CONTrol = NonAppleTalk
```

2 On each bridge/router, assign the Frame Relay DLCI of the other bridge/routers' ports and virtual ports connected to the network.

For example, on bridge/router A, enter:

```
ADD -AppleTalk ADDRess !3 @33
ADD -AppleTalk ADDRess !3 @44
```

Enter similar address information on bridge/routers B and C.

You can dynamically add and delete neighbors using the ADDRess parameter while a port is enabled and AppleTalk is routing.

3 Enable routing on each AppleTalk bridge/router port attached to the Frame Relay network by entering:

```
SETDefault !3 -AppleTalk CONTrol = ROute
```

AppleTalk Configuration

To configure AppleTalk routing over a Frame Relay network as an AppleTalk configuration, see Figure 42-2 and follow these steps.

The example in the following procedure assumes that the network range for the Frame Relay cloud shared by the configured routers is 2 to 4 and that at least one router is configured to send seed information to any other nonseed routers.

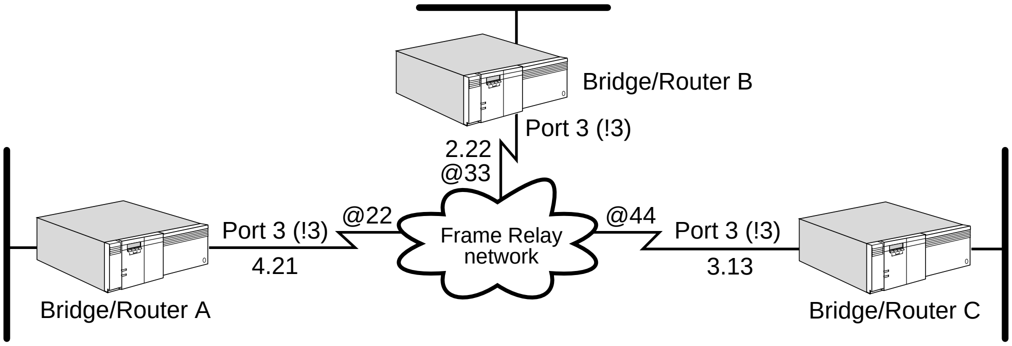


Figure 42-2 Configuring AppleTalk over Frame Relay

1 Specify the tentative network number and the tentative node ID for the specified port that the AppleTalk router uses during dynamic node address acquisition at port enable time.

Use:

```
SETDefault !<port> -AppleTalk StartupNET = <number>(0–65279)
SETDefault !<port> -AppleTalk StartupNODe = <number>(0–253)
```

Using these parameters allows the local router always to assign the same AppleTalk node address to the local port, assuming that the address is within the network range assigned to the Frame Relay cloud. These static configurations are saved nonvolatile storage and only need to be changed when the topology changes.

a For example, before routing is enabled on bridge/router A, enter:

```
SETDefault !3 -AppleTalk StartupNET = 4
SETDefault !3 -AppleTalk StartupNODe = 21
```

b Enter values for the StartupNET and StartupNODe parameters for bridge/routers B and C.

2 Configure static mapping of neighbor DLCIs to their AppleTalk node addresses on the ports and virtual ports of each bridge/router.

For example, on bridge/router A (AppleTalk address 4.21), enter the following DLCI addresses of the other routers connected to the Frame Relay network:

```
ADD -AppleTalk ADDRess 2.22 @33
ADD -AppleTalk ADDRess 3.13 @44
```

Configure static mapping of media addresses on bridge/routers B and C.

The valid range for Frame Relay DLCIs is 16 to 991 for user permanent virtual circuits.

You can dynamically add and delete neighbors using the ADDRess parameter.

3 Enable routing on each AppleTalk bridge/router port attached to the Frame Relay network by entering:

```
SETDefault !3 -AppleTalk CONTrol = ROute
```

Configuring APPN

This section provides information for configuring the Advanced Peer-to-Peer Networking (APPN) network node for communication over a Frame Relay network.

You can configure APPN over Frame Relay over logical ports and over virtual ports. If you plan to send APPN traffic only over the port, use logical ports. Use virtual ports only if you plan to send APPN traffic and other protocol traffic over the same path to the same DLCIs. If you plan to use virtual ports, refer to "Configuring APPN with Virtual Ports" on page 42-9.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.
- Obtain the Frame Relay DLCI addresses of the remote networks to set up mapping information.

Procedure

To configure APPN to operate over a Frame Relay network, see Figure 42-3 and follow these steps:

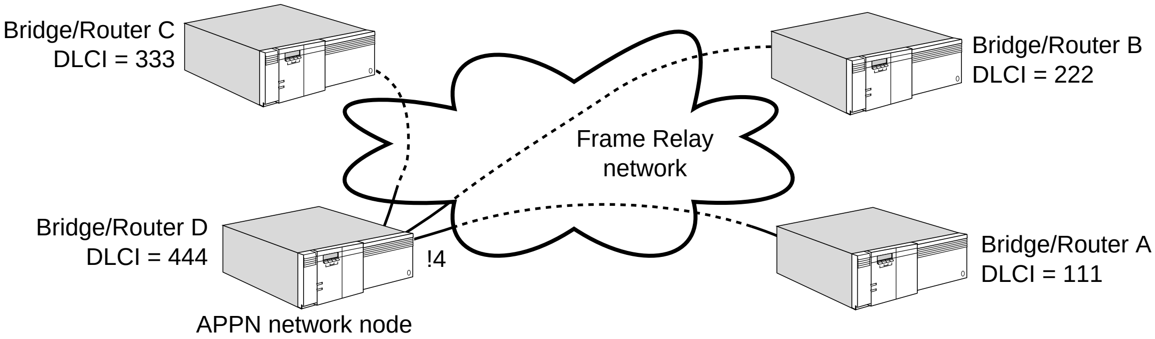


Figure 42-3 Configuring APPN over Frame Relay

1 On bridge/router D, if the ports you plan to use to send APPN over Frame Relay are active APPN ports, deactivate each one using the following syntax and specify deactivate:

```
SET !<port> -APPN PortControl = (<Activate [NoLinkStations] |
  Deactivate [Orderly | Immediate]>)
```

2 Define each APPN port that is connected to the Frame Relay network using:

```
SETDefault !<port> -APPN PortDef = <DLC type>(LLC2|FR|DLSW|UNdef)
  <max_btu_size>(99-8192) [ActLimit=<limit>(1-512)] [TGprof=<name>]
```

Make sure you specify Frame Relay as the data link control (DLC) type. For example, to configure port 2 for Frame Relay with a maximum basic transmission unit (BTU) size of 2,057, enter:

```
SETDefault !2 -APPN PortDef = FR 2057
```

If you are using logical ports, proceed to the next step. If you are using virtual ports, refer to "Configuring APPN with Virtual Ports" on page 42-9 before proceeding.

3 Configure the adjacent link stations for the Frame Relay logical port using:

```
ADD !<port> -APPN AdjLinkSta <type>(NN|EN|Learn)
  <max_btu_size>(99-8912) <[Cmac|Ncmac] dest media addr>
  [Sap=<num>] [CPName=[netid.]cpname] [Nodeid=<ID>]
  [LinkName=<name>] [TGprof=<name>] [AutoStart=(Yes|No)]
  [CPSess=(Yes|No)]
```

For the destination media address, specify the destination DLCI number. For example, if you are configuring port 4 on node D in the figure to set up a link to node A, then enter a DLCI of 111 for node A.

To configure nodes A, B, and C as adjacent link stations to node D, assuming a maximum BTU size of 2057 and a service access point (SAP) value of 08, enter:

```
ADD !4 -APPN AdjLinkSta NN 2057 111 Sap=08
ADD !4 -APPN AdjLinkSta NN 2057 222 Sap=08
ADD !4 -APPN AdjLinkSta NN 2057 333 Sap=08
```

Repeat this step for each APPN port on bridge/router A that will communicate with DLCIs on the Frame Relay network.

Because of memory storage utilization issues, do not set the maximum BTU size higher than 2057.

4 If you want to change the default link characteristics, configure any desired link characteristics using the following syntax:

```
SETDefault -APPN AdjLinkStaCHar = <LinkStation name>
  [EffectCap=<string>] [ConnectCost=<0-255>] [ByteCost=<0-255>]
  [Security=<string>] [PropDelay=<string>] [Usd1=<0-255>]
  [Usd2=<0-255>] [Usd3=<0-255>]
```

For more information on the AdjLinkStaCHar parameter, refer to Chapter 5 of *Reference for NETBuilder Family Software*.

5 To reduce the number of Logical Link Control, type 2 (LLC2) retries the system performs and the amount of time the LLC2 timer reply waits for a response to a test frame, change the values of the -LLC2 RetryCount and TimerInact parameters.

These steps are necessary because when you set the port owner as Frame Relay (using the -PORT OWNer parameter), different default values are assigned to the -LLC2 RetryCount and TimerInact parameters. It will take seven minutes to discover a link outage. If you try to deactivate the local APPN network node when this happens, the network node will not be able to deactivate until the reply is received, delaying the deactivation for up to seven minutes. If the local bridge/router is trying to contact a remote bridge/router that is not available, it will take seven minutes for the local bridge/router to discover this. To prevent this long delay, reset the values for these two parameters by entering:

```
SETDefault -LLC2 RetryCount = 3
SETDefault -LLC2 TimerInact = 30000
```

By changing these two values, you will reduce the time required for this process to 90 seconds.

6 Activate the APPN ports using:

```
SET !<port> -APPN PortControl = (<Activate [NoLinkStations])
```

7 Repeat steps 1 through 6 on nodes B, C, and D.

To ensure connectivity between two partner network nodes, the adjacent link station configuration should be performed on both sides.

You can fully mesh a configuration similar to the one shown in Figure 42-3 without using virtual ports.

Configuring APPN with Virtual Ports

You normally do not need to use virtual ports to configure APPN to operate over Frame Relay. The purpose of virtual ports is to enable multiple ports to be active on the same physical path. Because APPN allows multiple links to be active on a path at the same time, it provides the same type of capability that virtual ports provide. However, if you want to send APPN data and other protocols over the same physical path to a Frame Relay network, you may need to use virtual ports.

CAUTION: *Configure virtual ports before configuring the APPN network node.*

Prerequisite. Configure the virtual port using the procedures described in Chapter 1.

Procedure. Follow the procedure described in the previous section. However, when you configure adjacent link stations in step 3, use virtual ports. Configure the AdjLinkSta parameter as you normally would, but specify a virtual port instead of the logical port.

For example, to add a link from bridge/router D to bridge/router C using virtual port 4 with a maximum BTU size of 2057 and a SAP/TCP value of 08, enter:

```
ADD !V4 -APPN AdjLinkSta NN 2057 333 Sap=08
```

After you configure the adjacent link stations, follow the remainder of the previous procedure.

You can configure virtual ports for adjacent link stations only if the DLC type for the PortDef parameter is set to FR. If the DLC type is not set to FR, the virtual ports will not be valid.

If you configure virtual ports, when you enter the command to display link stations, the virtual ports will display as logical ports. The !V designation will not be shown in the display.

Deleting APPN Virtual Ports

After you have configured virtual ports for APPN over Frame Relay, you must be careful when deleting them. If you delete virtual ports without first deleting the adjacent link stations associated with the virtual port, you will not be able to access the link station, and you will lose all sessions over that link station.

To delete virtual ports used for APPN, follow these steps:

1 Delete the destination adjacent link station the virtual port was using, specifying the link name:

```
DELete !<port> -APPN AdjLinkSta <LinkName>
```

For example, to delete the adjacent link station with a link name of 00001 on virtual port 4, enter:

```
DELete !V4 -APPN AdjLinkSta 00001
```

2 Deactivate the physical port and specify deactivate, making sure to also specify the logical port that was mapped to the virtual port:

```
SET !<port> -APPN PortControl = (<Activate [NoLinkStations] |
  Deactivate [Orderly | Immediate]>)
```

This command deactivates all active sessions being used by the logical port.

3 Delete the virtual port using:

```
DELete !<port> -PORT VirtualPort {<path> {<FR_DLCI>}}
```

For more information, refer to “VirtualPort” on page 43-32 in *Reference for NETBuilder Family Software*.

4 Repeat this procedure for each virtual port being deleted.

Configuring DECnet

This section provides information for configuring DECnet routing for communication over a Frame Relay network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 15.
- Set up the Frame Relay service as described in “Setting Up the Frame Relay Service” on page 42-1.
- Obtain the DECnet addresses and the Frame Relay DLCI addresses of the remote networks to set up mapping information.

Procedure

To configure DECnet routing over a Frame Relay network, see Figure 42-4 and follow these steps:

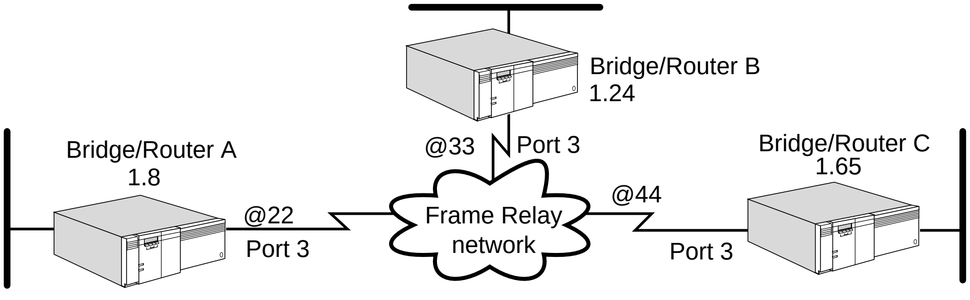


Figure 42-4 Configuring DECnet over Frame Relay

1 Specify DECnet-to-FR address mapping information on each port or virtual port that is attached to the Frame Relay network.

For example, on port 3 of bridge/router A, enter:

```
ADD !3 -DECnet Neighbor 1.24 @33
ADD !3 -DECnet Neighbor 1.65 @44
```

On bridge/routers B and C, specify the DECnet-to-Frame Relay address mapping information.

2 Enable DECnet routing on each port of each bridge/router that is attached to the Frame Relay network.

For example, to enable routing on port 3 of bridge/router A, enter:

```
SETDefault !3 -DECnet CONTrol = ROute
```

Enable routing on bridge/routers B and C.

This completes the procedure for configuring DECnet routing over a Frame Relay network.

Configuring IP

This section provides information for configuring IP routing for static and dynamic address resolution over a Frame Relay network. If your network is small and relatively stable, 3Com recommends that you configure the -ARP CONTrol parameter with the NoInArp value. This static address resolution reduces network overhead during initialization.

If your network is large and needs to be reconfigured frequently, 3Com suggests that you configure the -ARP CONTrol parameter with the InARP value. This dynamic configuration can save you some network administration work. InARP entries in the IP address table are learned when:

- IP addresses are configured.
- A new DLCI is available.

InARP entries in the IP address table are deleted when:

- The IP address table is flushed. After this occurs, InARP immediately sends out InARP requests and discovers new entries.
- An existing DLCI becomes unavailable.

To minimize the network overhead, once an IP address associated with a specific DLCI is discovered, it is treated as a static entry and is not aged out.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 6.
- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.
- Determine the IP addresses for each wide area port of your bridge/router that is attached to the Frame Relay network. If using virtual ports, each virtual port must have a unique IP address and be on a separate IP subnet.
- For static configurations only, obtain the Frame Relay DLCI addresses to create IP-to-FR mappings for static routes.
- Obtain the IP addresses of neighbors that should send or receive RIP update packets.

Procedure

To enable IP to operate over a Frame Relay network with a static or dynamic configuration, see Figure 42-5 and follow these steps:

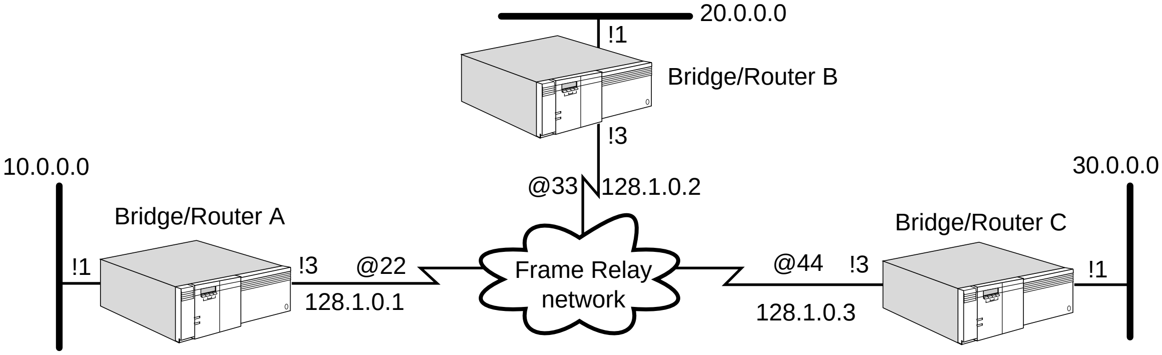


Figure 42-5 Configuring IP over Frame Relay

If you are configuring your network for static address resolution over a Frame Relay network, make sure you have set the -ARP CONTrol parameter to the NoInArp value by using the SETDefault !<port> -ARP CONTrol = NoInArp syntax.

1 Assign an IP address to each port or virtual port on each NETBuilder bridge/router that is directly attached to the Frame Relay network.

For example, to assign the address 128.1.0.1 to port 3 on bridge/router A, enter:

```
SETDefault !3 -IP NETaddr = 128.1.0.1
```

2 For static address resolution over a Frame Relay network, specify the IP-to-Frame Relay DLCI address mapping information. Specify the IP-to-Frame Relay DLCI address mapping information for each bridge/router connected to a Frame Relay network to which the system wants to communicate.

Do not perform this step if you are configuring the network for dynamic address resolution over a Frame Relay network. Proceed to step 3.

Using Figure 42-5 as an example, the following sequence of commands specify IP-to-Frame Relay DLCI mapping information for the routers directly attached to the Frame Relay network. The valid range for Frame Relay DLCIs is 16 through 991 for user permanent virtual circuits. (In the examples that follow, DLCI can be used in place of @.)

You must specify this information for DLCIs associated with ports as well as virtual ports.

For example, on bridge/router A (IP address 128.1.0.1) enter:

```
ADD -IP ADDRess 128.1.0.2 @22
ADD -IP ADDRess 128.1.0.3 @22
```

Enter similar commands on bridge/router B (IP address 128.1.0.2) and bridge/router C (IP address 128.1.0.3), specifying the IP address and DLCI mapping information.

3 For dynamic address resolution over a Frame Relay network, enable the ARP Service to automatically discover IP addresses for the DLCIs.

For example, to enable InArp on port 3, enter:

```
SETDefault !3 -ARP CONTrol = InArp
```

4 Add each bridge/router on a Frame Relay network to which the system wants to communicate with as a neighbor.

You can skip this step if the DynamicNbr option of the -RIPIP and -OSPF CONTrol parameters are enabled. If DynamicNbr is disabled, you must specify this information for ports as well as virtual ports.

Complete the following steps:

a Specify a list of neighbor addresses to which RIP will send update packets.

For example, to transmit RIP packets from bridge/router B, which is running RIP, to bridge/router C, enter:

```
ADD !3 -RIPIP AdvToNeighbor 128.1.0.3
```

b Add IP addresses of neighbors on each bridge/router port that is participating in RIP.

c Do not change the default for all neighbors.

d Specify a list of neighbor addresses to which OSPF will send update packets.

For example, to transmit OSPF packets from bridge/router B, which is running OSPF, to bridge/router C, enter:

```
ADD !3 -OSPF Neighbor 128.1.0.3
```

e Add IP addresses of neighbors on each bridge/router port that is participating in OSPF.

5 Enable the dynamic routing protocols for IP using RIPIP, OSPF, or Integrated IS-IS (IISIS) for each port and/or virtual port.

- To learn routes dynamically on port 3 using RIPIP, determine if the Frame Relay network is fully meshed or nonmeshed. If fully meshed, enter:

  ```
  SETDefault !3 -RIPIP CONTrol = (TAlk, Listen, FullMesh)
  ```

  If nonmeshed, enter:

  ```
  SETDefault !3 -RIPIP CONTrol = (TAlk, Listen, NonMesh)
  ```

  Setting the CONTrol parameter to the TAlk and Listen values enables the router to send and receive routing information with other routers using RIP.

- To enable routes dynamically on port 3 using OSPF, determine if the Frame Relay network is fully meshed or nonmeshed.

  If nonmeshed, you must run NonMesh. Enter:

  ```
  SETDefault !3 -OSPF CONTrol = (Enable, NonMesh)
  ```

  If fully meshed, you must run FullMesh. Enter:

  ```
  SETDefault !3 -OSPF CONTrol = (Enable, FullMesh)
  ```

  All of the OSPF neighboring routers must be configured with the same mode: FullMesh or NonMesh. Both of these modes apply to ports as well as virtual ports.

  Once OSPF operation is enabled, the router will exchange routing information with other routers using OSPF.

- To enable routes dynamically using IISIS, refer to Chapter 6.

6 Verify that IP routing is enabled on each bridge/router that is attached to the Frame Relay network by entering:

```
SHow -IP CONFiguration
```

If IP routing has been disabled, enable it by entering:

```
SETDefault -IP CONTrol = ROute
```

This completes the procedure for configuring IP for communication over a Frame Relay network.

Configuring IPX

This section provides information for configuring IPX routing for communication over a Frame Relay network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 13.
- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.
- Determine the IPX network numbers to be assigned to each wide area port attached to the Frame Relay network. If using virtual ports, each virtual port must have a unique network number.

Procedure

To configure IPX to operate over a Frame Relay network, see Figure 42-6 and follow these steps:

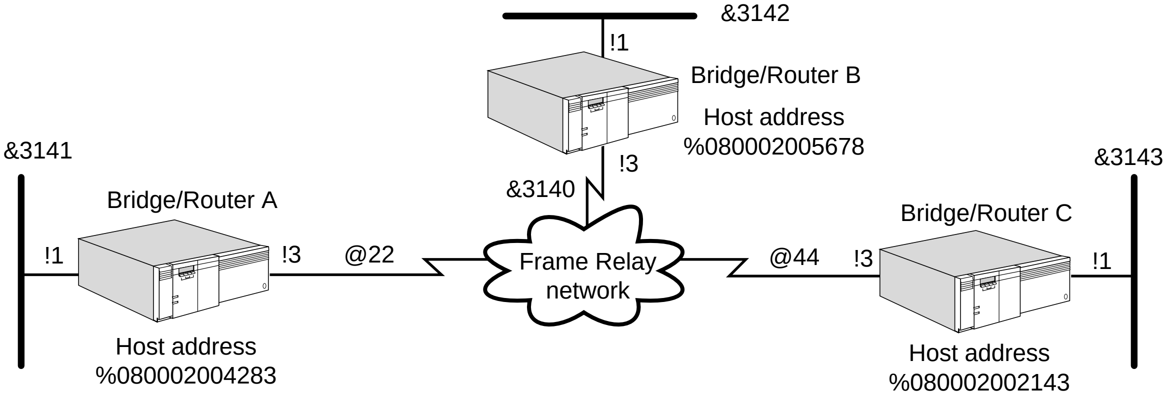


Figure 42-6 Configuring IPX over Frame Relay

1 Assign a network number to each port or virtual port on each NETBuilder bridge/router connected to the Frame Relay network.

For example, assign &3140 as the NETnumber to port 3 on bridge/routers A, B, and C by entering (on each router):

```
SETDefault !3 -IPX NETnumber = &3140
```

2 Obtain the Frame Relay DLCI addresses of the remote networks to set up mapping information.

3 You must set up mapping information between the Frame Relay addresses and host addresses for each bridge/router directly connected to the Frame Relay network.

For example, on bridge/router A, enter:

```
ADD !3 -IPX ADDRess @33 %080002005678
ADD !3 -IPX ADDRess @44 %080002002143
```

The physical MAC addresses of the neighbors are optional. If you want to use the physical MAC addresses of the neighbors, you can obtain them by using the SHow -SYS ADDRess command.

4 If you want the bridge/router to automatically send routing updates to all of the active data link connection identifiers (DLCIs), enable the DynamicNbr option in the NRIP, SAP, and NLSP CONTrol parameters. With DynamicNbr enabled, the router assumes every active DLCI points to another IPX router that is fully trusted.

If you want the bridge/router to exchange routing with only specific neighbors, disable the DynamicNbr option in the NRIP, SAP, and NLSP CONTrol parameters and configure each individual neighbor in the AdvToNeighbor parameter.

For example, on bridge/router A, to specify that bridge/router B receives route reachability information, enter:

```
ADD !3 -NRIP AdvToNeighbor &3140%080002005678
ADD !3 -SAP AdvToNeighbor &3140%080002005678
```

For NLSP, configure the Neighbor parameter for each individual neighboring router.

5 Specify the DLCI of neighbors that will be taking part in routing over Frame Relay using:

```
ADD !<port> -NLSP Neighbors @<DLCI>
```

For example on bridge/router A, enter the DLCIs of bridge/routers B and C:

```
ADD !3 -NLSP Neighbors @33
ADD !3 -NLSP Neighbors @44
```

6 Enable the use of policy parameters by entering:

```
SETDefault !3 -NRIP PolicyControl = AdvToNbr
SETDefault !3 -SAP PolicyControl = AdvToNbr
```

7 Verify that IPX routing is enabled on each bridge/router that is attached to the Frame Relay network by entering:

```
SHow -IPX CONFiguration
```

If routing has been disabled on bridge/router A, enable it by entering:

```
SETDefault !3 -IPX CONTrol = ROute
```

Enable routing on bridge/routers B and C.

In this example, bridge/routers A, B, and C are running software version 5.0 or later.

8 If you are using NRIP and SAP as your routing protocols, verify that routing is enabled on each wide area port of each bridge/router that is attached to the Frame Relay network by entering:

```
SHow -NRIP CONTrol
```

Using the SHow -SAP CONTrol command, verify that Auto, Talk and Listen, or DynamicNbr (for non-broadcast multiaccess (NBMA) networks) are set.

9 If you are using NetWare Link Services Protocol (NLSP) as the routing protocol, complete the following steps:

a If you are communicating to a non-3Com router over Frame Relay, enable the IpxWan option by entering:

```
SETDefault !3 -IPX CONTrol = IpxWan
```

b Make sure the NLSP is enabled by entering:

```
SHow -NLSP CONTrol
```

c Display the NLSP adjacencies by:

```
SHow -NLSP ADJacencies
```

This completes the procedure for configuring IPX routing over a Frame Relay network.

Configuring OSI

This section provides information for configuring OSI routing for communication over a Frame Relay network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 16.
- Set up the Frame Relay service as described in "Setting Up the Frame Relay Service" on page 42-1.
- Determine whether to use the PrefixRoute or the Neighbors parameter.
  - Use the PrefixRoute parameter if you view the remote site as another routing domain, for example, another company, with a different NSAP address. The PrefixRoute parameter allows you to specify interdomain reachability information without exchanging IS-IS packets.
  - Use the Neighbors parameter if the remote site is part of your routing domain. The neighbor information instructs the IS-IS Protocol to exchange packets and establish full connectivity.

Procedure

To configure OSI to operate over a Frame Relay network, see Figure 42-7 and follow these steps. If you are configuring the PrefixRoute parameter, begin with step 1. If you are configuring the Neighbors parameter, begin with step 2.

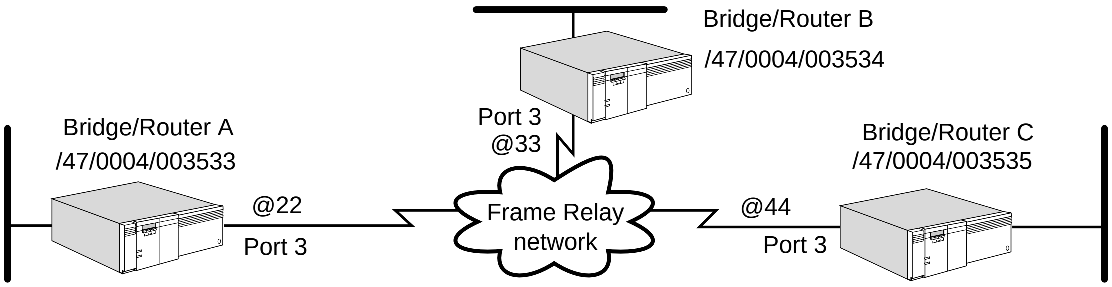


Figure 42-7 Configuring OSI over Frame Relay

1 Specify the OSI network service access point (NSAP) prefix and corresponding Frame Relay address for static interdomain routing across the Frame Relay network.

Use the -ISIS PrefixRoute parameter. The -ISIS MODE parameter must be set to L2 for the PrefixRoute parameter to take effect.

Set up static interdomain routing on bridge/router A by entering:

```
ADD !3 -ISIS PrefixRoute /47/0004/003534 @33
ADD !3 -ISIS PrefixRoute /47/0004/003535 @44
```

Specify OSI-to-FR address mapping information on bridge/routers B and C.

Proceed to step 3.

2 Specify neighbors on the Frame Relay network that support IS-IS for dynamic intradomain routing.

For example, from bridge/routers A and C enter:

```
ADD !3 -ISIS Neighbors @33
ADD !3 -ISIS Neighbors @44
```

Repeat this step for bridge/routers B and C.

3 Verify that ISIS routing is enabled on each bridge/router that is attached to the Frame Relay network by entering:

```
SHow -CLNP CONFiguration
```

If routing has been disabled, enable it on bridge/router A by entering:

```
SETDefault -CLNP CONTrol = Route
```

Enable routing on bridge/routers B and C.

This completes the procedure for configuring OSI routing over a Frame Relay network.

Configuring VINES

This section provides information for configuring VINES routing for communication over a Frame Relay network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 17.
- Set up the Frame Relay service as described in “Setting Up the Frame Relay Service” on page 42-1.
- Obtain the Frame Relay DLCI addresses of the remote networks.

Procedure

To enable the VINES Protocol to operate over a Frame Relay network, see Figure 42-8 and follow these steps:

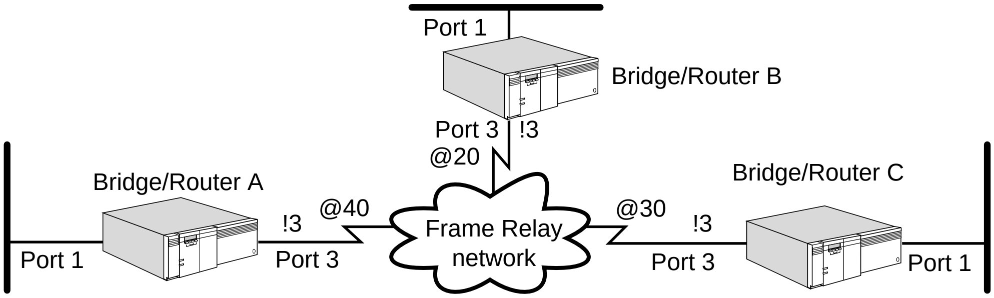


Figure 42-8 Configuring VINES over Frame Relay

1 Specify Frame Relay DLCI addresses for ports or virtual ports.

For example, to specify the DLCI address for port 3 on bridge/router A, enter:

```
ADD !3 -VIP WideAreaNbr @20
ADD !3 -VIP WideAreaNbr @30
```

On bridge/routers B and C, specify the DLCI addresses for the ports.

2 Verify that VINES routing is enabled on each bridge/router that is attached to the Frame Relay network by entering:

```
SHow -VIP CONFiguration
```

If routing has been disabled, enable it on bridge/router A by entering:

```
SETDefault -VIP CONTrol = Route
```

Enable routing on bridge/routers B and C.

Configuring XNS

This section provides information for configuring XNS routing for communication over a Frame Relay network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 18.
- Set up the Frame Relay service as described in “Setting Up the Frame Relay Service” on page 42-1.
- Determine the XNS network number for each wide area port of your bridge/router that is attached to the Frame Relay network. If using virtual ports, each virtual port must have a unique network number.
- Obtain the media access control (MAC) addresses and Frame Relay DLCI addresses of the remote networks to set up mapping information.

Procedure

To enable the XNS Protocol to operate over a Frame Relay network, see Figure 42-9 and follow these steps:

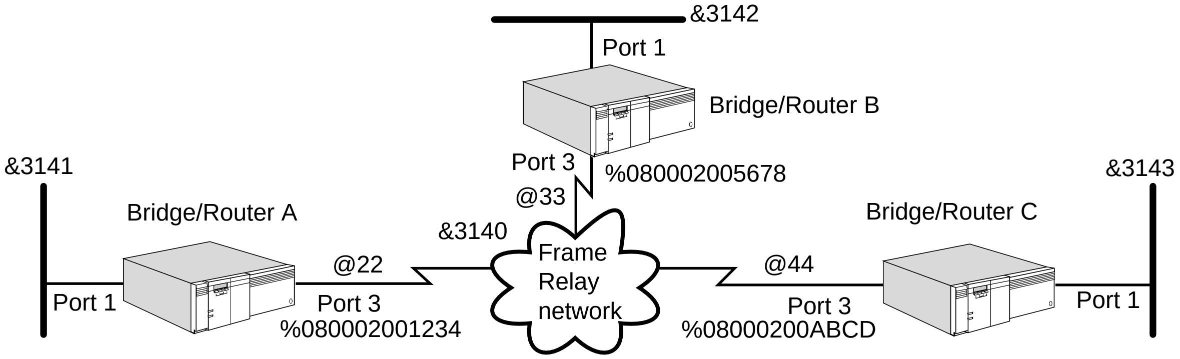


Figure 42-9 Configuring XNS over Frame Relay

1 Assign a NETnumber to each port or virtual port on each bridge/router that is connected to the Frame Relay network.

For an example, assign &3140 as the NETnumber to port 3 on bridge/routers A, B, and C by entering (on each router):

```
SETDefault !3 -IDP NETnumber = &3140
```

2 If your Frame Relay network supports the LMI Protocol and you selected the appropriate version of the protocol as described in “Setting Up the Frame Relay Service” on page 42-1, skip this step and go on to step 3. If your Frame Relay network does not support the LMI Protocol and you disabled this protocol as described in “Setting Up the Frame Relay Service” on page 42-1, set up mapping information between Frame Relay addresses and host addresses for each bridge/router directly connected to the Frame Relay network.

For example, on bridge/router A, enter:

```
ADD !3 -RIPXNS ADDRess %080002005678 @33
```

Set up mapping information on bridge/routers B and C.

3 Verify that IDP routing is enabled on each bridge/router that is attached to the Frame Relay network by entering:

```
SHow -IDP CONFiguration
```

If IDP routing has been disabled, enable it by entering:

```
SETDefault -IDP CONTrol = Route
```

Enable routing on bridge/routers B and C.

Configuring Disaster Recovery

This section discusses how to configure disaster recovery in a Frame Relay environment. The information in this section applies only to platforms that support the configuration of virtual ports.

Disaster recovery is a mechanism that allows you to maintain connectivity between your central and remote sites in the event of failure of a physical line or the Frame Relay network. This feature provides a way to recover from the loss of a primary permanent virtual circuit (PVC) in a Frame Relay network by triggering a backup PVC. If the primary PVC becomes unavailable, as determined by the LMI Protocol, the traffic destined for the primary PVC is forwarded over the backup PVC, maintaining connectivity between nodes. Upon recovery of the primary PVC, the backup PVC is deactivated, and traffic is again forwarded over the primary PVC. The backup PVC can be configured on a separate link to provide redundancy. The backup link can be either a leased or dial-up link.

For conceptual information on how disaster recovery works, refer to “How Disaster Recovery Works” on page 42-27.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your wide area interfaces according to procedures in Chapter 1.
- Configure virtual ports according to procedures in Chapter 1.
- Configure the primary link according to “Setting Up the Frame Relay Service” on page 42-1.
- Acquire services from a Frame Relay service provider. For more information refer to the *WAN Cabling and Connectivity Guide*. You can find this guide on the 3Com Corporation World Wide Web site by entering:

  ```
  http://www.3Com.com/
  ```

Procedure The following procedures describe how to configure a primary line, a backup PVC, a backup PVC on a separate link, and a backup PVC on a separate dial-up link in the event of the failure of a primary line. You must configure these lines on both ends of the link. Figure 42-10 is a configuration example.

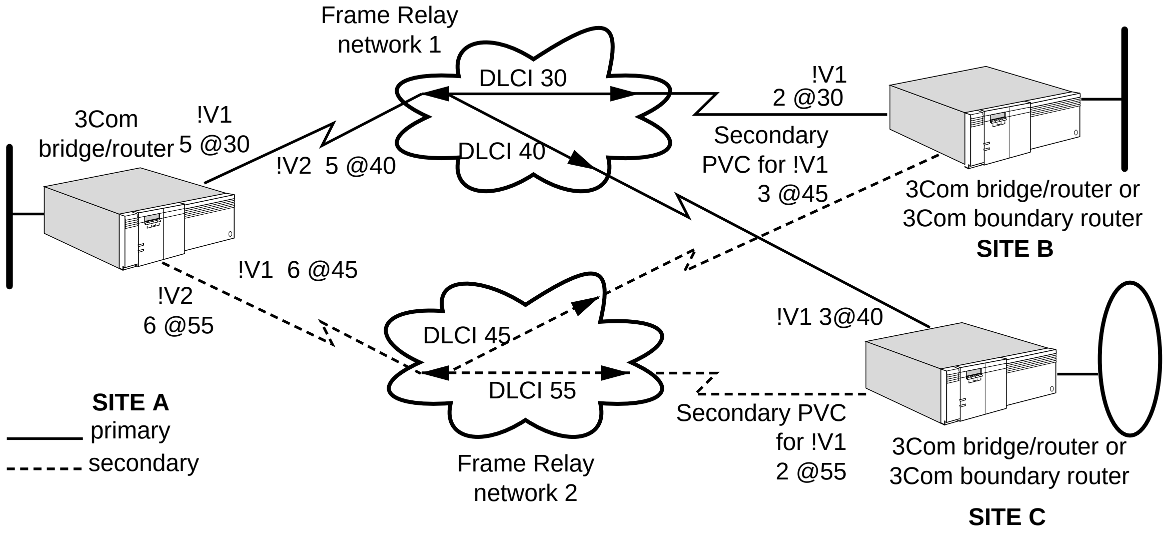


Figure 42-10 Fully Redundant Network Among All Sites

Figure 42-10 shows a configuration that is fully redundant between site A, the central site, and remote sites B and C. At site A, the primary PVCs are on link 5, and the backup PVCs are on link 6. Both of these lines are leased lines. At site B, the primary PVC is on link 2, and the backup link is on link 3. Link 3 at site B is a dial-up line. At site C, the primary PVC is on link 3, and the backup PVC is on a dial-up link 2.

Configuring a Primary PVC

To configure a primary PVC, you must first set up a virtual port with that PVC. Before setting up virtual ports, make sure the owner of the wide area port associated with the path through which the virtual ports will be defined is set to Frame Relay. Create a virtual port for each remote network that is attached to a Frame Relay cloud. You also must configure the primary link according to "Setting Up the Frame Relay Service" on page 42-1 of this chapter.

To configure a primary PVC, seeFigure 42-10 and follow these steps:

1 Configure the primary PVC between site A and site B.

a Set up a virtual port for site A by entering:

```
ADD !V1 -PORT VirtualPort 5@30
```

This command designates 5@30 as a primary PVC.

b Set up a virtual port for site B by entering:

```
ADD !V1 -PORT VirtualPort 2@30
```

This command designates 2@30 as a primary PVC.

When creating virtual ports, you must designate the same PVC on both ends of the connection as primary.

2 Configure the primary PVC between site A and site C.

a Set up a virtual port for site A by entering:

```
ADD !V2 -PORT VirtualPort 5@40
```

This command designates 5@40 as a primary PVC.

b Set up a virtual port for site B by entering:

```
ADD !V1 -PORT VirtualPort 3@40
```

This command designates 3@40 as a primary PVC.

When creating virtual ports, you must designate the same PVC on both ends of the connection as primary.

Configuring a Backup PVC

To configure a backup PVC for disaster recovery, see Figure 42-10 and follow these steps:

1 Configure the backup PVC from site A to site B by entering:

```
ADD !V1 -FR BackupPVC 6@45
```

2 Configure the backup PVC from site B to site A by entering:

```
ADD !V1 -FR BackupPVC 3@45
```

You must designate the same PVC on both ends of the connection between sites A and B as backup.

3 Configure the backup PVC from site A to site C by entering:

```
ADD !V2 -FR BackupPVC 6@55
```

4 Configure the backup PVC from site C to site A by entering:

```
ADD !V1 -FR BackupPVC 2@55
```

You must designate the same PVC on both ends of the connection between sites A and C as backup.

By default, the port is brought down when the primary PVC fails, even when the backup PVC is available.

Configuring a Backup Link

You can add a backup PVC to a previously configured virtual port to provide redundancy. If the backup PVC is on a separate path, this path must be attached to a separate port.

To configure a backup link see Figure 42-10 and follow these steps:

1 Enable the Frame Relay Service by setting the owner of the serial interface to Frame Relay by entering:

```
SETDefault !6 -PORT OWNer = FrameRelay
```

2 If your Frame Relay network supports the LMI Protocol, make sure that the appropriate LMI Protocol is enabled. For more information about enabling the LMI Protocol, refer to “Setting Up the Frame Relay Service” on page 42-1 of this chapter.

To manually enable the LMI Protocol, enter:

```
SETDefault !6 -FR CONTrol = LMI
```

How Frame Relay Works

This section provides the following basic information about Frame Relay networks:

- Definitions of fully meshed, partially meshed, and nonmeshed Frame Relay topologies and solutions to work around the connectivity problems that partially meshed and nonmeshed topologies present
- Frame Relay addresses
- LMI Protocol
- Disaster recovery over Frame Relay
- Partially and fully redundant Frame Relay networks

The wide area bridge/router supports both bridging and routing of multiple protocols over Frame Relay. ATM, X.25, and SMDS allow only one path to be assigned to a port, which means only one of these wide area protocols can run over a path or serial line. This is mostly true of Frame Relay; however, it is possible to configure dual PVCs over a single physical port on a boundary router.

In Figure 42-11, Frame Relay is being run over port 3 on bridge/router 1 and X.25 is being run over port 4.

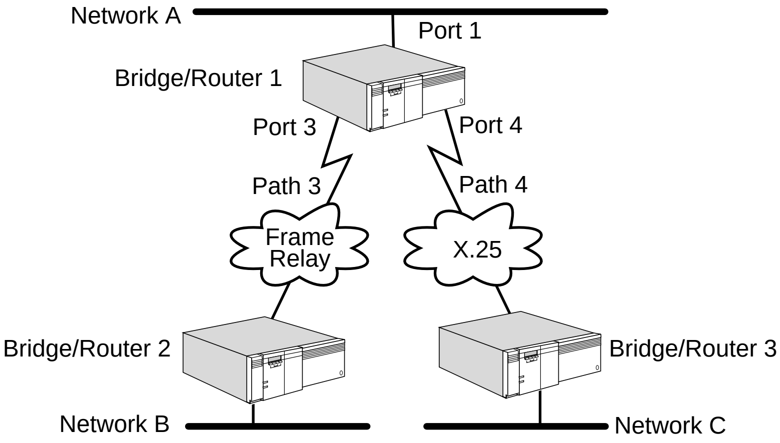


Figure 42-11 One Wide Area Protocol per Serial Line

Frame Relay allows your bridge/router to transmit and receive data over a permanent virtual circuit link with any other device on the Frame Relay network, providing a virtual connection to all other nodes on the network.

Fully Meshed, Partially Meshed, and Nonmeshed Topologies

A fully meshed Frame Relay topology (Figure 42-12) is a topology where each node on a network is directly connected to all other nodes on the network. Each node is connected to the other nodes through a PVC, and each PVC has a DLCI associated with it. This DLCI may appear as a different number to each end of the PVC.

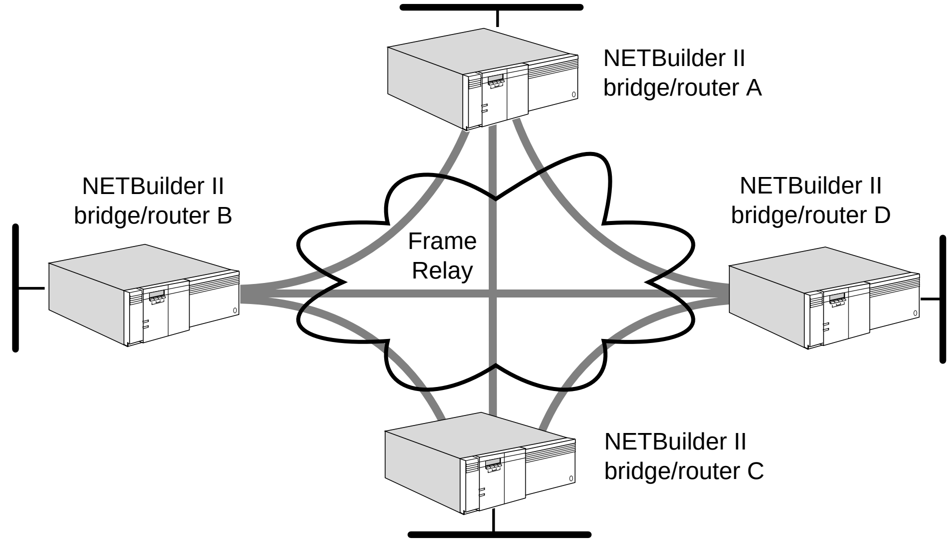


Figure 42-12 Fully Meshed Frame Relay Topology

The topology in Figure 42-12 is composed of NETBuilder II bridge/routers. Through the established PVCs, bridge/router A is connected to bridge/routers B, C, and D; bridge/router B is connected to bridge/routers A, C, and D; and so on.

A nonmeshed Frame Relay topology (Figure 42-13) is a topology where each node on a network is not necessarily connected to all other nodes on the network.

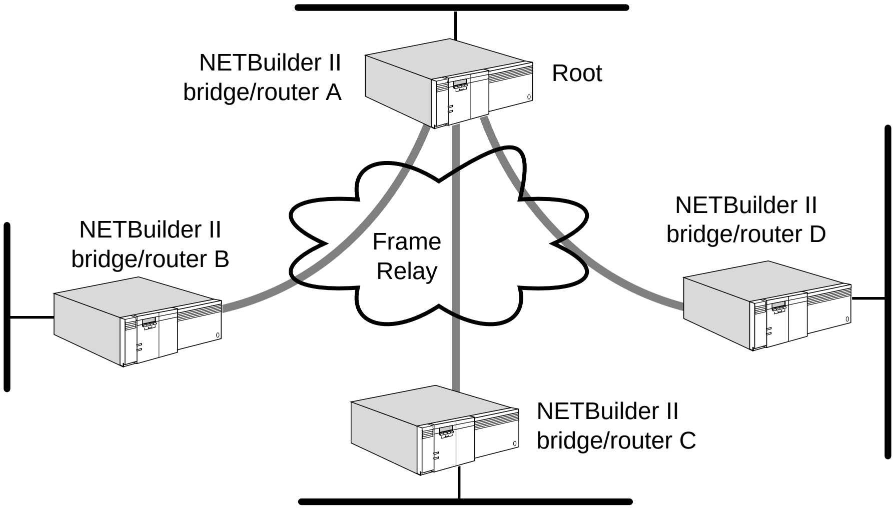


Figure 42-13 Nonmeshed Frame Relay Topology

The topology in Figure 42-13 is composed of NETBuilder II bridge/routers. Through the established PVCs, bridge/router A is connected to bridge/routers B, C, and D. bridge/routers B, C, and D are connected to bridge/router A only, but not to one another.

Transparent bridging does not correctly operate in some nonmeshed topologies. For example, in Figure 42-14, the transparent bridge properly forwards traffic received on !v1 to !v2. However, traffic received from one of its *remote connections on !v3 is not properly forwarded to the other two remote connections on !v3; therefore, do not configure transparent bridging in this type of nonmeshed topology. The flooding algorithm floods packets on a per-port basis, not on a neighbor-per-port basis.*

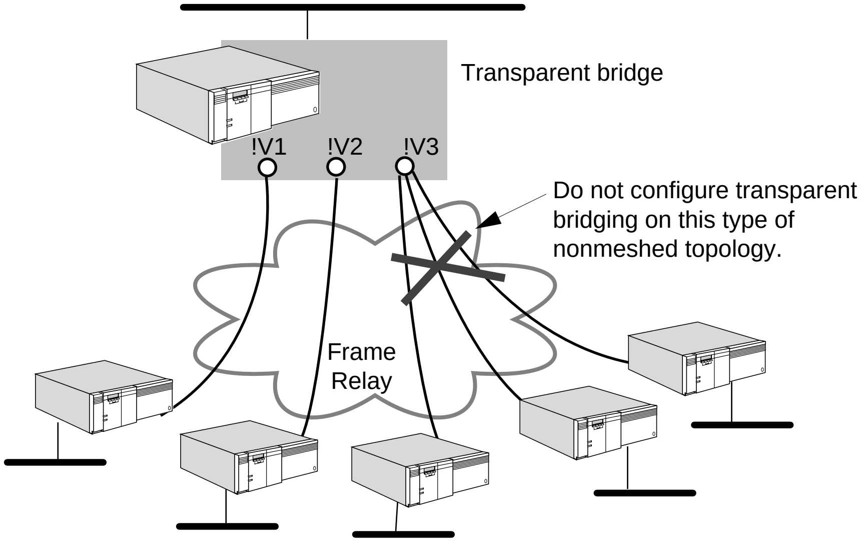

Transparent bridge
!V1
!V2
!V3
Do not configure transparent bridging on this type of nonmeshed topology.
Frame Relay

Figure 42-14 Transparent Bridging in Nonmeshed Frame Relay Topologies

A partially meshed Frame Relay topology is a topology where some nodes on a network are directly connected to nodes on the network (as in a fully meshed topology) and other nodes are not (as in a nonmeshed topology). Figure 42-15 is an example of a partially meshed Frame Relay topology.

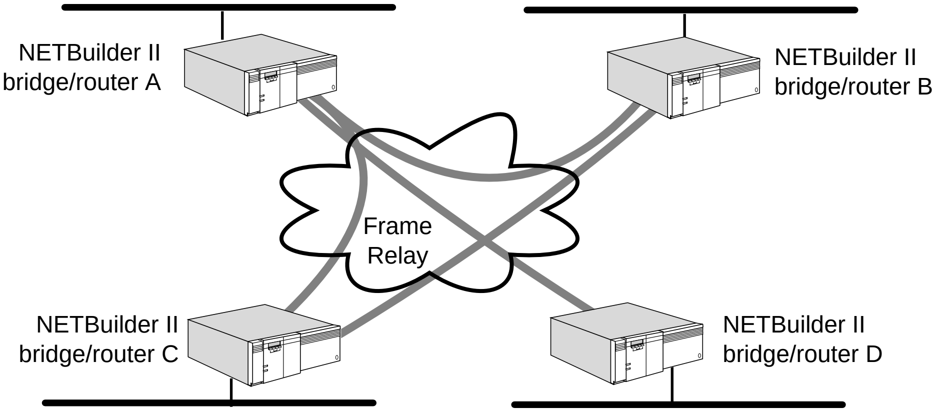

NETBuilder II bridge/router A
NETBuilder II bridge/router B
Frame Relay
NETBuilder II bridge/router C
NETBuilder II bridge/router D

Figure 42-15 Partially Meshed Frame Relay Topology

The topology in Figure 42-15 is composed of four NETBuilder II bridge/routers. Through the established PVCs, bridge/routers A, B, and C are connected to one another, but bridge/router D is connected to bridge/router A only.

Two possible solutions exist to work around the lack of connectivity between bridge/routers B, C, and D in nonmeshed and partially meshed topologies. If you are routing IP-RIP, IPX, or AppleTalk, these protocols offer the next-hop split horizon feature. In IP-RIP, set -RIPIP CONTrol to NonMesh to enable next-hop split horizon. In IPX, next-hop split horizon is enabled by manually configuring neighbors. In AppleTalk, next-hop split horizon is enabled by adding static mappings to the address mapping table.

For example, if you are routing IP-RIP and you set -RIPIP CONTrol to NonMesh, a list of neighbors containing bridge/routers B, C, and D will be generated by the system (for more information, refer to Chapter 47 in *Reference for NETBuilder Family Software*), or you can configure them as neighbors using the -RIPIP AdvToNeighbor parameter.

If routing IPX, you can configure bridge/routers B, C, and D as neighbors using the -NRIP PolicyControl and -NRIP AdvToNeighbor parameters. If routing AppleTalk, you can add the address of bridge/routers B, C, and D to an address mapping table. Bridge/router A, the root bridge/router, learns available routes from each neighbor and then updates each neighbor with available routes other than the routes of that particular neighbor. Even though bridge/routers B, C, and D are not directly connected to one another, they can still learn of routes other than their own through bridge/router A. For more information on next-hop split horizon, refer to Chapter 14, Chapter 6, and Chapter 13.

Another solution for the lack of connectivity is to create virtual ports. Virtual ports are supported by bridging and all routing protocols over a Frame Relay network. You must use virtual ports in a Boundary Routing over Frame Relay topology and when bridging or routing DECnet, VINES, or XNS over Frame Relay in a partially meshed or nonmeshed topology. Using virtual ports in all other bridging and routing scenarios over a Frame Relay network is optional. For information on the number of virtual ports supported per platform, see Table 1-1.

Virtual ports allow the creation of multiple logical ports on one path. Each PVC attaches a separate logical network. Figure 42-16 is a Boundary Routing over Frame Relay topology where virtual ports are configured. In this topology, even though the SuperStack II boundary routers are not directly connected to one another, information about each of their networks can still be propagated through the NETBuilder II bridge/router.

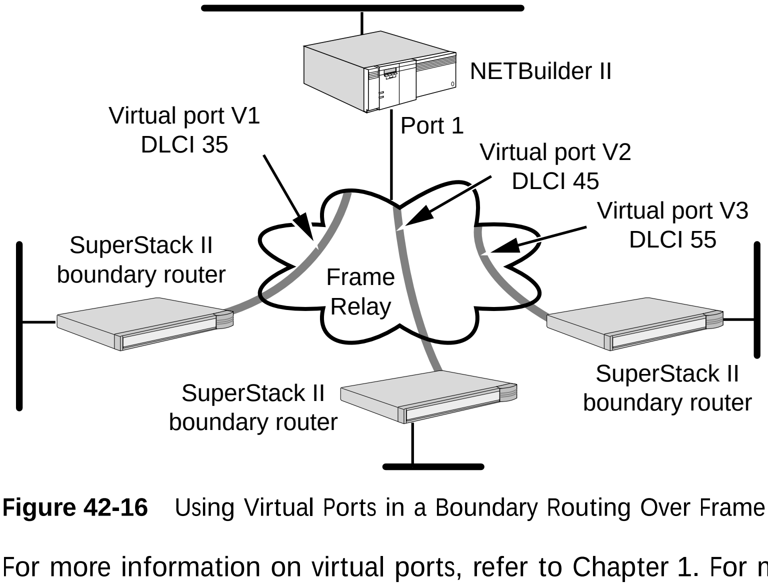


Figure 42-16 Using Virtual Ports in a Boundary Routing Over Frame Relay Topology

For more information on virtual ports, refer to Chapter 1. For more information on Boundary Routing over a Frame Relay topology, refer to Chapter 32.

Frame Relay Addresses

Before attaching your bridge/router to a Frame Relay network, obtain one or more virtual circuit identifiers, called DLCIs, from the Frame Relay service provider. A DLCI identifies a circuit between two devices from the end users' perspective. Each end of the circuit can have a different DLCI number for the link.

The DLCI number can range from 0 to 1023, but the service provider only assigns subscriber numbers ranging from 16 to 991. For ANSI and NTT LMI, 0 to 15 and 992 to 1023 are reserved. For LMI, 0 to 15 and 999 to 1023 are reserved.

In Figure 42-17, bridge/routers A, B, C, and D are assigned the DLCI numbers 36, 38, 40, and 41, respectively. The following items are examples of what occurs when packets are sent from one bridge/router to another:

- When bridge/router A sends a packet to bridge/router B, it uses DLCI 38. When the packet arrives at bridge/router B, the network changes the DLCI to 36 to indicate to bridge/router B that the packet originated at bridge/router A.
- When bridge/router A sends a packet to bridge/router C, it uses DLCI 40. When the packet arrives at bridge/router C, the network changes the DLCI to 36 to indicate to bridge/router C that the packet originated at bridge/router A.
- When bridge/router B sends a packet to bridge/router C, it also uses DLCI 40. When the packet arrives at bridge/router C, the Frame Relay network changes the DLCI to 38 to indicate to bridge/router C that the packet originated at bridge/router B.

3Com bridge/routers can operate in both local and global addressing schemes used by the Frame Relay network. In the standard (local) addressing convention, the DLCI number has only local significance; a duplicate number can be used by other bridge/routers. In the global addressing convention, identifiers used throughout the Frame Relay network are unique, and all traffic to a node has the same destination DLCI number.

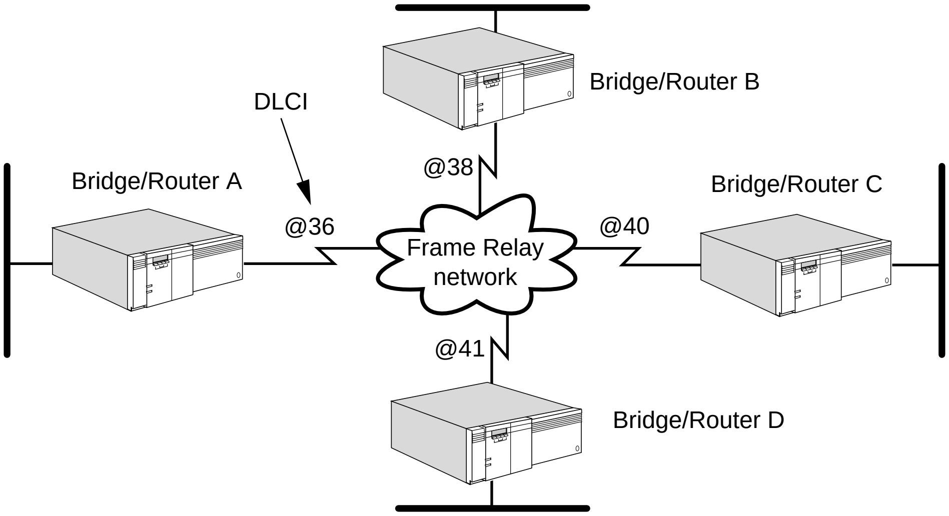


Figure 42-17 Frame Relay Addressing Example

Local Management Interface Protocol

The LMI Protocol runs between the bridge/router data terminal equipment (DTE) and the Frame Relay network switching equipment data communications equipment (DCE). The LMI Protocol provides information about all devices that are accessible on the Frame Relay network by listing all DLCIs connecting the local system with the remote ones. The LMI Protocol improves reliability between the DTE and DCE by exchanging keepalive packets that are sent every 5 to 30 seconds, depending on the configuration. If the LMI Protocol is disabled, the bridge/router assumes that all the DLCIs are active whether they are up and running or not. The LMI Protocol is enabled by default on your bridge/router.

Some switches do not run the LMI Protocol. In this situation, set the -FR CONTrol parameter to NoLMI. For complete information on this parameter, refer to Chapter 25 in *Reference for NETBuilder Family Software*.

How Disaster Recovery Works

This section describes how disaster recovery works, including the use of virtual ports, dial-up, and leased lines.

Disaster recovery is a mechanism that allows you to maintain connectivity between your central and remote sites in the event of a primary line failure or the Frame Relay network failure. This section describes how to use virtual ports and explains possible points of failure in a Frame Relay network.

If you are configuring disaster recovery over Frame Relay on the peripheral node of a Boundary Routing environment, do not configure network resiliency or the central MAC address.

Using Virtual Ports for Disaster Recovery

To configure disaster recovery, you must create virtual ports. A PVC attached to the virtual port is designated as the primary PVC. After you create a virtual port, you can add a backup PVC to that virtual port to support disaster recovery. For additional information about configuring virtual ports over Frame Relay, refer to Chapter 1.

You must configure virtual ports on both ends of the connection, and you must designate the same PVC on both ends of the connection as a primary PVC. When configuring a backup PVC, you must designate the same PVC on both ends of the connection as the backup PVC. Figure 42-10 is an example. The PVCs 5@30 at site A and 2@30 at site B are both designated as primary PVCs. 6@45 at site A and 3@45 at site B are both designated as backup PVCs. If you designate a PVC as primary on one end and backup on the other end, all packets are dropped.

You can configure the backup PVC on a separate link to provide redundancy. If the backup PVC is on a separate link, this link can be a leased line or a dial-up line. You can use a dial-up line only on one end of the connection. In most cases, a permanent connection is established between the router at the central site and the switch. At the remote site, the router establishes a connection with the switch using a dial-up link.

When using a dial-up link on a Frame Relay network, only the router can dial out by establishing a connection with the Frame Relay switch. The Frame Relay switch cannot dial out and can accept only incoming connections. When the switch accepts these incoming connections, it activates the PVC associated with that link. Since the switch cannot dial out to establish an end-to-end connection with the router, you must establish a permanent connection between the router and the Frame Relay switch on one side of your configuration. Establishing this connection enables the other side to dial out if the primary PVC fails.

The following points of failure are possible on a Frame Relay network:

- Failure of a local path
- Failure of a local switch
- Failure of a remote path
- Failure of a remote switch
- Failure in the Frame Relay network

A fully redundant network is one on which there is no single point of failure. In a partially redundant network, at least one point of possible failure exists. Data is not transmitted across a backup PVC when the primary PVC is active, which ensures correct sequencing of packets.

The triggering of the backup PVC is kept transparent to the network layer protocols by using DLCI substitution. In the case of a primary PVC failure, Frame Relay sends and receives data using the backup DLCI by substituting the primary DLCI for the backup DLCI in the packet before passing data to the network layer protocols. As long as a PVC exists that can carry the traffic on a virtual port, there is no change in port status, and the network layer protocols are unaware that the backup DLCI is being used.

In a Frame Relay disaster recovery environment, sessions may need to be restarted if a failure occurs on the primary PVC. By default LMI and Annex-D LMI provide full status information for the DLCIs every 60 seconds. If a failure occurs at one end of the primary PVC, it may take up to 60 seconds to inform the other end of the PVC. If a dial-up line is used for the backup PVC, additional time may be necessary to establish the connection. These delays can cause session timeout.

If you are configuring disaster recovery over Frame Relay on the peripheral node of a Boundary Routing environment, do not configure network resiliency. You must decide which type of redundancy best suits your needs.

Partially Redundant Networks

The following examples show the locations of redundant links and possible points of failure in partially redundant networks.

Example 1 In Figure 42-18, central site A is connected to remote site B. A redundant link (shown as the dotted line) is configured at site B, but not at site A. In this configuration, a failure of the link at site A or a failure of the Frame Relay network can bring down the connection between site A and site B.

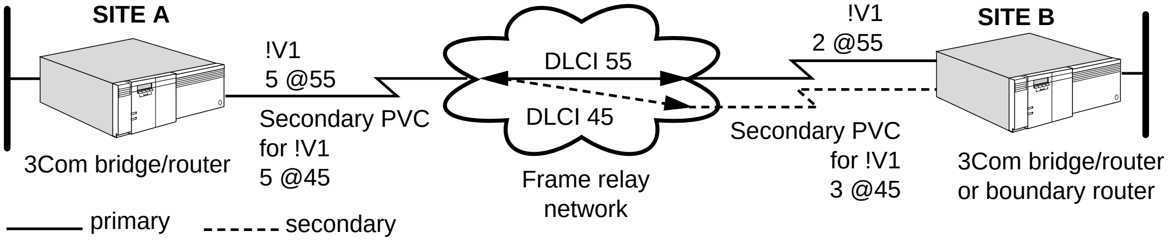


Figure 42-18 Partially Redundant Network with Redundant Link at Site B

Table 42-1 shows the covered and uncovered links for Figure 42-18 if a primary line or switch fails.

Table 42-1 Covered and Uncovered Links

Links Covered for Failure	Links Not Covered for Failure
Link failure at site B	Link failure at site A
	Frame Relay network failure

Example 2 Figure 42-19 shows a partially redundant configuration in which the redundant link is located at site A. At site B, a link failure or a Frame Relay network failure can bring down the connection between the two sites.

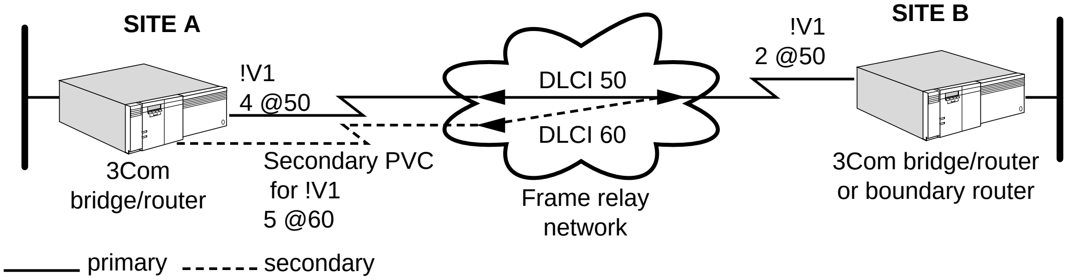


Figure 42-19 Partially Redundant Network with Redundant Link at Site A

Table 42-2 shows the covered and uncovered links for Figure 42-19 if a primary line or switch fails.

Table 42-2 Covered and Uncovered Links

Links Covered for Failure	Links Not Covered for Failure
Link failure at site A	Link failure at site B
	Frame Relay network failure

Example 3 In Figure 42-20 the network configuration has redundant links at both site A and site B. The point of failure in this configuration is the Frame Relay network.

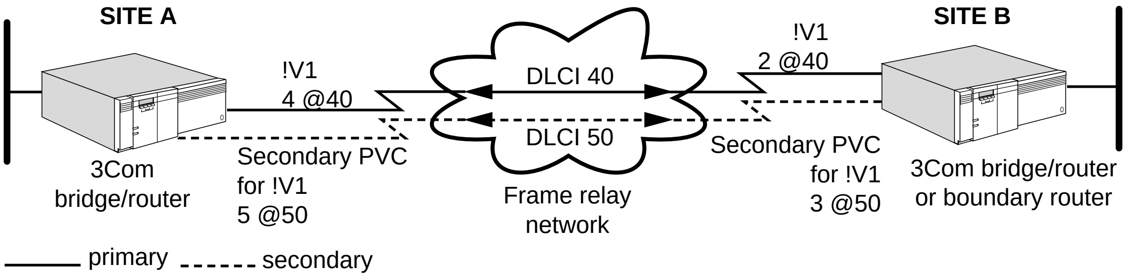


Figure 42-20 Partially Redundant Network with Redundant Links at Site A and Site B

Table 42-3 shows the covered and uncovered links in this network configuration if a primary line or switch fails.

Table 42-3 Covered and Uncovered Links

Links Covered for Failure	Links Not Covered for Failure
Link failure at either end	Frame Relay network failure

Fully Redundant Networks

The following example shows the location of a redundant link between two sites.

Example 4 Figure 42-21 shows a Frame Relay configuration that is fully redundant between site A and site B. There is no single point of failure between sites A and B, because redundancy is provided for all possible points of failure between these nodes. Sites A and B have redundant links, and both sites A and B are connected to two different Frame Relay networks. DLCI 40 is the primary PVC going between site A and site B. DLCI 45 is the backup PVC between site A and site B. Each of these PVCs belongs to a different Frame Relay network accessible through different NETBuilder bridge/router interfaces. The primary and backup PVCs are on different interfaces. The network layer protocols believe they are talking to DLCI 40, even when the backup DLCI 45 has taken over. In this configuration, there is no redundancy between site A and site C.

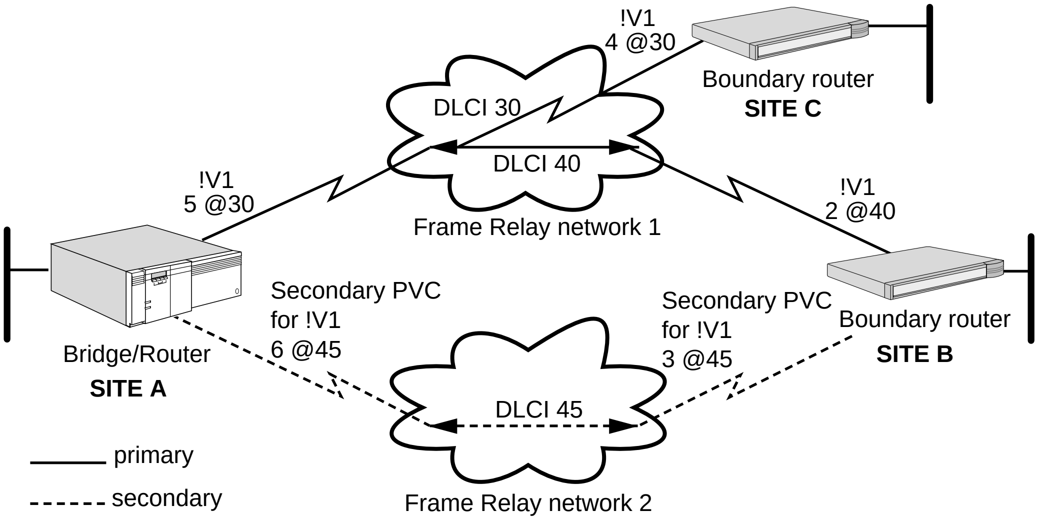


Figure 42-21 Fully Redundant Network Between Site A and Site B

Frame Relay Auto Startup

Auto startup does not determine the type of data network you subscribe to. After auto startup, you can set this value using the PDNtype parameter in the FR Service. This is a tuning feature.

For example, the following command configures port 2 to FR Service with the Sprint public data network:

```
SETDefault !2 -FR PDNtype = SPRint
```

This information applies to all platforms.

CONFIGURING WIDE AREA NETWORKING USING THE ATM DXI

This chapter describes how to configure your bridge/router to establish serial line connectivity through the Asynchronous Transfer Mode data exchange interface (ATM DXI).

For conceptual information, refer to "How ATM DXI Works" on page 43-4.

The wide area bridge/router supports both bridging and routing of multiple protocols over ATM DXI. The ATM DXI software allows your bridge/router to transmit and receive data over a permanent virtual circuit (PVC) link with any other device on the ATM network, without requiring the installation of an ATM hardware module.

By using ATM DXI software in software version 8.0 or later, your bridge/router can access the ATM switch and network through an external ATM digital service unit (DSU). The ATM DSU segments and reassembles cells, provides the ATM adaptation layer (AAL3/4 or AAL5), and provides the user-to-network interface (UNI) needed to connect to the ATM switch, (see Figure 43-1).

Your bridge/router acts as data terminal equipment (DTE), and the ATM DSU acts as data communications equipment (DCE). Bridge/routers from other vendors may attach to the ATM switch either through a DSU such as the NETBuilder II bridge/router, or directly through a UNI interface.

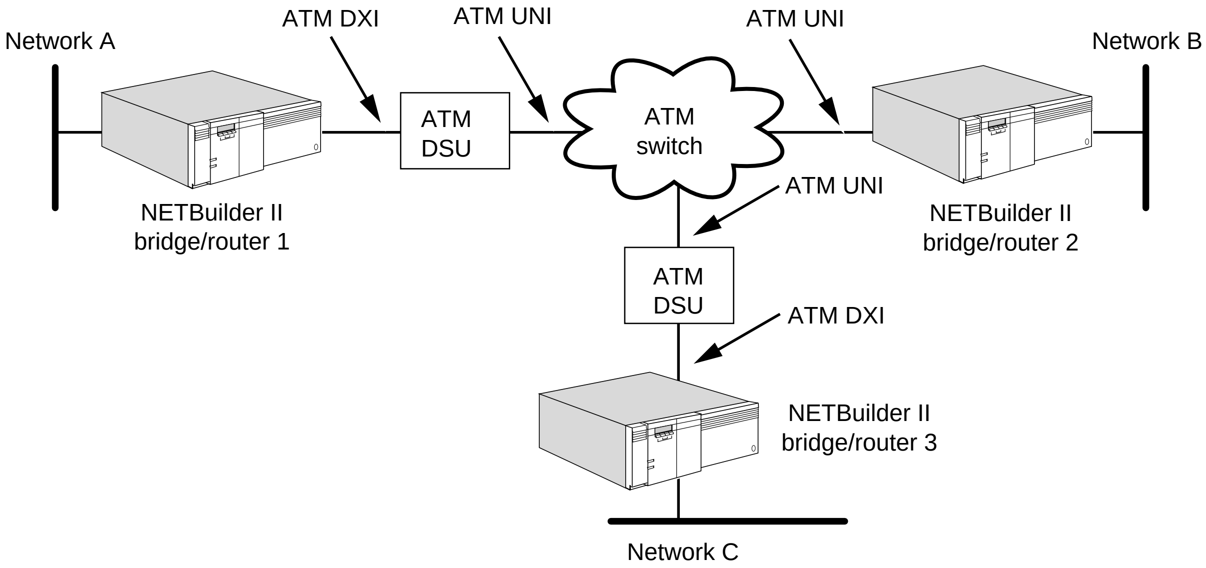


Figure 43-1 Typical ATM Connectivity

Configuring ATM DXI

Networking over ATM using the ATM DXI mode 1A is similar to networking over Frame Relay. You configure ATM DXI on the bridge/router as part of the 3Com FR Service, and all higher-level protocols use the Frame Relay configurations. To configure bridging and routing over ATM DXI, follow the procedures in Chapter 42, as if you were configuring a Frame Relay network. There are differences between ATM DXI and Frame Relay in addressing, higher-layer protocol encapsulation, and LMI Protocol features. These differences, and the corresponding changes in the configuration procedures, are explained in this section. You must consider these differences when you configure an ATM network.

Your bridge/router is also Frame-based UNI (FUNI) capable. FUNI is a variation of ATM DXI and is intended as a carrier service interface. A router currently running ATM DXI can successfully operate across a FUNI with no change. The ATM DSU is replaced with a conventional channel service unit/digital service unit (CSU/DSU), and the segmentation and reassembly function is moved into the carrier network.

ATM Address Mapping

In Frame Relay, PVCs are identified by 10-bit data link connection identifiers (DLCIs), usually represented as a decimal number between 0 and 1,023. You enter these DLCIs when you configure bridging and routing protocols, as described in Chapter 42.

In ATM, PVCs are identified by an 8-bit virtual path identifier (VPI) and a 16-bit virtual circuit identifier (VCI). The PVC is usually represented in VPI.VCI format, where VPI is a decimal number between 0 and 255 and VCI is a decimal number between 0 and 65,535.

You use the FR Service when you configure an ATM network on a NETBuilder II bridge/router, and you must enter addresses in Frame Relay format. The AtmToFr and FrToAtm utilities convert between the two address formats:

- The following syntax returns the decimal DLCI address corresponding to a decimal VPI.VCI address:

  ```
  AtmToFr <vpi.vci> (0–255.0–65535)
  ```

- The following syntax returns the decimal VPI.VCI address corresponding to a decimal DLCI address:

  ```
  FrToAtm <dlci> (0–1023)
  ```

Many different VPI.VCI addresses can map to a single DLCI. To avoid addressing errors, do not use multiple VPI.VCI addresses that map to the same DLCI.

Some vendors' DSUs require an ATM address that consists of a 0-bit VPI and a 10-bit VCI. In this case, the 10-bit VCI maps directly to a DLCI. You do not need the address conversion utilities with these addresses.

If your DSU vendor converts between VPI.VCI and DLCI addresses by bit mapping, use the address conversion utilities wherever the Frame Relay configuration procedures require a DLCI address. Otherwise, use the VCI portion of the VPI.VCI address directly as the DLCI address.

Encapsulation Type and AAL Support

In the procedure for “Setting Up the Frame Relay Service” on page 42-1, add the following step to set the encapsulation type and provide ATM Application Layer (AAL) support:

Set the ATM mode for the physical port, or selectively on each virtual port, using:

```
SETDefault !<port> -FR AtmMode = {Enable | Disable, AAL34 | AAL5}
```

If the router at the other end of the virtual circuit supports LLC/SNAP encapsulation, enable ATM mode. This sets the encapsulation type to LLC/SNAP, the normal ATM mode. If the router does not support encapsulation, disable this mode. A NETBuilder II bridge/router running software prior to 8.0 or another vendor’s router may not support encapsulation. This sets the encapsulation type to NLPID, the normal Frame Relay mode. The default is disabled. bridge/routers at both ends of a virtual circuit must use the same encapsulation type for successful operation.

Use the AAL34 parameter when connecting to an ATM DSU that supports only ATM Adaptation Layer AAL3/4. Use AAL5 when connecting to a DSU that supports AAL5. The default is AAL5. Bridge/routers at both ends of a virtual circuit must use the same adaptation layer.

LMI Protocol

ATM DXI supports an LMI Protocol that is very different from the LMI Protocol used with Frame Relay. NETBuilder II bridge/routers do not support the ATM DXI LMI Protocol. This difference causes the following changes in the configuration procedure.

Setting Up the ATM Service

In the procedure for “Setting Up the Frame Relay Service” on page 42-1, you must disable the Frame Relay LMI Protocol in step 2, using:

```
SETDefault !<port> -FR CONTrol = NoLMI
```

Step 3 of this procedure then becomes unnecessary.

Configuring Transparent Bridging

Because ATM does not support Frame Relay LMI, you must configure transparent bridging by manually adding DLCI neighbors to the static DLCI neighbor table. This procedure is explained in step 1 under “Configuring Transparent Bridging” on page 42-3.

Remember to convert the neighbors’ VPI.VCI addressees to DLCI format, if necessary, using the AtmToFr utility.

Configuring IPX over an ATM Network

Because ATM does not support LMI, you must manually enter mapping information between the ATM addresses and host addresses for each bridge/router directly connected to the ATM network. This procedure is explained in step 2 of “Configuring IPX” on page 42-14.

Configuring XNS over an ATM Network

Because ATM does not support LMI, you must manually enter mapping information between the ATM addresses and host addresses for each bridge/router directly connected to the ATM network. This procedure is explained in step 2 of “Configuring XNS” on page 42-18.

How ATM DXI Works

This section explains the differences between ATM and Frame Relay in address mapping and encapsulation type.

Address Mapping

The PVC addresses that the user obtains from the ATM switch usually are in VPI.VCI format. These addresses must be converted into DLCI format in order to configure higher-level protocols according to the procedures in Chapter 42.

NETBuilder software provides the AtmToFr and FrToAtm utilities to convert between the two address formats. For further information about VPI.VCI and DLCI formats and the conversion utilities, refer to "ATM Address Mapping" on page 43-2.

Encapsulation Type

In Frame Relay, higher-layer protocols are encapsulated using the one-byte Network Layer Protocol Identifier (NLPID) specified by RFC 1490. In ATM they are normally encapsulated using the logical link control/Subnetwork Access Protocol (LLC/SNAP) method defined in RFC 1483. If you need connectivity between a NETBuilder II bridge/router running ATM and another router that supports Frame Relay but not ATM (such as a NETBuilder II bridge/router running software prior to 8.0, or another vendor's router), you can set the encapsulation type to NLPID by disabling the -FR AtmMode parameter.

44 CONFIGURING WIDE AREA NETWORKING USING SMDS

This chapter describes how to configure your bridge/router to establish serial line connectivity through Switched Multimegabit Data Service (SMDS). It also describes how this wide area protocol works and it provides guidelines for operating, managing, and troubleshooting the protocol.

The wide area bridge/router supports bridging and routing over SMDS. SMDS allows your bridge/router to bridge or route over SMDS connectionless data service to other bridge/routers on the same wide area network.

SMDS, X.25, Asynchronous Transfer Mode (ATM), and Frame Relay allow only one path to be assigned to a port. Only one of these wide area protocols can run over a single path or serial line. For example, in Figure 44-1, SMDS is being run over port 3 on bridge/router 1, while X.25 is being run over port 4.

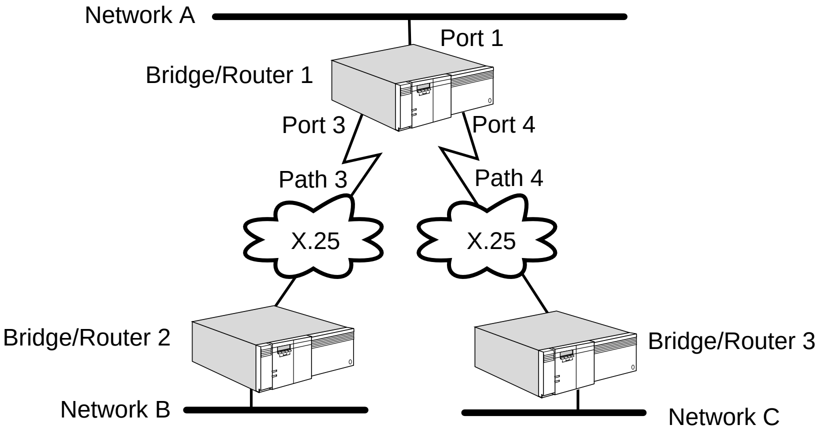


Figure 44-1 One Wide Area Protocol per Serial Line: SMDS and X.25

For conceptual information, refer to "How SMDS Works" on page 44-19.

Setting Up the SMDS Service

This section describes how to configure your bridge/router to transmit and receive data over an SMDS interface.

You must follow the steps in this section whether you are configuring for bridging or for routing. After you have completed these steps, proceed to "Setting Up Basic Bridging over SMDS" on page 44-3 for bridging configuration information or to "Setting Up Basic Routing over SMDS" on page 44-6 for routing configuration information.

For detailed descriptions of all commands and parameters, refer to *Reference for NETBuilder Family Software.*

Prerequisites Before beginning this procedure, complete the following tasks:

- Log on to the bridge/router with Network Manager privilege.
- Configure your wide area bridge/router ports and paths according to Chapter 1.
- Obtain SMDS individual and group addresses from your SMDS service provider. For more information, refer to "SMDS Addresses" on page 44-19.
- If you need to connect the SMDS interface to more than 127 other routers, or to more than one logical network segment (or more than 32 logical segments under IP routing), or if you want to use selective filtering and route policies such as those described in "SMDS Addresses" on page 44-19, create virtual ports. For information about creating virtual ports, refer to "Configuring Virtual Ports" on page 1-20.

Procedure To allow your bridge/router to transmit and receive data over an SMDS network, follow these steps:

1 Assign an SMDS individual address for each port or virtual port to be used for SMDS Service, using:

```
SETDefault !<port> -SMDS SMDSIndivAddr = $C1<address>
```

SMDS individual addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the C in the address when reporting it to you, but you must include the C when configuring the bridge/router. The digit that follows the letter C is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

For packets received on the SMDS port, in addition to checking the address syntax, the software checks the first digit (country code). If the first digit is a 1, then the software flags the packet as an error if 10 digits do not follow the country code. This error appears as a syntactic error and can be displayed with the SHow -SYS STATistics -SMDS command. For more information, refer to Appendix H. This address checking applies to both individual and group addresses.

2 Make sure that the bridge/router and the digital service units (DSUs) are configured identically.

a Enable the NewDXI option if it is supported by the DSU.

The NewDXI option corresponds to DXI 3.2 and is enabled by default. To verify this setting, enter:

```
SHow -SMDS CONFiguration
```

If the setting is incorrect and the DSU supports DXI 3.2, change it by using:

```
SETDefault !<port> -SMDS CONTrol = NewDXI
```

A virtual port inherits its CONTrol value from the parent port. You cannot configure it directly.

b Enable 32-bit cyclic redundancy check (CRC) on the path, if necessary.

Because some DSUs are configured for 32-bit CRC, the bridge/router must also be configured for the same value using:

```
SETDefault !<path> -PATH CONTrol = CRC32
```

3 Verify the clock, baud rate, and T1Mode settings for the path by entering:

```
SHow -PATH CONFiguration
```

The clock should be set to external, the baud rate should be set to 1,536 kbps, and the CONTrol parameter should be set to NoT1Mode (the default). If the settings are incorrect, change them using:

```
SETDefault !<path> -PATH CLock = External
SETDefault !<path> -PATH BAud = 1536
SETDefault !<path> -PATH CONTrol = NoT1Mode
```

If you change the clock or baud rate settings, you must re-enable the path before the new settings take effect, using SETDefault !<path> -PATH CONTrol = Enabled

4 If the DSU connected to the bridge/router is configured to use the Local Management Interface (LMI) Protocol, verify that LMI is enabled on the port or ports you are using for SMDS Service.

Confirm that the LMI Protocol is enabled using:

```
SHow [!<port>] -SMDS CONFiguration
```

The default is for LMI to be disabled. You can enable it using:

```
SETDefault !<port> -SMDS CONTrol = LMI
```

For information about the LMI Protocol, refer to “Local Management Interface Protocol” on page 44-20.

5 Enable the SMDS interface by setting the port owner to SMDS, using:

```
SETDefault !<port> -PORT OWNer = SMDS
```

This completes the procedure for configuring the SMDS Service.

Verifying the Configuration

To verify the SMDS configuration, enter:

```
SHow -SMDS CONFiguration
```

The bridge/router displays current SMDS configuration information. For information on using this parameter, refer to Chapter 53 in *Reference for NETBuilder Family Software*.

Setting Up Basic Bridging over SMDS

This section describes how to configure transparent and source route bridging over SMDS.

Configuring Transparent Bridging

This section describes how to configure your bridge/router for transparent bridging over the SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 3.
- Set up the SMDS Service as described in “How SMDS Works” on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to “SMDS Addresses” on page 44-19.

Procedure

To enable transparent bridging to operate over the SMDS network based on the example in Figure 44-2, follow these steps.

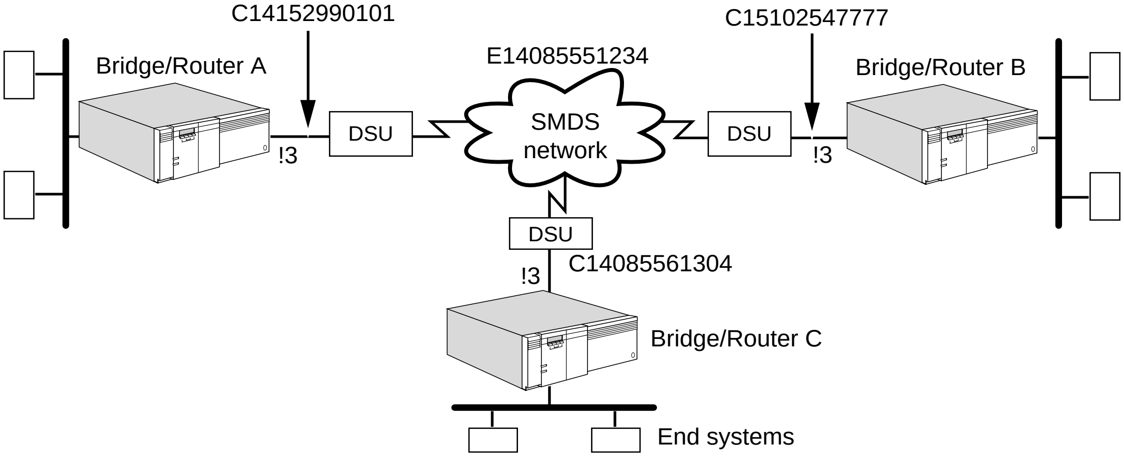


Figure 44-2 Configuring Bridging over SMDS

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network using:

```
SETDefault !<port> -BRidge SMDSGroupAddr = $E1<address>
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when configuring the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address. When you transmit a packet from bridge/router A over the SMDS network using the group address, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Verify that transparent bridging has been enabled for the appropriate wide area port by entering:

```
SHow -BRidge CONFiguration
```

By default, bridging and transparent bridging are enabled on all ports.

If bridging has been disabled, enable it for the bridge/router by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

If transparent bridging has been disabled for the wide area port (for example, on port 3, you can enable it by entering:

```
SETDefault !3 -BRidge TransparentBRidge = TransparentBRidge
```

Configuring Source Route Bridging

This section provides information for configuring source route bridging over SMDS.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 5.
- Set up the SMDS Service as described in "How SMDS Works" on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to "SMDS Addresses" on page 44-19.
- Assign a ring number to the SMDS wide area network.
- If your topology includes parallel bridges, determine unique bridge numbers.

Procedure

To configure source route bridging over SMDS, follow these steps:

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network using:

```
SETDefault !<port> -BRidge SMDSGroupAddr = $E1<address>
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

Enter the same group address on all bridge/routers attached to the SMDS network. The software uses this group address as a broadcast address. When you transmit a packet from one bridge/router over the SMDS network using the group address, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Assign each wide area port of each bridge/router attached to the SMDS network the ring number of the network it accesses.

To assign a ring number, use:

```
SETDefault !<port> -SR RingNumber = <number> (1-4095) | 0x<number>
  (1-FFF)
```

You can enter the ring number in decimal or hexadecimal format. Precede a hexadecimal number with 0x.

3 Verify that source route bridging is enabled on the wide area port using:

```
SHow !<port> -SR CONFiguration
```

If source route bridging is disabled, you need to enable it for your wide area port. For example, to enable source route bridging on port 3, enter:

```
SETDefault !3 -SR SrcRouBridge = SrcRouBridge
```

4 Disable transparent bridging on the wide area port.

For example, to disable transparent bridging on port 3, enter:

```
SETDefault !3 -BRidge TransparentBRidge = NoTransparentBRidge
```

This step does not apply to model 32x and 52x SuperStack II NETBuilder bridge/routers. Transparent bridging is not supported on these bridge/routers.

5 Verify that bridging is enabled by entering:

```
SHow -BRidge CONFiguration
```

If bridging has been disabled, enable it for the bridge/router by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

Setting Up Basic Routing over SMDS

This section describes how to configure your bridge/router to route data over an SMDS interface. The SMDS Service allows your bridge/router to perform routing over SMDS to other routers on the same wide area network.

Procedures for the following routing protocols are provided:

- AppleTalk
- DECnet
- IP
- IPX
- OSI
- VINES
- XNS

For detailed descriptions of all commands and parameters, refer to *Reference for NETBuilder Family Software.*

Configuring AppleTalk

This section provides information for configuring AppleTalk routing with group addresses or individual addresses for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your AppleTalk LAN according to the procedures in Chapter 14.
- Set up the SMDS Service as described in “How SMDS Works” on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to “SMDS Addresses” on page 44-19.
- Obtain the SMDS individual address of the remote router so that you can configure static mapping. For neighboring routers configured through static mapping, split horizon decisions are made at the next router link level. It allows for support of partially meshed and nonmeshed topologies. For neighboring routers configured through a group address, split horizon decisions are made at the port level.

Procedures

Use the following procedures and Figure 44-3 to enable the AppleTalk Protocol to operate over an SMDS network.

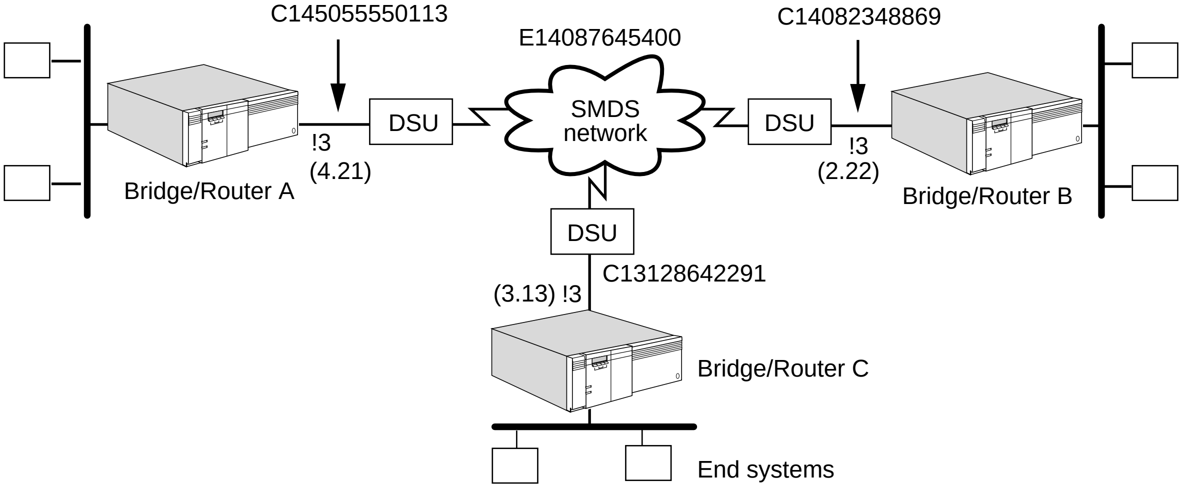


Figure 44-3 Configuring AppleTalk Routing over SMDS

Group Address Configuration

To configure AppleTalk routing over a SMDS network using a group address configuration, use Figure 44-3 as an example and follow these steps:

1 Assign a group address to each port or virtual port of each bridge/router attached to an SMDS network.

For example, to assign a group address to port 3 of bridge/router A in Figure 44-3, enter:

```
SETDefault !3 -AppleTalk SMDSGroupAddr = $E14087645400
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address. When you transmit a packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Assign a network number to the port or virtual port using:

```
SETDefault !3 -AppleTalk NetRange = <network-range>
```

3 Enable AppleTalk routing on the port attached to the SMDS network by entering:

```
SETDefault !3 -AppleTalk CONTrol = ROute
```

4 Assign a zone for the port attached to the SMDS network using:

```
ADD !3 -AppleTalk ZONe “<zone-string>” (1–32 char)
```

Individual Address Configuration

You can configure SMDS individual addresses for both non-AppleTalk and AppleTalk configurations.

Non-AppleTalk Configuration. To configure AppleTalk routing over a SMDS network configured as a non-AppleTalk network, use Figure 44-3 as an example and follow these steps:

1 Configure all the ports on bridge/routers connected to the SMDS network to be connected to a non-AppleTalk network.

On bridge/routers A, B, and C, enter:

```
SETDefault !3 -AppleTalk CONTrol = NonAppleTalk
```

2 On each bridge/router, assign the SMDS individual address of the other bridge/routers ports and virtual ports connected to the network.

For example, on bridge/router A, enter:

```
ADD -AppleTalk ADDRess !3 $C14082348869
ADD -AppleTalk ADDRess !3 $C13128642291
```

Enter similar address information on bridge/routers B and C.

You can dynamically add and delete neighbors using the ADDRess parameter while a port is enabled and AppleTalk is routing.

3 Enable routing on each AppleTalk bridge/router port attached to the SMDS network by entering:

```
SETDefault !3 -AppleTalk CONTrol = ROute
```

AppleTalk Configuration. To configure AppleTalk routing over an SMDS network as an AppleTalk configuration, use Figure 44-3 as an example and follow these steps:

The example in the following procedure assumes that the network range for the SMDS cloud shared by the configured routers is 2 to 4 and that at least one of the routers is configured to send seed information to any other nonseed routers.

1 Specify the tentative network number and the tentative node ID for the specified port that the AppleTalk router uses during dynamic node address acquisition at port enable time using:

```
SETDefault !<port> -AppleTalk StartupNET = <number>(0–65279)
SETDefault !<port> -AppleTalk StartupNODe = <number>(0–253)
```

With these parameters, the local router can always assign the same AppleTalk node address to the local port, assuming that the address is within the network range assigned to the SMDS cloud. These static configurations are saved nonvolatile storage and only need to be changed when the topology changes.

a For example, before routing is enabled on bridge/router A, enter:

```
SETDefault !3 -AppleTalk StartupNET = 4
SETDefault !3 -AppleTalk StartupNODe = 21
```

b Enter values for the StartupNET and StartupNODe parameters for bridge/routers B and C.

2 Configure static mapping of SMDS individual addresses to their AppleTalk node addresses on each bridge/router's ports and virtual ports.

For example, on bridge/router A (AppleTalk address 4.21), enter the following SMDS individual addresses of the other routers connected to the SMDS network:

```
ADD -AppleTalk ADDRess 2.22 $C14082348869
ADD -AppleTalk ADDRess 3.13 $C13128642291
```

Configure static mapping of media addresses on bridge/routers B and C.

You can dynamically add and delete neighbors using the ADDRess parameter.

3 Enable routing on each AppleTalk bridge/router port attached to the SMDS network by entering:

```
SETDefault !3 -AppleTalk CONTrol = ROute
```

To route through an SMDS network, you can either configure neighboring route through an SMDS group address, or configure using the -AppleTalk ADDRess parameter, or you can configure both.

Configuring DECnet

This section provides information for configuring DECnet routing for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your DECnet LAN according to the procedures in Chapter 15.
- Set up the SMDS Service as described in "How SMDS Works" on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to "SMDS Addresses" on page 44-19.

Procedure

To enable the DECnet Protocol to operate over an SMDS network, use Figure 44-4 as an example and follow these steps:

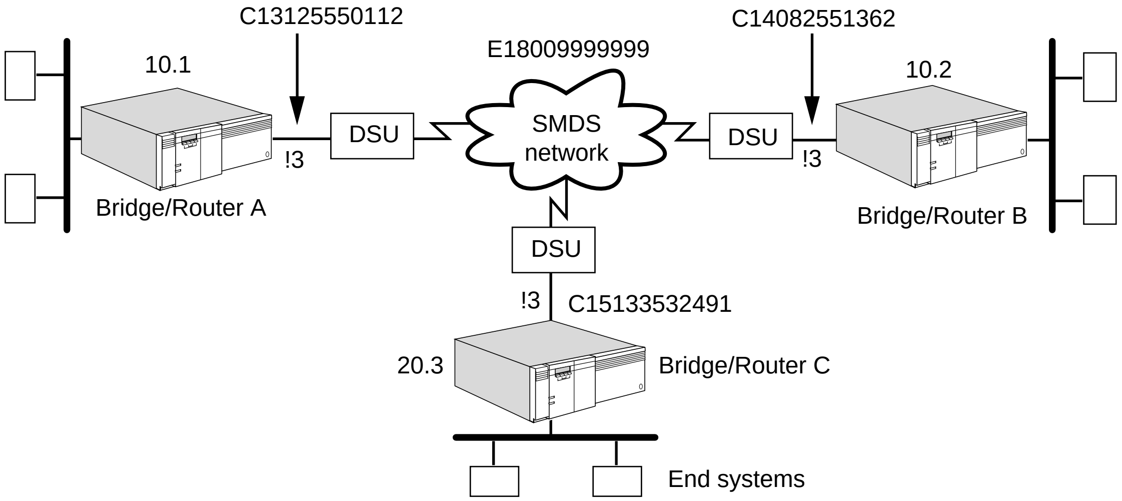


Figure 44-4 Configuring DECnet Routing over SMDS

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network.

For example, to assign a group address to port 3 of bridge/router A in Figure 44-4, enter:

```
SETDefault !3 -DECnet SMDSGroupAddr = $E18009999999
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address for routing protocol packets. When you transmit a protocol packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Enable DECnet routing on each port or virtual port of each bridge/router attached to the SMDS network.

For example, to enable routing on port 3 of bridge/router A, enter:

```
SETDefault !3 -DECnet CONTrol = ROute
```

Enable routing on bridge/routers B and C.

Configuring IP

This section provides information for configuring IP routing for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 6.
- Set up the SMDS Service as described in "How SMDS Works" on page 44-19.
- Contact the SMDS Service provider, and obtain a group address. For more information about group addresses, refer to "SMDS Addresses" on page 44-19.
- Determine the IP addresses for each wide area port of each bridge/router attached to the SMDS network.

Procedure

To enable the IP Protocol to operate over an SMDS network, use Figure 44-5 as an example and follow these steps.

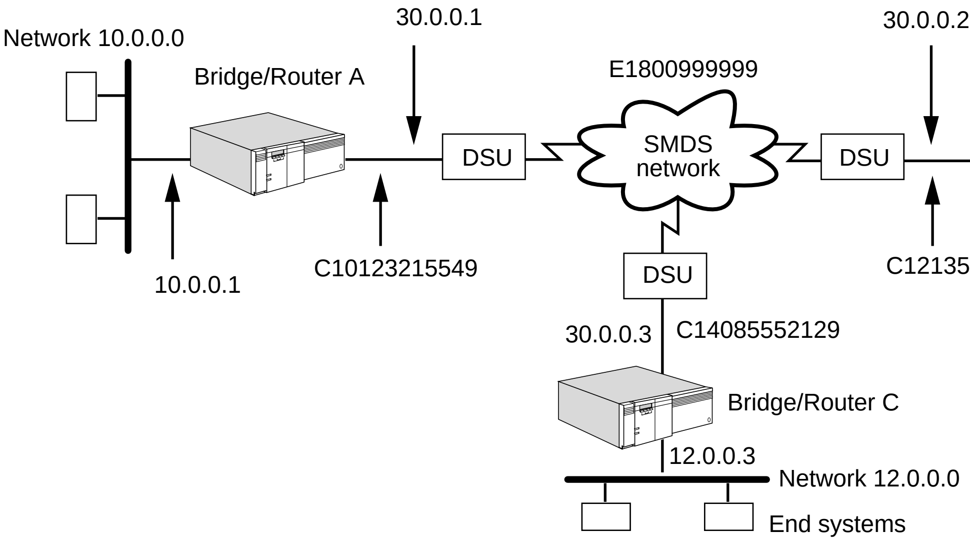


Figure 44-5 Configuring IP Routing over SMDS

1 Assign an IP address to each port or virtual port attached to the SMDS network.

For example, the following command assigns the address 30.0.0.1 with subnet mask 255.255.255.0 to port 3 on bridge/router A:

```
SETDefault !3 -IP NETaddr = 30.0.0.1 255.255.255.0
```

Assign IP addresses for bridge/router B and C on the same subnet, for example 30.0.0.2, 30.0.0.3.

2 Specify the IP-to-SMDS group address mapping information per subnet.

For example, on each of bridge/routers A, B, and C, enter:

```
ADD -IP SMDSGroupAddr 30.0.0.0 $E18009999999
```

You may configure multiple IP subnets on the same SMDS port. If you do, you must specify IP address-to-SMDS group address mapping for each subnet.

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address for routing protocol packets. When you transmit a protocol packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

3 Enable the dynamic routing protocols for IP using Routing Information Protocol for IP (RIPIP), Open Shortest Path First (OSPF), or Integrated IS-IS (IISIS).

- To learn routes dynamically on port 3 using RIPIP, enter:

  ```
  SETDefault !3 -RIPIP CONTrol = (TAlk, Listen)
  ```

  Setting the CONTrol parameter to the TAlk and Listen values enables the router to send and receive routing information with other routers using RIP.

- To enable routes dynamically on port 3 using OSPF, enter:

  ```
  SETDefault !3 -OSPF CONTrol = Enable
  ```

  Once OSPF operation is enabled, the router exchanges routing information with other routers using OSPF. OSPF does not support multiple IP subnets on a single SMDS port. Use virtual ports if you need multiple IP subnets on SMDS.

- To enable routes dynamically using Integrated IS-IS, refer to Chapter 6.

4 Optionally, specify the network-to-router IP routing information to configure static routing.

In the example shown in Figure 44-5, the following sequence of commands uses the ADD -IP ROUte <IP address> [<mask>] syntax to specify network-to-IP routing information for the bridge/routers and their respective networks directly attached to the SMDS wide area network.

On bridge/router A (IP address 30.0.0.1), enter:

```
ADD -IP ROUte 11.0.0.0 30.0.0.2
ADD -IP ROUte 12.0.0.0 30.0.0.3
```

Enter similar commands on bridge/router B (IP address 30.0.0.2) and bridge/router C (IP address 30.0.0.3), specifying the network-to-IP routing information.

5 Enable IP routing by entering:

```
SETDefault -IP CONTrol = ROute
```

This completes the procedure for configuring IP routing over SMDS.

Configuring IPX

This section provides information for configuring IPX routing for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your IPX LAN according to the procedures in Chapter 13.
- Set up the SMDS Service as described in "How SMDS Works" on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to "SMDS Addresses" on page 44-19.
- Determine the IPX network number to be assigned to the bridge/routers.

Procedure

To enable the IPX Protocol to operate over an SMDS network, use Figure 44-6 as an example and follow these steps.

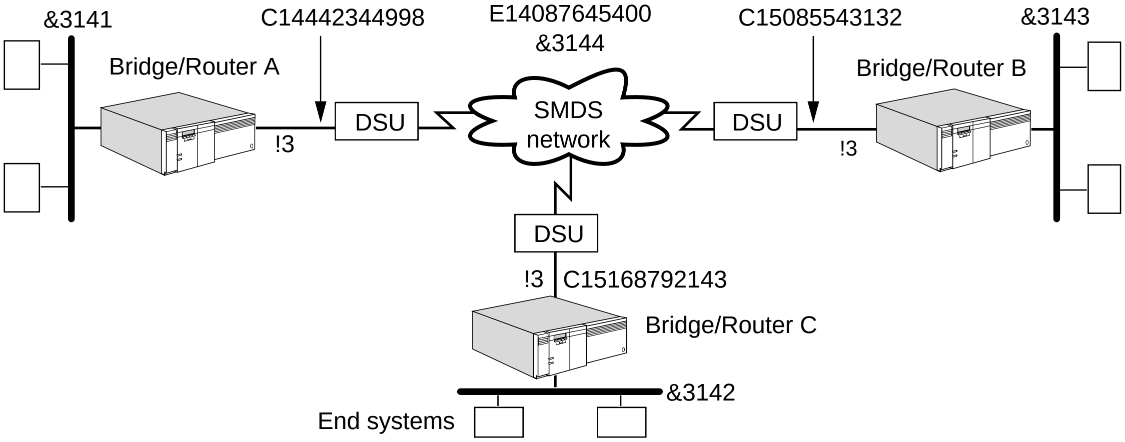


Figure 44-6 Configuring IPX Routing over SMDS

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network.

For example, to assign a group address to port 3 of bridge/router A in Figure 44-6, enter:

```
SETDefault !3 -IPX SMDSGroupAddr = $E14087645400
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address. When you transmit a packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Assign a network number to each port or virtual port attached to the SMDS network.

For example, to assign a network number to port 3 of bridge/router A, enter:

```
SETDefault !3 -IPX NETnumber = &3144
```

Assign the same network number to bridge/routers B and C.

3 Verify that IPX routing is enabled on each bridge/router attached to the SMDS network by entering:

```
SHow -IPX CONFiguration
```

If routing has been disabled on the SMDS port of bridge/router A, enable it by entering:

```
SETDefault !3 -IPX CONTrol = ROute
```

Enable routing on bridge/routers B and C.

4 Verify that dynamic learning is enabled on each wide area port of each bridge/router attached to the SMDS network.

The -NRIP CONTrol and -SAP CONTrol parameters are set to TAlk and Listen by default. To verify this setting for bridge/router A, enter:

```
SHow !3 -NRIP CONTrol
SHow !3 -SAP CONTrol
```

If the setting are not correct, you need to change the settings. For example, to enable dynamic learning on port 3 of bridge/router A, enter:

```
SETDefault !3 -NRIP CONTrol = (TAlk, Listen)
SETDefault !3 -SAP CONTrol = (TAlk, Listen)
```

Verify the settings on bridge/routers B and C.

5 Configure an internal network number on WAN links where only routers are attached using:

```
SETDefault -IPX InternalNET = &<number>(1–FFFFFFFD)
```

6 Enable the NetWare Link Services Protocol (NLSP) Protocol on the WAN links and disable NetWare Routing Information Protocol (NRIP) and Services Advertising Protocol (SAP) by entering:

```
SETDefault !3 -NLSP CONTrol = Enable
SETDefault !3 -NRIP CONTrol = (NoTalk, NoListen)
SETDefault !3 -SAP CONTrol = (NoTalk, NoListen)
```

By disabling NRIP and SAP, you conserve network bandwidth which is useful over WAN links. The NLSP Protocol uses the SMDS group address to send and receive routing packets.

7 Display the NLSP adjacencies by entering:

```
SHow -NLSP ADJacencies
```

Configuring OSI

This section provides information for configuring OSI routing for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your OSI LAN according to the procedures in Chapter 16.
- Set up the SMDS Service as described in "How SMDS Works" on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to "SMDS Addresses" on page 44-19.

Procedure

To enable the OSI Protocol to operate over an SMDS network, use Figure 44-7 as an example and follow these steps.

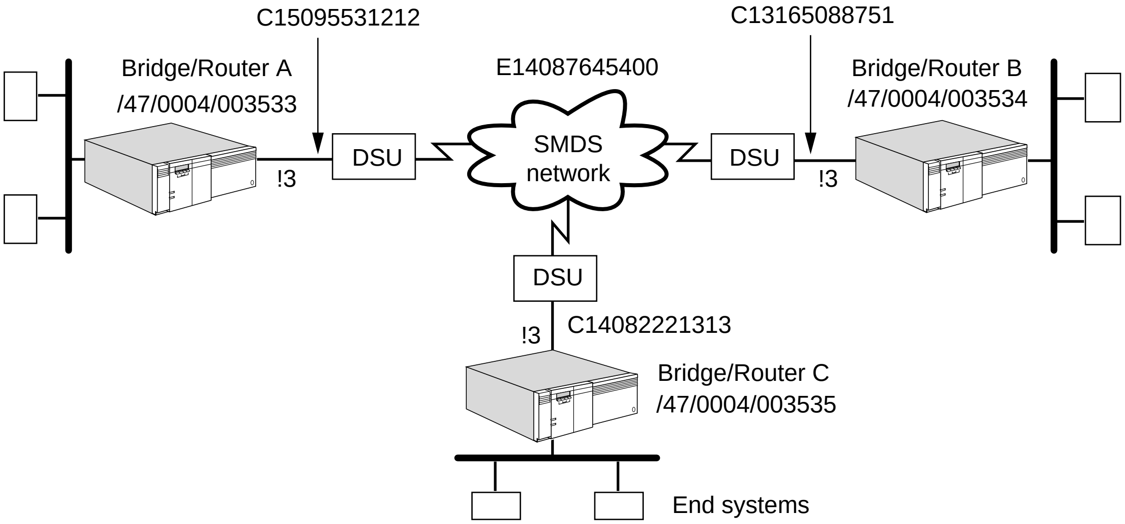


Figure 44-7 Configuring OSI Routing over SMDS

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network.

For example, to assign a group address to port 3 of bridge/router A in Figure 44-7, enter:

```
SETDefault !3 -ISIS SMDSGroupAddr = $E14087645400
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address. When you transmit a packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Verify that IS-IS routing is enabled on each bridge/router attached to the SMDS network by entering:

```
SHow -CLNP CONFiguration
```

If routing has been disabled, enable it on bridge/router A by entering:

```
SETDefault -CLNP CONTrol = Route
```

Enable routing on bridge/routers B and C.

Configuring VINES

This section provides information for configuring VINES routing for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your VINES LAN according to the procedures in Chapter 17.
- Set up the SMDS Service as described in "How SMDS Works" on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to "SMDS Addresses" on page 44-19.

Procedure

To enable the VINES Internet Protocol (VIP) to operate over an SMDS network, use Figure 44-8 as an example and follow these steps.

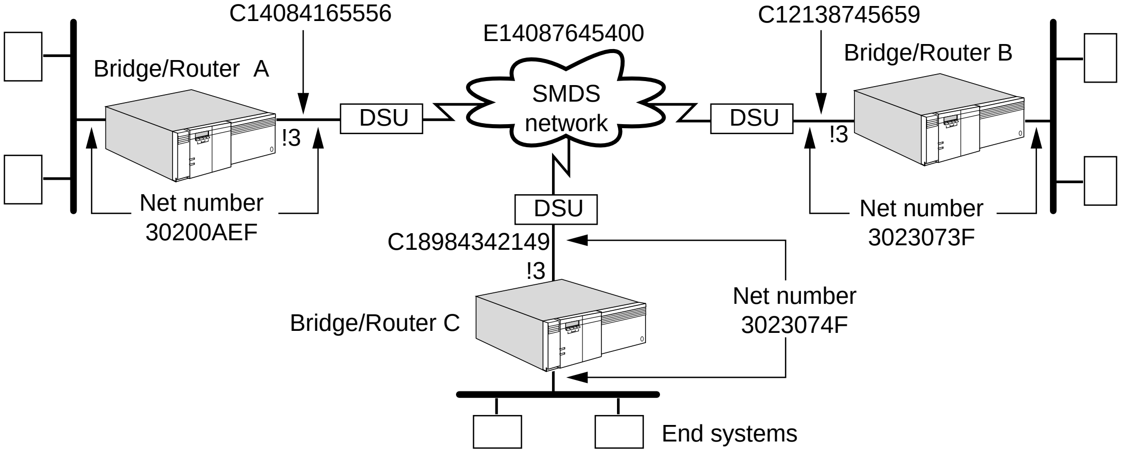


Figure 44-8 Configuring VINES Routing over SMDS

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network.

For example, to assign a group address to port 3 of bridge/router A in Figure 44-8, enter:

```
SETDefault !3 -VIP SMDSGroupAddr = $E14087645400
```

2 SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a broadcast address. When you transmit a packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

3 Verify that VINES routing is enabled on each bridge/router attached to the SMDS network by entering:

```
SHow -VIP CONFiguration
```

If routing has been disabled, enable it on bridge/router A by entering:

```
SETDefault !3 -VIP CONTrol = Route
```

4 Enable routing on bridge/routers B and C.

Configuring XNS

This section provides information for configuring XNS routing for communication over an SMDS network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your XNS LAN according to the procedures in Chapter 18.
- Set up the SMDS Service as described in “How SMDS Works” on page 44-19.
- Contact your SMDS service provider and obtain group addresses. If you are using SMDS virtual ports, obtain a separate group address for each virtual port. For more information about group addresses, refer to “SMDS Addresses” on page 44-19.
- Determine the XNS network number to be assigned to the bridge/routers.

Procedure

To enable the XNS Protocol to operate over an SMDS network, use Figure 44-9 as an example and follow these steps:

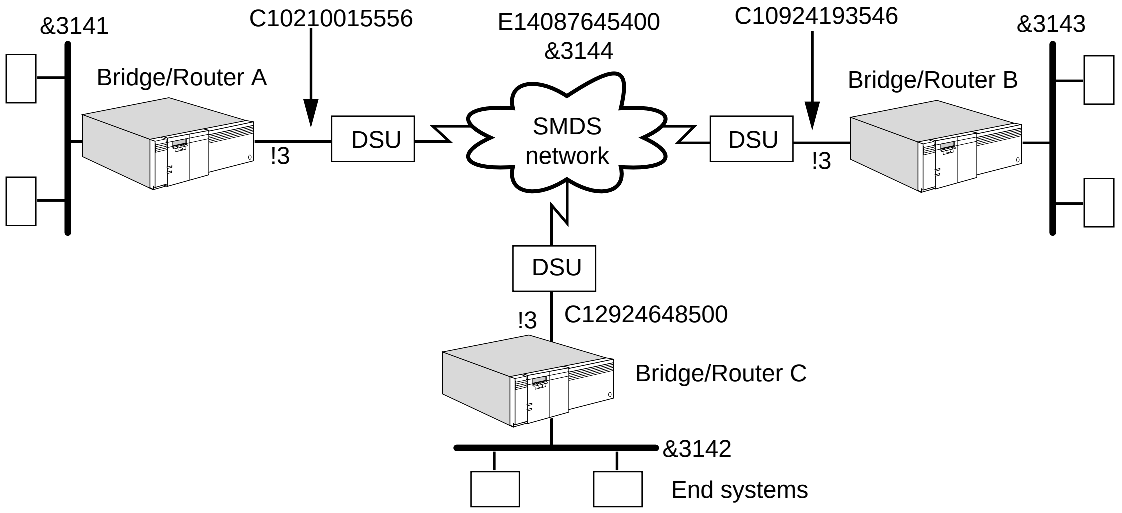


Figure 44-9 Configuring XNS Routing over SMDS

1 Assign a group address to each port or virtual port of each bridge/router attached to the SMDS network.

For example, to assign a group address to port 3 of bridge/router A in Figure 44-9, enter:

```
SETDefault !3 -IDP SMDSGroupAddr = $E14087645400
```

SMDS group addresses are provided when you subscribe to the SMDS network. Your SMDS service provider may omit the E in the address when reporting it to you, but you must include the E when you configure the bridge/router. The digit that follows the letter E is the country code. This example uses digit 1, the country code for the United States. You should use the country code for your own country reported by your SMDS service provider.

On bridge/routers B and C, enter the same group address that you assigned to bridge/router A. The software uses this group address as a multicast address. When you transmit a packet from bridge/router A over the SMDS network, all bridge/routers in the same group receive the packet.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

2 Assign a network number to each port or virtual port of each bridge/router attached to the SMDS network.

For example, to assign a network number to port 3 of bridge/router A, enter:

```
SETDefault !3 -IDP NETnumber = &3144
```

Assign the same network number to bridge/router B and bridge/router C.

3 Verify that dynamic learning is enabled on each port of each bridge/router attached to the SMDS network.

The CONTrol parameter is enabled by default. To verify this setting for bridge/router A, enter:

```
SHow !3 -RIPXNS CONTrol
```

If the CONTrol parameter is not enabled, you need to enable it. For example, to enable it on port 3, enter:

```
SETDefault !3 -RIPXNS CONTrol = Enabled
```

Verify that dynamic learning is enable on bridge/routers B and C.

4 Verify that IDP routing is enabled on each bridge/router attached to the SMDS network by entering:

```
SHow -IDP CONFiguration
```

If IDP routing has been disabled, enable it on bridge/router A by entering:

```
SETDefault -IDP CONTrol = Route
```

Enable routing on bridge/routers B and C.

How SMDS Works

SMDS is a connectionless packet switched service provided by telephone companies. Hosts or internetworking equipment connected to SMDS nodes can exchange packets across the wide area network.

To connect a router to an SMDS network, three levels of the SMDS Interface Protocol (SIP) must be supported. Your 3Com router provides the SIP-3 Protocol, which encapsulates the user data into an L3PDU. SIP-1 and SIP-2 are provided by a third-party CSU/DSU.

The sections that follow provide some basic information about SMDS Service:

- SMDS addresses
- LMI Protocol

SMDS Addresses

SMDS addresses are of two types: individual addresses, for unicast traffic, and group addresses, for multicast traffic. The addresses are distinguished by the value of the first or control digit, which has the value hexadecimal C for an individual address and hexadecimal E for a group address. Each address has 15 decimal digits following the control digit, and resembles a telephone number. If an address has fewer than 15 digits, the software automatically right-pads it with hexadecimal Fs to the full length.

C14085551212FFFF	Individual Address
E14085551234FFFF	Group Address

An individual address routes data to a unique node, a device attached to an SMDS network through a Subscriber Network Interface (SNI). The SMDS service provider assigns a block of up to 16 individual addresses to each SNI. NETBuilder software can use the extra addresses to create virtual SMDS ports through the SNI. For information about configuring virtual ports on SMDS, refer to "Configuring Virtual Ports" on page 1-20.

SMDS permits multiple nodes to be assigned the same group address (in addition to their individual addresses). Packets sent to a group address are delivered to all nodes in the group. This feature gives SMDS the appearance of a LAN.

Local Management Interface Protocol

The LMI Protocol runs between the bridge/router data terminal equipment (DTE) and the CSU/DSU data communications equipment (DCE). The LMI Protocol improves reliability between the DTE and DCE by exchanging heartbeat packets every 5 to 30 seconds, depending on the configuration.

If the LMI Protocol is not enabled, the line between the router and the CSU/DSU is assumed to be up. The LMI Protocol is disabled by default on your bridge/router.

Some DSUs do not run the LMI Protocol. In this case, set the CONTrol parameter in the SMDS Service to NoLMI (the default setting).

SMDS Service Limits

The SMDS Service sets upper limits on the number of members in a group, the number of groups an individual address can belong to, and the total number of addresses (individual and group) that any one SNI can exchange packets with.

- Each group address can represent up to 128 individual addresses.
- Each individual address can belong to up to 32 groups.
- A single SNI can exchange data among 128 total individual or group addresses.

The set of addresses that an SNI can exchange data with is configured by the service provider, following the subscriber's specifications, into a feature of the SMDS switch called the address screen. The NETBuilder bridge/router does not implement the address screen and is not aware of it.

Although an SMDS virtual port can have more than one group address, a group address cannot be shared by multiple ports on the same NETBuilder bridge/router, since NETBuilder software uses the group address to identify the virtual port for which a packet is intended.

SMDS group addresses can be used in a variety of applications where it is desirable to divide nodes on the network into several groups that are treated in different ways. The following sections give some examples of these applications.

Separating Routing Protocols

The simplest SMDS configuration allows each router to exchange data with each of the subscriber's other routers, creating a full mesh across the SMDS network. Within this configuration, group addresses can be used to separate routing protocols. For example, all routers support IP, so all routers would belong to the IP group. Only routers that support AppleTalk would belong to the AppleTalk group. By addressing them to the AppleTalk group, AppleTalk routing updates and name service queries can be sent only to AppleTalk routers.

Figure 44-10 illustrates this configuration. Routers A and B route both IP and AppleTalk. Router C routes only IP. Routers A and B are assigned one SMDS group address (creating an AppleTalk group), while all three routers are assigned another SMDS group address (creating an IP group). When the routing protocols have been properly set up, AppleTalk routing and name service broadcast packets are delivered only to routers A and B, not to router C.

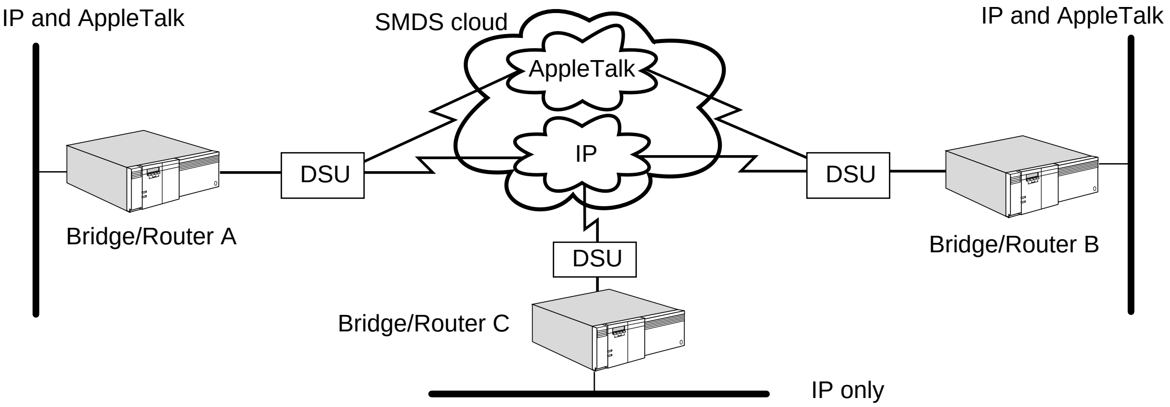


Figure 44-10 SMDS Full Mesh Configuration with Two Groups

Transparent Bridging

A more complex configuration might use virtual ports to provide additional control over the traffic. Consider a situation in which transparent bridging over SMDS is configured. Several organizations whose LANs are located close together (for example, several small companies in the same office building) all need a wide area connection to their branch offices. These organizations would like to share the cost of a bridge, but they do not want to compromise the privacy of their data or allow others to use bandwidth that they pay for. Virtual ports over SMDS, together with bridge filtering, can allow these organizations to share equipment without mixing bandwidth or broadcast traffic. The traffic of each organization is filtered to a separate virtual port, and the SMDS group address is used to identify these virtual ports.

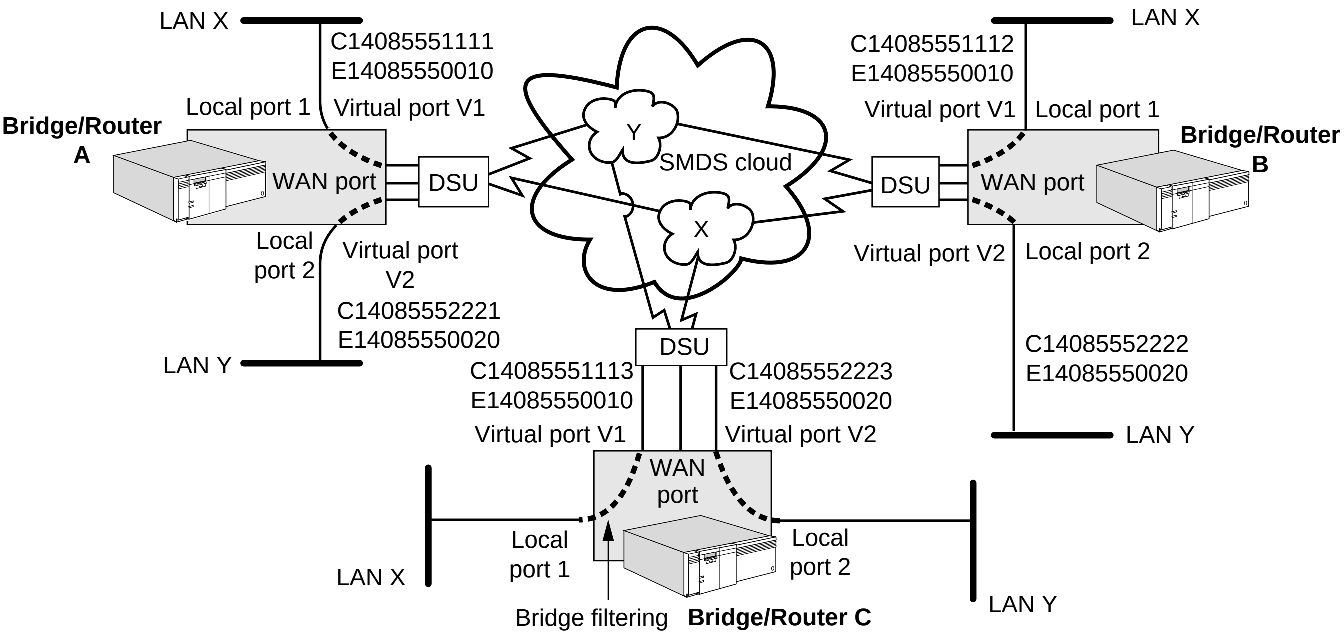


Figure 44-11 Transparent Bridging over SMDS

In Figure 44-11, LANs X and Y share bridge/routers A, B, and C, one at each location, which are all configured as bridges. Each bridge has two local ports. On each bridge, LAN X is attached to local port 1, and LAN Y is attached to local port 2.

In practice, LANs X and Y may be close to each other at only one location, not three. The techniques described in this section can be used to separate traffic at that location.

Each bridge also has a wide area port, which has been configured for the SMDS Service, as described in this chapter. NETBuilder software has also been used to create two virtual ports, V1 and V2, for this SMDS port, again on all three bridges. (One of these two ports could actually be the parent port rather than a virtual port.) At each bridge, the parent SMDS port is used to configure the SMDS CONTrol parameter, selecting the data exchange interface (DXI) that matches the DSU, and enabling or disabling LMI operation.

Two SMDS group addresses, E14085550010 and E14085550020, have been obtained from the SMDS service provider. Group address E14085550010 is assigned to virtual port V1 on all three bridges, and group address E14085550020 is assigned to virtual port V2 on all three bridges, as described in this chapter. Each virtual port on each bridge also has a unique SMDS individual address, as required by the SMDS Service.

The bridge filters on each bridge are configured so that packets are bridged only between virtual SMDS port V1 and local port 1, and between virtual SMDS port V2 and local port 2.

The bridge filters can be configured using the following commands. First, set the default action of the FIlter Service to Discard by entering:

```
SETDefault -FIlter DefaultAction = Discard
```

Define a filter mask called ANY that matches any packet by entering:

```
ADD -FIlter MASK ANY %0 | %ff = %ff
```

Add filter policies using the mask ANY by entering:

```
ADD -FIlter POLicy LANX-V1 forward ANY between !1 and !V1
ADD -FIlter POLicy LANY-V2 forward ANY between !2 and !V2
```

At each bridge, traffic from LAN X travels over local port 1 and is bridged to virtual SMDS port V1, where it is multicast to group address E14085550010. Traffic from LAN Y travels over local port 2 and is bridged to virtual SMDS port V2, where it is multicast to group address E14085550020.

The SNI address screen is configured as a full mesh, so all SMDS traffic from each bridge is sent to the other two bridges. At each bridge, traffic received for group address E14085550010 is assigned to virtual port V1 and bridged to local port 1, which is attached to LAN X. Traffic received for group address E14085550020 is assigned to virtual port V2 and bridged to local port 2, which is attached to LAN Y.

Even local bridging between ports attached to the same bridge is filtered, so data from the two organizations is always kept separate.

Source Route and Transparent Bridge Separation

You may require source route bridging over the SMDS cloud between some LAN ports (for instance, token ring and FDDI) but not others. To keep source-route-bridged traffic separate from transparently bridged traffic, you can create a virtual SMDS port to carry one kind of bridged traffic, and use the parent port or another virtual port for the other.

AppleTalk Route Filtering

Route filtering in AppleTalk is configured for each port with of the NetFilter parameter. You can selectively filter routing information learned on one port and propagated to other ports by creating virtual SMDS ports and distinct SMDS groups. Entity filtering in AppleTalk is controlled in a similar way by the EntityFilter and EntityFilterNum parameters and can be propagated selectively by the same technique.

IPX Migration from RIP/SAP to NLSP

Over IPX routing, SMDS virtual ports can be used for phased introduction of NLSP to the network, where some remote bridge/routers have not yet been upgraded to support NLSP but still support RIP/SAP. Instead of defaulting to RIP/SAP, those remote bridge/routers that understand NLSP can be collected into a new subgroup, while RIP/SAP routers remain in the original subgroup until they can be upgraded.

IP Route Policy

With IP routing, you can use SMDS virtual ports to control routing information with varying policies or protocols among the different SMDS virtual ports. For instance, one subgroup of equipment may already be using OSI IS-IS to support CLNP. The solution is to enable Integrated IS-IS selectively for these nodes under IP.

Large Hierarchical Networks

You can connect a large IP network over an SMDS cloud by combining the multiple area techniques of OSPF with SMDS virtual ports. This hierarchical approach expands the total number of bridge/routers that can be interconnected over SMDS by limiting the number that must communicate directly. Dividing the SMDS-connected bridge/routers into regions has two advantages:

- The SMDS address screen limitations are bypassed because each backbone router need communicate only with its own stub area and the other backbone routers. Different stub areas do not need to belong to the same address screen; they communicate through the backbone.
- The size of the OSPF database is reduced, saving network bandwidth for data.

The network bandwidth and router CPU time saved by OSPF summarization techniques will, in many cases, compensate for the extra hop needed by traffic traveling from a stub area to the backbone or another stub area. This configuration also saves the cost of additional SNIs that would otherwise be needed for the regional and backbone routers.

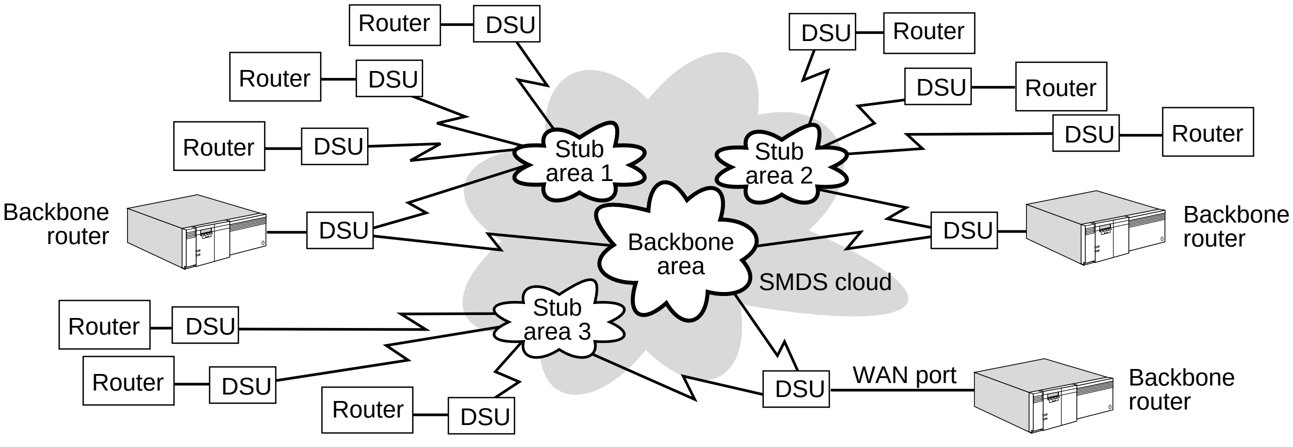


Figure 44-12 Large Hierarchical SMDS Network

45

CONFIGURING WIDE AREA NETWORKING USING X.25

This chapter describes the procedures for preparing your wide area bridge/router for X.25 wide area networking and describes how to configure your bridge/router to establish serial line connectivity through X.25. This chapter also describes how this wide area protocol works and gives guidelines for operating, managing, and troubleshooting it.

The wide area bridge/router supports bridging and routing of multiple protocols over X.25. The X25 Service allows your bridge/router to transmit and receive data over an X.25 private or public data network (PDN). (See Figure 45-1)

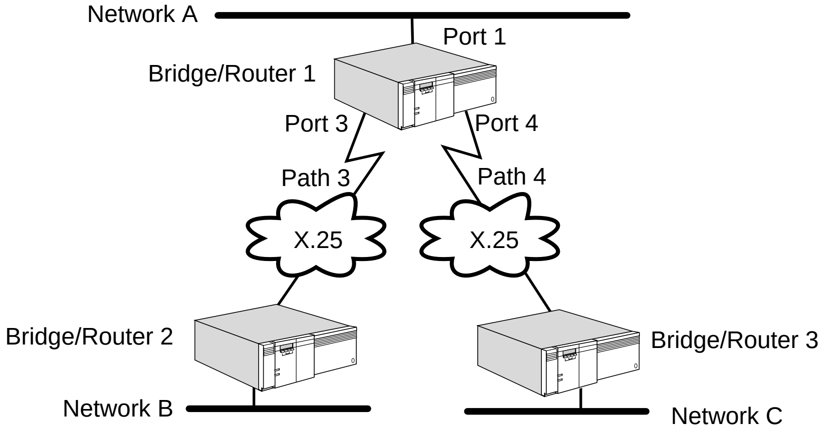


Figure 45-1 X.25 Wide Area Protocol Over Serial Lines

For conceptual information, refer to "How X.25 Works" on page 45-31.

Setting Up the X25 Service

This section describes how to configure your bridge/router to transmit and receive data over an X.25 interface. After you have completed these steps, proceed to "Setting Up Basic Routing over X.25" on page 45-9 for routing configuration information.

Figure 45-2 and the procedures and examples that follow describe how to configure the X25 Service.

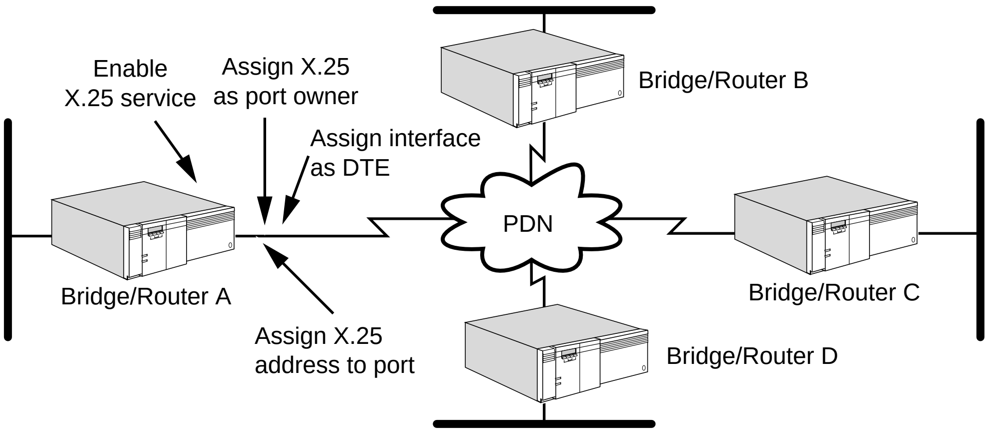


Figure 45-2 X.25 Configuration Overview

Prerequisites Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your wide area bridge/router ports and paths according to Chapter 1.
- Obtain the X.25 address from the X.25 service provider. For more information, refer to "X25Address" on page 65-4.
- Determine if you have a meshed, partially meshed, or nonmeshed topology.

  If you have any of these topologies and plan to enable the Internet Protocol/Routing Information Protocol (IP-RIP), Internetwork Packet Exchange (IPX), or AppleTalk routing, ensure that the next-hop split horizon feature is enabled.

  Refer to "How X.25 Works" on page 45-31 for information on meshed, partially meshed, and nonmeshed topologies; next-hop split horizon and how to configure it for IP-RIP, IPX, and AppleTalk; and virtual ports.
- If you plan to enable bridging or IP-OSPF (Internet Protocol/Open Shortest Path First), Xerox Network Systems (XNS), VINES, or DECnet IV routing, make sure that you have created a virtual port for each remote network that is attached to an X.25 cloud.

  For instructions on setting up virtual ports, refer to Chapter 1.

Procedure To enable your bridge/router to transmit and receive data over an X.25 PDN, follow these steps:

1 Assign X.25 as the owner of the path mapped to the port for each bridge/router port to be used for the X.25 Service using:

```
SETDefault !<port> -POrt Owner = X25
```

2 Assign an X.25 data terminal equipment (DTE) address for each bridge/router port to be used for the X25 Service using:

```
SETDefault !<port> -X25 X25Address = <address>
```

Specify this parameter for a nonvirtual port only; *do not specify it for a virtual port.*

X.25 addresses are provided by the PDN at the time of subscription.

3 Adjust other X.25 parameters to suit your installation.

The default values for the X.25 parameters adhere to the default values for the X.25 standard. However, depending on the requirements of your installation, you may need to adjust parameters, such as X25PacketSiZe and X25WindowSiZe in the PROFile Service, and X25PROFileid in the X25 Service. Additional information about these parameters is described in Chapter 45 of *Reference for NETBuilder Family Software*.

CCITT X.25 specifications recommend that the logical channels used for virtual calls be configured in the following order: one-way incoming, two-way, and one-way outgoing. In this version, TwowaySVCs are currently configured for 1 through 4095 and the others are configured to NONE. For more information on these parameters, refer to Chapter 65 in the *Reference for NETBuilder Family Software*.

The bridge/router is configured by default for communication over a private data network. If you subscribe to one of the public data networks, specify the name of the PDN using:

```
SETDefault !<port> -X25 PDNetworkType = <pdnetworkname>
```

If you select a specific PDN, you may need to configure parameters to match the required settings of the PDN. The information on the specific settings of the PDN should be provided to you at when you subscribe to the PDN.

Verifying the Configuration

To verify the X.25 configuration, enter:

```
SHow -X25 CONFiguration
```

The bridge/router displays the current X.25 configuration information.

You can use the Trace parameter for debugging and troubleshooting purposes. For information on using Trace and CONFiguration parameters, refer to Chapter 65 in *Reference for NETBuilder Family Software*.

Using X.25 Profiles

This section provides information about how X.25 profiles are used, and it briefly describes the X25 and PROFile Service parameters that help you configure source route bridging, transparent bridging, or routing over X.25.

The default characteristics for communicating over an X.25 interface is called the default DTE profile for a port. In general, the default characteristics provide optimum communications and no additional profiles are necessary. Under certain circumstances creating a profile may be advantageous. For example you may wish to create a profile under the following conditions:

- When the remote site has unique communication characteristics.
- When you have a closed user group.
- When you wish to use throughput class negotiation.

There are two types of profiles: X.25 DTE profiles and X.25 user profiles.

User Profiles

The X.25 user profile can be created and assigned to a specific port for a specific protocol. You use the X25PROFileid parameter to assign the profile to a specific port. If you are routing a network protocol such as IP or DECnet over X.25, you can use the X.25 user profiles to qualify the type of virtual circuit over which the packet is forwarded.

When a user profile is assigned, all calls to and from the protocol on that port will use the X.25 parameters in that user profile. You must use the X.25 user profile parameters to reconfigure a virtual circuit for any network protocol. X.25 user parameters are a subset of X.25 DTE parameters and are listed in Table 45-1. The remaining X.25 parameters (not defined in the user profile) are taken from the default DTE profile for establishing a call, that is, the X.25 parameters in the user profile will overwrite the parameters in DTE profile for that call.

For example, AppleTalk wants to use a VCLimit of 4, IPX wants to use a VCLimit of 6, and other protocols want to use the default VCLimit in the DTE profile. A user profile can be created with an X25VCLimit set to 4 and another user profile can be created with an X25VCLimit set to 6. These profiles are then assigned to their respective protocols.

DTE Profiles

The X.25 DTE profile contains a set of parameters that are used to establish a connection to a DTE. These parameters are listed in Table 45-1.

An X.25 DTE profile can be assigned to a specific port using the X25PROFileid parameter. The default DTE profile is assigned profile ID zero (0). All calls to and from the DTEs on a port use the DTE profile zero if the X25PROFileid parameter has not been configured for that port.

If you want to configure different X.25 parameters for different DTEs, you can create separate X.25 DTE profiles and assign each profile to a DTE using the -X25 NbrPROFile parameter. All the DTEs to which an X.25 DTE profile has not been assigned will use the default DTE profile.

For example, by default, incoming calls are allowed from all the DTEs. For security reasons, some DTEs may be allowed to establish the outgoing calls only. For those DTEs, you can create an X.25 DTE profile with NoIncomingCall and assign it to them.

X.25 Profile Parameter Usage

When an incoming call request is received, the incoming call facility parameters initially are compared with the configured DTE profiles. Before the call is accepted, a match must be found with a configured DTE profile or with the default DTE profile. Once the DTE profile is found, X.25 compares a subset of the incoming call facility parameters with the configured user profiles. The call facility parameters are X25PacketSiZe, X25ThruputClass, and X25WindowSiZe. Once the user profile is found, the user profile parameters (X25VCLimit, X25VCQueueSize, and X25VCTimer) are used to handle the congestion control for the virtual circuit. If a user profile is not found, the values from the matched DTE profile or the default DTE profile are used for congestion control.

Table 45-1 lists the X.25 DTE and X.25 user profile parameters.

Table 45-1 X.25 User and X.25 DTE Parameters

X.25 User Profile Parameters	X25 User Profile Parameter Default	X.25 DTE Profile Parameters	X.25 DTE Profile Parameters Default
X25COMPressType	DEFault	X25COMPressType	DEFault
X25PacketSiZe	128	X25PacketSiZe	128
X25ProfileName	No default	X25ProfileName	No default
X25ThruputClass	9600	X25ThruputClass	9600
X25VCLimit	2	X25VCLimit	2
X25VCQueueSize	10	X25VCQueueSize	10
X25VCTimer	5	X25VCTimer	5
X25WindowSiZe	2	X25WindowSiZe	2
X25CUDSuffix	No default	X25ClosedUsrGrp	0
		X25CONTrol	IncomingCall, OutgoingCall, NoPSN, NoWSN, NoTCN
		X25FastSelect	NoRequest, NoAccess
		X25ReverseCharge	NoRequest, Accept

Configuration Parameters

This section describes the most useful parameters in configuring the characteristics of a particular DTE. Table 45-2 lists the parameters and services that can help you configure your bridge or bridge/router for the X25 Service. For more complete information on these parameters, refer to Chapter 45 and Chapter 65 in *Reference for NETBuilder Family Software.*

Table 45-2 X.25 Configuration Parameters

Parameter	Service	Description
ProfileType	PROFile	Creates an X.25 profile that is used when X.25 virtual circuits are set up to carry bridge/router traffic.
X25Address	X25	The international data number (IDN) assigned by the network provider. Can be up to 15 decimal digits.
X25PacketSiZe	PROFile	Specifies the packet size (in bytes) for a specified virtual circuit.
X25VCLimit	PROFile	Specifies the maximum number of virtual circuits to a specific DTE for a specific protocol.
X25VCQueueSize	PROFile	Specifies the maximum number of packets that can be queued for any single virtual circuit to a specific DTE when the virtual circuit on the X.25 port is congested.
X25VCThruputClass	PROFile	Specifies the throughput rate in bits per second. This parameter is used by the PDN to guarantee the bandwidth for the virtual circuit.
X25VCTimer	PROFile	Specifies the maximum amount of time (in minutes) that can elapse when there is no activity on the X.25 virtual circuit before it is cleared.
X25WindowSiZe	PROFile	Determines the X.25 packet layer window size for the virtual circuit.

X.25 Profiles Configuration Examples

This section provides examples of how X.25 DTE and X.25 user profiles can be applied to an X.25 network.

Example 1 Using Figure 45-3 as a sample network, assume you want to change the packet size and window size for all calls on port 3. You need to create an X.25 DTE profile with new packet size and window size values and assign them to port 3.

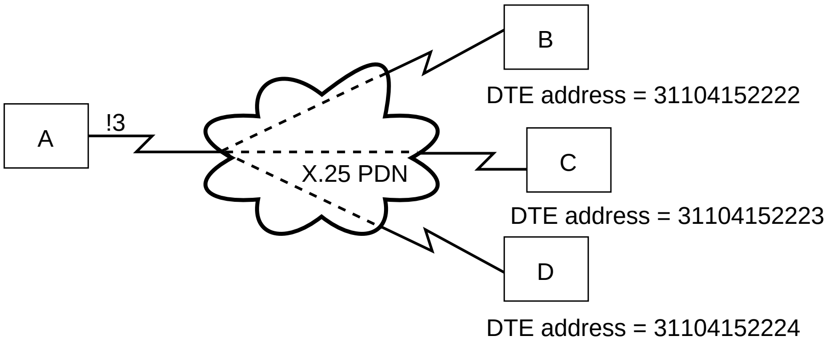

B
DTE address = 31104152222
A
!3
X.25 PDN
C
DTE address = 31104152223
D
DTE address = 31104152224

Figure 45-3 Creating X.25 DTE Profiles

To change the packet size and window size for all calls on port 3, follow these steps:

1 Create an X.25 DTE profile 4 by entering:

```
ADD !4 -PROFile ProfileType X25Dte
```

2 Increase the packet size in profile 4 from 128 (the default) to 1024 by entering:

```
SETDefault !4 -PROFile PacketSiZe = 1024
```

3 Increase the window size in profile 4 from 2 (the default) to 4 by entering:

```
SETDefault !4 -PROFile X25WindowSiZe = 4
```

4 Assign profile 4 to port 3 by entering:

```
SETDefault !3 -X25 X25PROFileid = 4
```

Example 2 In Figure 45-3, A is using port 3 to route IP and IPX to B, C, and D. You want to increase the throughput from A to B, from A and C, and from A to D. For security reasons, you want to allow B to establish only outgoing calls. To accomplish this, follow these steps:

1 Create an X.25 DTE profile 10 for A to B traffic by entering:

```
ADD !10 -PROFile ProfileType X25Dte
```

2 Create an X.25 DTE profile 20 for A to C traffic by entering:

```
ADD !20 -PROFile ProfileType X25Dte
```

3 Create an X.25 DTE profile 30 for A to D traffic by entering:

```
ADD !30 -PROFile ProfileType X25Dte
```

4 Increase the throughput rate in profile 10 from 9600 (the default) to 19200 by entering:

```
SETDefault !10 -PROFile X25ThruputClass = 19200
```

5 Allow B to establish only outgoing calls by entering:

```
SETDefault !10 -PROFile X25CONTrol = NoIncomingCall
```

6 Increase the throughput rate in profile 20 from 9600 to 38400 by entering:

```
SETDefault !20 -PROFile X25ThruputClass = 38400
```

7 Increase the throughput rate in profile 30 from 9600 to 48000 by entering:

```
SETDefault !30 -PROFile X25ThruputClass = 48000
```

8 Assign profiles 10, 20, and 30 to B, C, and D, respectively, by entering:

```
ADD -X25 NbrPROFile #31104152222 10
ADD -X25 NbrPROFile #31104152223 20
ADD -X25 NbrPROFile #31104152224 30
```

Example 3 You are routing IP, IPX, and DECnet between A and B over an X.25 PDN. You have been assigned six circuits for all traffic and want to allocate three to IP traffic, two to IPX, and one to DECnet. To allocate the traffic, refer to Figure 45-4 and follow these steps:

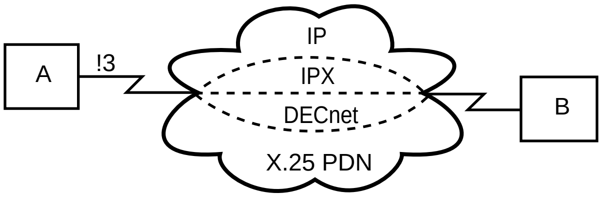


Figure 45-4 Creating X.25 User Profiles

1 Create an X.25 user profile 15 to be used for routing IP traffic between A and B by entering:

```
ADD !15 -PROFile ProfileType X25User
```

2 Create an X.25 user profile 25 to be used when routing IPX traffic between A and B by entering:

```
ADD !25 -PROFile ProfileType X25User
```

3 Create an X.25 user profile 35 to be used when routing DECnet traffic between A and B by entering:

```
ADD !35 -PROFile ProfileType X25User
```

4 Change the number of virtual circuits available for IP, IPX, and DECnet using the profiles established in steps 1–3 and by entering:

```
SETDefault !15 -PROFile X25VCLimit = 3
SETDefault !25 -PROFile X25VCLimit = 2
SETDefault !35 -PROFile X25VCLimit = 1
```

5 Assign your X.25 user profiles to port 3 by entering:

```
SETDefault !3 -IP X25PROFileid = 15
SETDefault !3 -IPX X25PROFileid = 25
SETDefault !3 -DECnet X25PROFileid = 35
```

Example 4 Data prioritizing over X.25 does not use the Data Prioritizing scheme, that is the four levels of priority, as used by other WAN Services. This is because X.25 does not use LMF queuing. X.25 uses it's own virtual circuit queue and maintains it's own queues for each virtual circuit. If an IP data packet had a priority set to High under it's global parameter, the bit would be set, but the X.25 queue would not check for this bit and would not place any priority on this packet.

Instead, X.25 uses X.25 user profiles to obtain the best bandwidth characteristics, that is the number of virtual circuits, the packet size, window size and so on. X.25 maintains it's own queue and each virtual circuit can have different depths of queues and multiple queues per protocol. X.25 also does it's own sequencing of packets and its own fragmentation. There are my X.25 parameters that are available to determine the number of virtual circuits per protocol, the queue size for each protocol, and the length of time the switched virtual circuit will stay open when there is not data. Other X.25 parameters give the ability to set the packet size, window size the throughput of each switched virtual circuit. These parameters allow for better control of the X.25 traffic.

User profiles for the IP service can be configured, so all traffic such as Telnet and FTP use the same profile. However, it is also possible to establish a user profile per IP protocol, that is one for Telnet and one for FTP. In this situation all the virtual circuits assigned for Telnet could be give better X.25 characteristics such as window size and throughput, compared with the switched virtual circuits assigned to FTP. In addition, all the IP protocol traffic could be given better X.25 characteristics than other protocol traffic. It also means that non-I/O traffic also gets a fair allocation of virtual circuits.

By default the IP protocol does not have an X.25 user profile configured. You much create an X.25 user profile if you want to assign a priority to IP packets over other traffic.

To prioritize FTP IP packets using and X.25 user profiles, follow these steps:

1 Create an X.25 user profile by entering:

```
Add !1 -PROFile ProfileType X25user
```

2 Assign the X.25 user profile the IP service by entering:

```
SETDefault !2 -IP X25profileid = 1
```

Giving the x25 profile an identity of 1, is an arbitrary number assigned by the user.

3 Adjust the number of virtual circuits for each profile ID by entering:

```
SETDefault !1 -PROfile X25VClimit = 4
```

This allows IP protocols to use 4 virtual circuits, the default is 2.

4 Improve the response for FTP traffic by adjusting the X.25 window size by entering:

```
SETDefault !1 -PROFile X25WindowSiZe = 7
```

5 Further improve response for FTP traffic by adjusting the X.25 packet size by entering:

```
SETDefault !1 -PROFile X25PacketSIZe = 1024
```

When setting the X25PaketSIZE for Telnet traffic, be aware that Telnet will only use 64K packets so changing the size to larger than 64K will not help performance.

Setting Up Basic Routing over X.25

This section describes how to configure your router to transmit and receive data over an X.25 interface. Procedures for the following routing protocols are provided:

- AppleTalk
- Open System Interconnection (OSI)
- DECnet
- VINES
- IP
- Xerox Network Systems (XNS)
- IPX

A router can be configured to simultaneously route multiple protocols over X.25. For example, in Figure 45-5, the local network supports both XNS and TCP/IP traffic and routes information through a single X.25 connection to both types of remote networks.

If you are using X.25 to communicate with multiple routers over a single high-speed serial interface, you must have a fully meshed topology. Configure neighbors so the router can use next-hop split horizon to multiple routers on the same network, or use virtual ports where applicable.

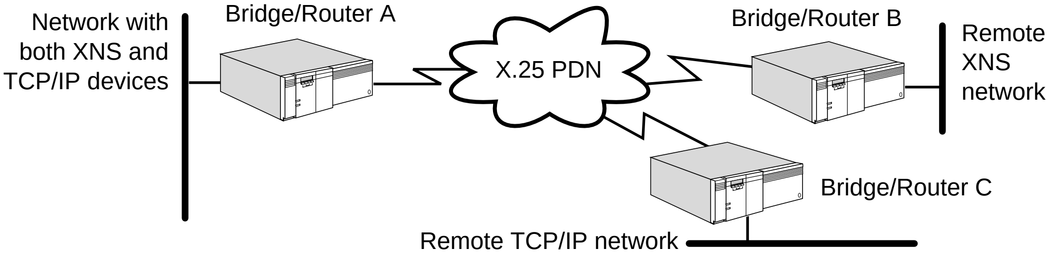


Figure 45-5 Routing Multiple Protocols over X.25 PDN

In this example, bridge/router A must be configured for operation with both XNS and TCP/IP, and the X.25 ports on the remote routers must be configured for their respective protocols.

Be sure that each router attached to the PDN is configured with the same protocol ID.

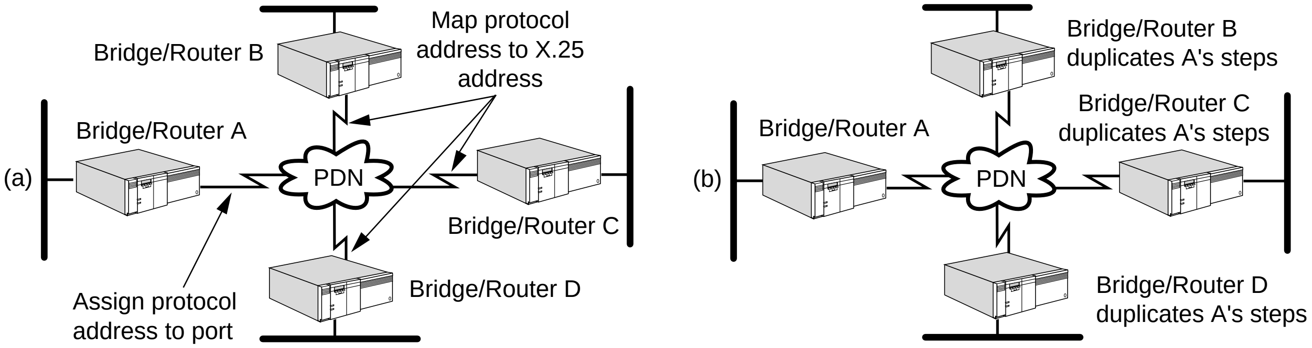

Bridge/Router B
Map protocol address to X.25 address
Bridge/Router A
(a)
PDN
Bridge/Router C
Bridge/Router D
Assign protocol address to port
Bridge/Router B duplicates A's steps
Bridge/Router C duplicates A's steps
Bridge/Router A
(b)
PDN
Bridge/Router D duplicates A's steps

Figure 45-6 Configuration Overview for Routing over X.25

Configuring AppleTalk

To allow the AppleTalk Protocol to operate over an X.25 PDN, you can configure the PDN to operate as either an AppleTalk or non-AppleTalk network. In both cases, the Routing Table Maintenance Protocol (RTMP) packet broadcasts are sent as directed broadcasts every 10 seconds (this is the default) to reach a router configured on a port.

The following section provides information for configuring AppleTalk routing for communication over an X.25 network.

For X.25 ports, split horizon decisions are made at the next router link level instead of at the port level. The next-hop split horizon feature allows support for nonmeshed topologies by allowing a router to use an X.25 port as a virtual hub, sending route information to each router out of the port learned from all other routers out of the same port. If the decisions were made at the port level, as for AppleTalk on LANs and SMDS, no routing information learned from any router out of the port will be sent to any router out of the same port.

Non-AppleTalk Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your AppleTalk LAN according to the procedures in Chapter 14.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the X.25 addresses of each bridge/router participating in AppleTalk routing.

Non-AppleTalk Procedure

To configure AppleTalk routing over an X.25 PDN configured as a non-AppleTalk network, see Figure 45-7 and follow these steps:

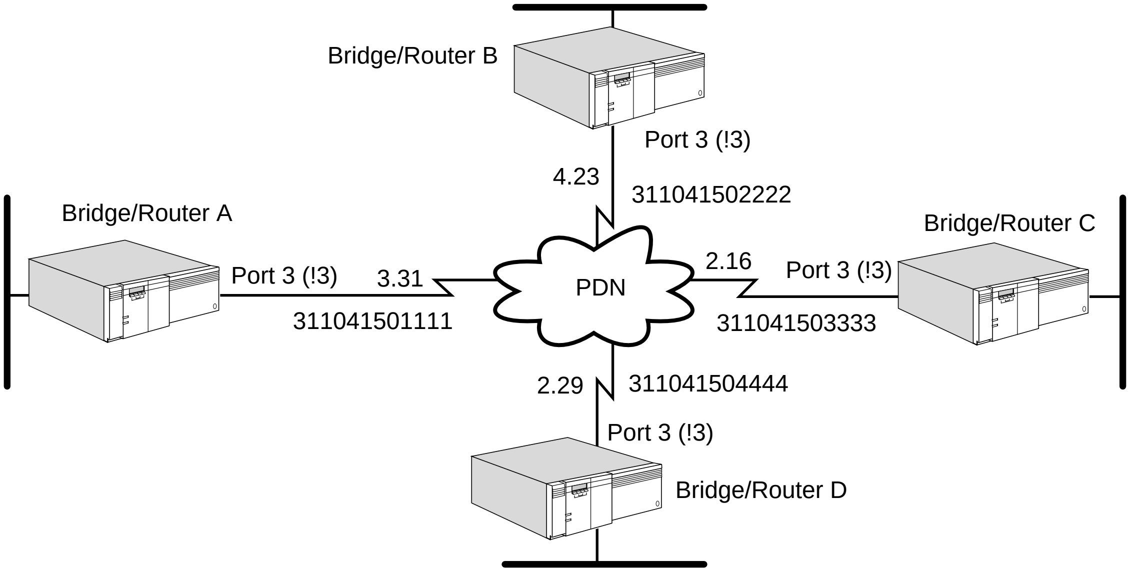


Figure 45-7 Configuring AppleTalk over X.25

1 Configure all the ports on bridge/routers connected to the PDN to be connected to a non-AppleTalk network.

For example, on bridge/routers A, B, C, and D enter:

```
SETDefault !3 -AppleTalk CONTrol = NonAppleTalk
```

2 On each bridge/router, assign the X.25 address of the other bridge/routers connected to the PDN.

For example, on bridge/router A enter:

```
ADD -AppleTalk ADDRess !3 #311041502222
ADD -AppleTalk ADDRess !3 #311041503333
ADD -AppleTalk ADDRess !3 #311041504444
```

Enter similar commands on bridge/routers B, C, and D.

You can dynamically add and delete VCs using the ADDRess parameter.

3 Prioritize AppleTalk traffic over other protocols.

By default, the AppleTalk Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to AppleTalk packets over other traffic. To prioritize AppleTalk packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to "ProfileType" on page 45-2 and to "X.25 Profiles Configuration Examples" on page 45-6 for more information.

b Assign the X.25 user profile to the AppleTalk Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying AppleTalk traffic over port 3. Enter:

```
SETDefault !3 -AppleTalk X25PROFileid = 1
```

4 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all NETBuilder bridge/routers use the hexadecimal value of 0xCA as the AppleTalk protocol identifier. This value ensures acceptance of an incoming call request when AppleTalk routing is enabled.

If you have a bridge/router from another vendor that needs to receive AppleTalk-routed packets, make sure that the protocol ID for all devices matches. You can change the value on the NETBuilder bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -AppleTalk X25ProtID = 22
```

You can enter a hexadecimal value between 0 and FF.

5 Enable routing on each bridge/router by entering:

```
SETDefault !3 -AppleTalk CONTrol = ROute
```

AppleTalk Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your AppleTalk LAN according to the procedures in Chapter 14.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the AppleTalk node address and the X.25 address for each bridge/router participating in AppleTalk routing.

AppleTalk Procedure

To configure the X.25 PDN to operate as an AppleTalk network, refer to Figure 45-7 and follow this procedure.

Use the AppleTalk StartupNET and the StartupNODe commands to configure the local X.25 port's AppleTalk address. This allows the local router to always assign the same AppleTalk node address to the local port, assuming that the address is within the AppleTalk network range of the X.25 cloud. These static configurations are saved on the diskette and only need to be changed when the topology changes.

Set up mapping information between AppleTalk node addresses and X.25 addresses for each bridge/router directly connected to the PDN using the ADD -AppleTalk ADDRess command.

The following sequence of commands sets up an AppleTalk network for an X.25 cloud with four routers (A–D) attached. This example assumes that the AppleTalk network range for the X.25 cloud shared by the configured routers is 2 to 4 and that at least one of the routers is configured to send seed information to any other nonseed routers.

To set up an AppleTalk network for an X.25 cloud, follow these steps:

1 Set the local AppleTalk address before routing is enabled by entering the following commands on bridge/router A:

```
SETDefault !3 -AppleTalk StartupNET = 3
SETDefault !3 -AppleTalk StartupNODe = 31
```

2 Set the local AppleTalk address before routing is enabled by entering the following commands on bridge/router B:

```
SETDefault !3 -AppleTalk StartupNET = 4
SETDefault !3 -AppleTalk StartupNODe = 23
```

3 Set the local AppleTalk address before routing is enabled by entering the following commands on bridge/router C:

```
SETDefault !3 -AppleTalk StartupNET = 2
SETDefault !3 -AppleTalk StartupNODe = 16
```

4 Set the local AppleTalk address before routing is enabled by entering the following commands on bridge/router D:

```
SETDefault !3 -AppleTalk StartupNET = 2
SETDefault !3 -AppleTalk StartupNODe = 29
```

5 Configure static mapping of neighbor X.25 DTE addresses to their AppleTalk node addresses on each bridge/router.

For example, on bridge/router A (AppleTalk address 3.31), enter the following X.25 addresses of the other bridge/routers connected to the PDN:

```
ADD -AppleTalk ADDRess 4.23 #311041502222
ADD -AppleTalk ADDRess 2.16 #311041503333
ADD -AppleTalk ADDRess 2.29 #311041504444
```

Configure static mapping of media addresses on bridge/routers B (AppleTalk address 4.23), C (AppleTalk address 2.16), and D (AppleTalk address 2.29).

You can dynamically add and delete VCs using the ADDRess parameter.

6 Enable the X.25 ports on each router for routing over an AppleTalk network by using:

```
SETDefault !3 -AppleTalk CONTrol = (ROute, AppleTalk)
```

Configuring DECnet

This section provides information for configuring DECnet routing for communication over an X.25 network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your DECnet LAN according to the procedures in Chapter 15.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the DECnet address and X.25 address of each bridge/router participating in DECnet routing.

Procedure

To configure DECnet routing over an X.25 PDN, see Figure 45-8 and follow these steps.

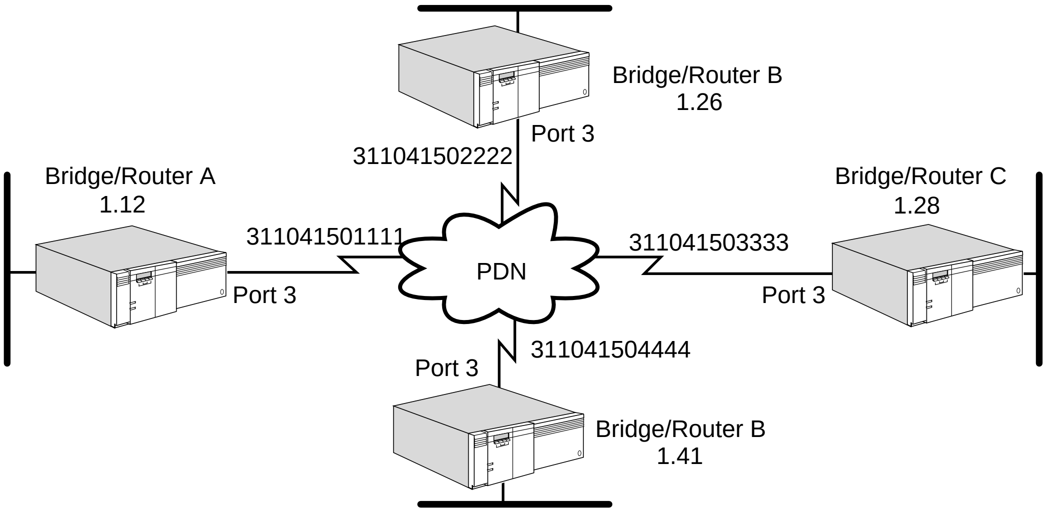


Figure 45-8 Configuring DECnet over X.25

1 Set up mapping information between DECnet addresses and X.25 addresses for each bridge/router end node that is directly connected to the PDN.

Use the ADD !<port> -DECnet Neighbor syntax to set up mapping information. For example, on bridge/router A, enter:

```
ADD !3 -DECnet Neighbor 1.26 #311041502222
ADD !3 -DECnet Neighbor 1.28 #311041503333
ADD !3 -DECnet Neighbor 1.41 #311041504444
```

On bridge/routers B, C, and D, enter similar commands to specify the DECnet-to-X.25 address mapping information.

If you are configuring more than two neighbors, be sure that the X.25 parameters in the DECnet Service are configured as described in the remaining steps. For more information, refer to Chapter 17 in the Reference for NETBuilder Family Software.

2 Prioritize DECnet traffic over other protocols.

By default, the DECnet Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to DECnet packets over other traffic. To prioritize DECnet packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to "ProfileType" on page 45-2 and to "X.25 Profiles Configuration Examples" on page 45-6 for more information.

b Assign the X.25 user profile to the DECnet Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying DECnet traffic over port 3. Enter:

```
SETDefault !3 -DECnet X25PROFileid = 1
```

3 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all NETBuilder II bridge/routers use the hexadecimal value of 0xDE as the DECnet protocol identifier. This value ensures acceptance of an incoming call request when DECnet routing is enabled.

If you have a bridge/router from another vendor that needs to receive DECnet routed packets, make sure that the protocol ID for all devices matches. You can change the value on the NETBuilder II bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -DECnet X25ProtID = 33
```

You can enter a hexadecimal value between 0 and FF.

4 Enable DECnet routing on each port of each bridge/router that is attached to the X.25 PDN.

For example, to enable routing on port 3 of bridge/router A, enter:

```
SETDefault !3 -DECnet CONTrol = ROute
```

Enable routing on bridge/routers B, C, and D.

Configuring IP

This section provides information for configuring IP routing over an X.25 network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 6.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Determine the IP addresses for each wide area port of your bridge/router that is attached to the X.25 network.
- Obtain the IP address and X.25 address of each bridge/router that is attached to the X.25 network and participating in IP routing.

Procedure

To enable IP to operate over an X.25 network, see Figure 45-9 and follow these steps:

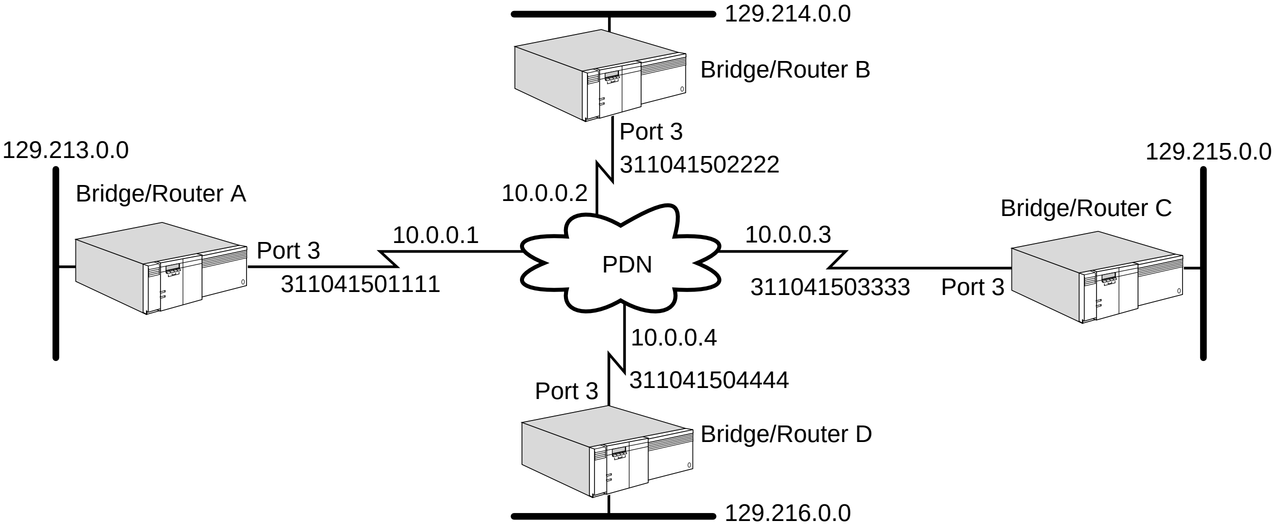


Figure 45-9 Configuring IP over X.25

1 Assign an IP address to each port on each 3Com router that is directly attached to the PDN.

The following command assigns the address 10.0.0.1 to port 3 on bridge/router A:

```
SETDefault !3 -IP NETaddr = 10.0.0.1
```

2 Specify IP to X.25 address mapping information for all neighbors.

The following sequence of commands specifies IP to X.25 address mapping information for the bridge/routers directly attached to the PDN. (In the examples that follow, DTE can be used in place of the pound sign [#].)

For example, enter the following commands on bridge/router A (IP address 10.0.0.1):

```
ADD -IP ADDRess 10.0.0.2 #311041502222
ADD -IP ADDRess 10.0.0.3 #311041503333
ADD -IP ADDRess 10.0.0.4 #311041504444
```

Enter similar commands on bridge/router B (IP address 10.0.0.2), bridge/router C (IP address 10.0.0.3), and bridge/router D (IP address 10.0.0.4), specifying the IP address and DTE mapping information.

3 Optionally, if you are going to be running Open Shortest Path First (OSPF) as the routing protocol over X.25 switched virtual circuits, you can configure a demand interface circuit using:

```
SETDefault !<port> -OSPF DemandInterface = Enable
```

CAUTION: *Do not configure any interface on any router in a single OSPF area as a demand circuit (DC) interface unless all routers in that area have been upgraded to at least software version 8.3.*

With this setting, the router negotiates with the neighbor at the other end of the link. If the neighbor agrees that the link is a demand circuit, the router suppresses sending OSPF Hello packets and routing refresh information, allowing the data link connection to be closed when not carrying application traffic. In order for the demand circuit to be cost-effective, make sure that it is isolated from as many topology changes as possible because topology changes bring up the interface.

For more information, refer to "Reducing Network Costs Using Demand Interface Circuits" on page 6-53.

4 Enable the dynamic routing protocols using Routing Information Protocol-Internet Protocol (RIP-IP) or OSPF for each port and/or virtual port.

- To learn routes dynamically on port 3 using RIPIP, determine if the X.25 network is fully meshed or nonmeshed. If it is fully meshed, then enter:

```
SETDefault !3 -RIPIP CONTrol = (TAlk, Listen, FullMesh)
```

If it is partially meshed or nonmeshed, enter the following command:

```
SETDefault !3 -RIPIP CONTrol = (TAlk, Listen, NonMesh)
```

Setting the CONTrol parameter to the TAlk and Listen values enables the router to send and receive routing information with other routers using RIP. If the FullMesh value is selected, RIP uses normal split horizon; if NonMesh is selected, RIP uses next-hop split horizon.

If the port owner is X.25, the port is up, and the -RIPIP CONTrol parameter is set to TALK, the DynamicNbr option for the -RIPIP and -OSPF CONTrol parameter are automatically enabled, which means that the software automatically adds neighbors and you can skip step 5 and proceed to step 6. If the NoDynamicNbr option for the CONTrol parameter is set, you must add neighbors by completing step 5.

- To enable routes dynamically on port 3 using OSPF, determine whether the X.25 network is fully meshed or nonmeshed.

If the network is fully meshed, enter:

```
SETDefault !3 -OSPF CONTrol = (Enable, FullMesh)
```

If the network is nonmeshed, enter:

```
SETDefault !3 -OSPF CONTrol = (Enable, NonMesh)
```

All of the OSPF neighboring routers must be configured with the same mode: FullMesh or NonMesh. Both of these modes apply to ports as well as virtual ports.

After OSPF operation is enabled, the router exchanges routing information with other routers using OSPF.

5 Specify neighbors for the routing protocols.

a If your network is running RIP, add every router to which the configured router communicates to the neighbor list, either statically configured or learned dynamically.

For example, on bridge/router A, you must add the IP addresses of neighboring bridge/routers B, C, and D:

```
ADD !3 -RIPIP AdvToNeighbor 10.0.0.2
ADD !3 -RIPIP AdvToNeighbor 10.0.0.3
ADD !3 -RIPIP AdvToNeighbor 10.0.0.4
```

On bridge/router B, you must add the IP addresses of neighboring bridge/routers A, C, and D. In addition, add IP addresses of neighboring bridge/routers on bridge/routers C and D.

b If your network is running OSPF, add every router to which the configured router communicates to the neighbor list, either statically configured or dynamically learned.

For example on bridge/router A, you must add the IP addresses of neighboring bridge/routers B, C, and D:

```
ADD !3 -OSPF Neighbor 10.0.0.2
ADD !3 -OSPF Neighbor 10.0.0.3
ADD !3 -OSPF Neighbor 10.0.0.4
```

On bridge/router B, you must add the IP addresses of neighboring bridge/routers A, C, and D. Also, add IP addresses of neighboring bridge/routers on bridge/routers C and D.

6 Prioritize IP traffic over other protocols.

By default, the IP Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to IP packets over other traffic. Currently, you can prioritize all IP packets or specific IP traffic based on IP filters. Refer to “FIlters” on page 29-10 in *Reference for NETBuilder Family Software* to create custom filters.

To prioritize IP packets using an X.25 user profile, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to “ProfileType” on page 45-2 and to “X.25 Profiles Configuration Examples” on page 45-6 for more information.

b Assign the X.25 user profile to the IP Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying IP traffic over port 3. Enter:

```
SETDefault !3 -IP X25PROFileid = 1
```

If a user profile is configured for the IP Service (an IP user profile ID), all IP traffic uses the IP user profile ID. You can also prioritize traffic using the X25Profile action in the FilterAddrs parameter. For example, you can set the FilterAddrs parameter to select different user profile IDs that prioritize Telnet traffic over FTP. The user profiles configured using the FilterAddrs parameter overwrite the IP user profile ID. When separate user profiles are configured for Telnet/FTP traffic using filters, Telnet and FTP can establish separate virtual circuits to carry the traffic, guaranteeing that FTP packets will not take over the virtual circuits. You can adjust the X25WindowSiZe and X25PacketSiZe parameters in the user profile to improve the response of Telnet traffic over X.25.

7 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all NETBuilder II bridge/routers use the hexadecimal value of 0xCC as the IP protocol identifier. This value ensures acceptance of an incoming call request when IP routing is enabled.

If you have a bridge/router from another vendor that needs to receive IP-routed packets, make sure that the protocol ID for all devices matches. You can change the value on the NETBuilder II bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -IP X25ProtID = 44
```

You can enter a hexadecimal value between 0 and FF.

Configuring IPX

This section provides information for configuring IPX routing for communication over an X.25 network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your IPX LAN according to the procedures in Chapter 13.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Determine the IPX network number to be assigned to each bridge/router.

Procedure

To configure IPX to operate over an X.25 PDN, see Figure 45-10 and follow these steps:

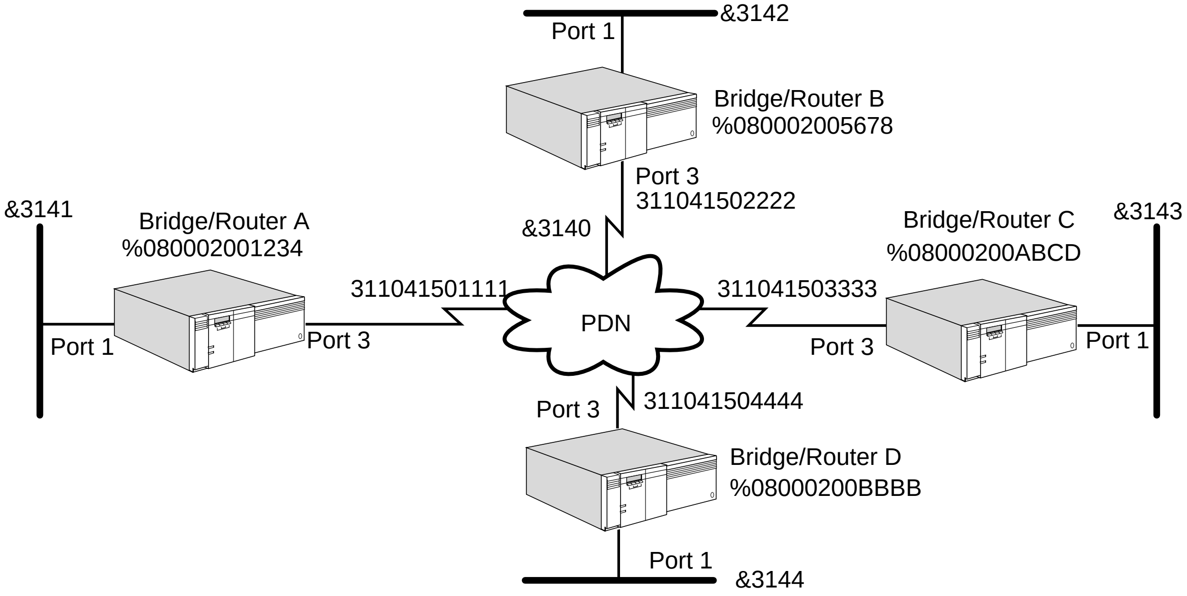


Figure 45-10 Configuring IPX over X.25

1 Assign a network number to each port on each 3Com bridge/router connected to the X.25 PDN.

For example, assign &3140 as the network number to port 3 on bridge/routers A, B, C, and D by entering the following command on each bridge/router:

```
SETDefault !3 -IPX NETnumber = &3140
```

2 Specify IPX network number to X.25 address mapping information for each bridge/router directly connected to the PDN.

For example, on bridge/router A, enter:

```
ADD !3 -IPX ADDRess #311041502222 %080002005678
ADD !3 -IPX ADDRess #311041503333 %08000200ABCD
ADD !3 -IPX ADDRess #311041504444 %08000200BBBB
```

The commands specify IPX to X.25 address mapping information; the network number in each case corresponds to port 3 on the remote bridge/router.

Enter similar commands on bridge/routers B, C, and D.

3 If you are using NetWare Routing Information Protocol (NRIP) and Service Advertising Protocol (SAP) as your routing protocols, verify that routing is enabled on each wide area port of each bridge/router that is attached to the X.25 network by entering:

```
SHow -NRIP CONTrol
```

To verify that Talk and Listen are set, enter the SHow -SAP CONTrol command.

4 If you are using NetWare Link Services Protocol (NLSP) as the routing protocol, follow these steps:

a Make sure the NLSP routing protocol is enabled by entering:

```
SHow -NLSP CONTrol
```

b Skip this step if dynamic neighbor is enabled on the port. Specify the DTE address neighbors that will be taking part in routing over X.25 using:

```
ADD !<port> -NLSP Neighbors #<DTE address>
```

For example on bridge/router A, enter the DTE address of bridge/routers B, C, and D as follows:

```
ADD !3 -NLSP Neighbors #311041502222
ADD !3 -NLSP Neighbors #311041503333
ADD !3 -NLSP Neighbors #311041504444
```

c Display the NLSP adjacencies by entering:

```
SHow -NLSP ADJacencies
```

If you are configuring more than two neighbors, be sure that the X.25 parameters in the PROFile Service are configured as described in the remaining steps. For more information, refer to Chapter 45 in Reference for NETBuilder Family Software.

5 Prioritize IPX traffic over other protocols.

By default, the IPX Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to IPX packets over other traffic. To prioritize IPX packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to “ProfileType” on page 45-2 and to “X.25 Profiles Configuration Examples” on page 45-6 for more information.

b Assign the X.25 user profile to the IPX Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying IPX traffic over port 3. Enter:

```
SETDefault !3 -IPX X25PROFileid = 1
```

6 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all 3Com bridge/routers use the hexadecimal value of 0xEE as the IPX protocol identifier. This value ensures acceptance of an incoming call request from other 3Com routers.

If you have a bridge/router from another vendor that needs to receive IPX-routed packets, make sure that the protocol IDs are compatible. You can change the value on the 3Com bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -IPX X25ProtID = 55
```

You can enter a hexadecimal value between 0 and FF.

You can force the 3Com bridge/router to comply with the RFC 1356 by setting the value to Internet Engineering Task Force (IETF). For example, to change the router to IETF compliancy, enter:

```
SETDefault !3 -IPX X25ProtID = IETF
```

Configuring IPX with Different Software Versions

To configure IPX to operate over an X.25 PDN when bridge/router A is running 6.0 software or later and bridge/router B is running a version earlier than 6.0, refer to Figure 45-11 and follow these steps.

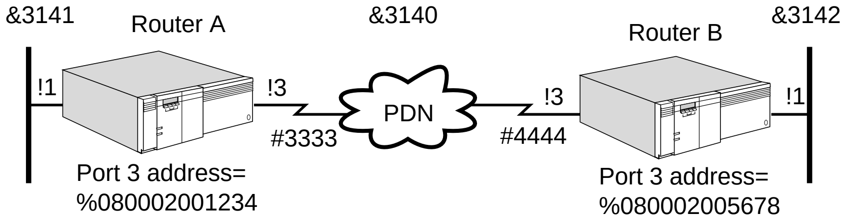


Figure 45-11 Configuring IPX with Different Software Versions

On bridge/router A, follow these steps:

1 Assign a network number to the port that is connected to the X.25 PDN.

For example, assign &3140 as the NETnumber to port 3 on bridge/router A by entering:

```
SETDefault !3 -IPX NETnumber = &3140
```

2 Configure bridge/router A to interoperate with software earlier than 6.0 by using the ripConTRoL parameter:

```
SETDefault !3 -IPX ripConTRoL = OldNbrMap
```

In software release 8.0 and later, use:

```
SETDefault !3 -NRIP CONTrol = OldNbrMap
```

3 Specify an IPX network number to X.25 address mapping information for the bridge/router A port that is directly connected to the PDN.

Using Figure 45-11 as an example, enter:

```
ADD !3 -IPX ADDRess #4444 %080002005678
```

The address is optional.

On bridge/router B, follow these steps:

1 Assign a network number to the port that is connected to the X.25 PDN.

Using Figure 45-11 as an example, assign &3140 as the network number to port 3 on bridge/router B by entering:

```
SETDefault !3 -IPX NETnumber = &3140
```

2 Specify an IPX network number to X.25 address mapping information for the bridge/router B port that is directly connected to the PDN.

Use Figure 45-11 as an example, enter:

```
ADD !3 -IPX ADDRess &3141 #3333
```

When adding a neighbor to bridge/router B, you must assign the Router A port 1 network number (&3141) to the bridge/router B port.

Configuring OSI

This section provides information for configuring OSI routing for communication over an X.25 network.

Prerequisites

Before beginning the procedure, decide whether to use the PrefixRoute parameter or the Neighbors parameter using the following criteria:

- Use the PrefixRoute parameter if you view the remote site as another routing domain (for example, another company) with different NSAP addresses. The PrefixRoute parameter allows you to specify interdomain reachability information without exchanging Intermediate System-to-Intermediate System (IS-IS) packets.
- Use the Neighbors parameter if the remote site is part of your routing domain. The neighbor information instructs the IS-IS Protocol to exchange packets and establish full connectivity.

In addition, you need to complete the following tasks:

- Configure your OSI LAN according to the procedures in Chapter 16.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- If you are using the PrefixRoute parameter, obtain the NSAP address prefix and the X.25 address for each bridge/router participating in OSI routing.
- If you are using the Neighbors parameter, obtain the X.25 address of each bridge/router participating in OSI routing.

Procedure

To configure OSI routing, see Figure 45-12 and follow these steps. If you want to use the PrefixRoute parameter, begin with step 1. If you want to use the Neighbors parameter, skip step 1 and begin with step 2.

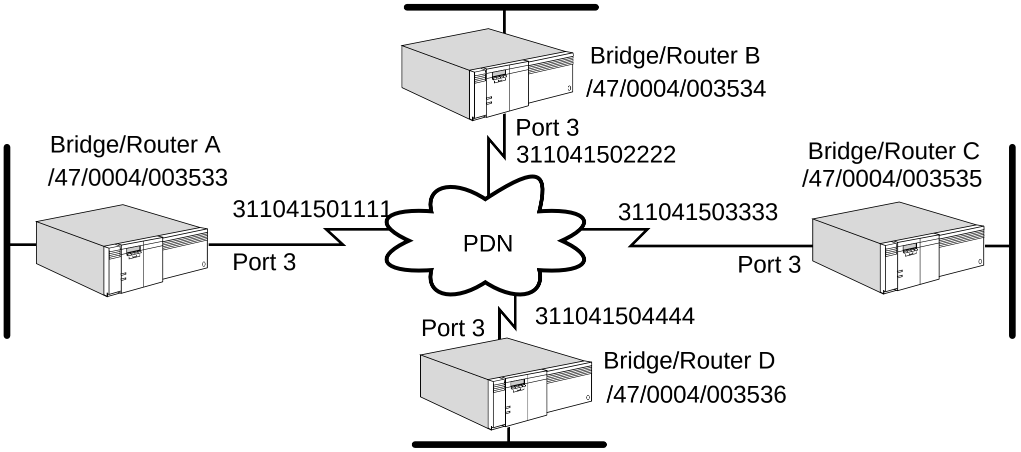


Figure 45-12 Configuring OSI over X.25

1 Using the PrefixRoute parameter, specify an OSI address prefix and corresponding X.25 The MODE parameter in the ISIS Service must be set to L2 for the PrefixRoute parameter to take effect.

For example, on bridge/router A, enter:

```
ADD !3 -ISIS PrefixRoute /47/0004/003534 #311041502222
ADD !3 -ISIS PrefixRoute /47/0004/003535 #311041503333
ADD !3 -ISIS PrefixRoute /47/0004/003536 #311041504444
```

Enter similar commands on bridge/router B, C, and D, specifying OSI-to-X.25 address mapping information.

Proceed to step 3.

2 Using the Neighbors parameter, specify an X.25 address for any neighbors on the X.25 PDN that support IS-IS.

IS-IS operates over X.25 in a point-to-point manner and does not require a fully meshed connectivity between all the bridge/routers.

Using Figure 45-12 as an example, if bridge/router B supports IS-IS and you want to operate it over X.25, you would enter the following command from bridge/routers A, C, and D:

```
ADD !3 -ISIS Neighbors #311041502222
```

On bridge/router B, enter:

```
ADD !3 -ISIS Neighbors #311041501111
ADD !3 -ISIS Neighbors #311041503333
ADD !3 -ISIS Neighbors #311041504444
```

3 Prioritize OSI traffic over other protocols.

By default, the OSI Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to OSI packets over other traffic. To prioritize OSI packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to "ProfileType" on page 45-2 and to "X.25 Profiles Configuration Examples" on page 45-6 for more information.

b Assign the X.25 user profile to the OSI Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying OSI traffic over port 3. Enter:

```
SETDefault !3 -CLNP X25PROFileid = 1
```

Configuring VINES

This section provides information for configuring VINES routing for communication over an X.25 network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your VINES LAN according to the procedures in Chapter 17.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the X.25 addresses of each bridge/router participating in VINES routing.

Procedure

To enable the VINES Protocol to operate over an X.25 PDN, see Figure 45-13 and follow these steps:

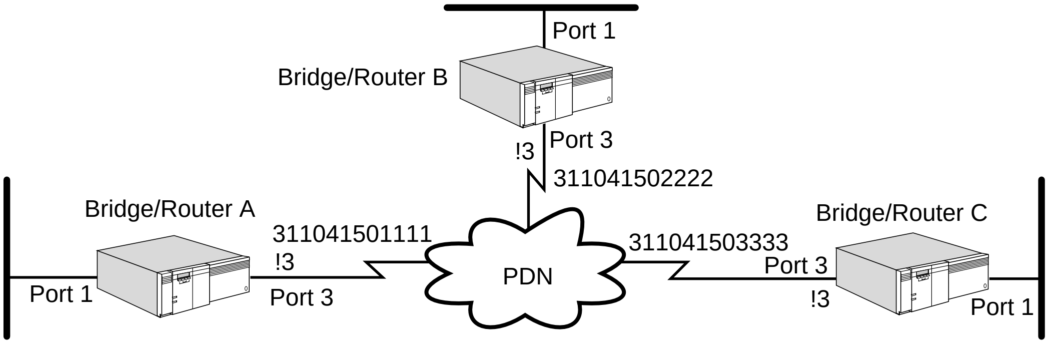


Figure 45-13 Configuring VINES over X.25

1 Specify X.25 DTE addresses for port or virtual ports.

For example, on bridge/router A, enter:

```
ADD !3 -VIP WideAreaNbr #311041502222
ADD !3 -VIP WideAreaNbr #311041503333
```

Enter similar commands on bridge/routers B and C, specifying the DTE addresses for the ports.

2 Prioritize VINES traffic over other protocols.

By default, the VINES Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to VINES packets over other traffic. To prioritize VINES packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to “ProfileType” on page 45-2 and to “X.25 Profiles Configuration Examples” on page 45-6 for more information.

b Assign the X.25 user profile to the VINES Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying VINES traffic over port 3. Enter:

```
SETDefault !3 -VIP X25PROFileid = 1
```

3 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all NETBuilder II bridge/routers use the hexadecimal value of 0xBC as the VINES protocol identifier. This value ensures acceptance of an incoming call request when VINES routing is enabled.

If you have a bridge/router from another vendor that needs to receive VINES-routed packets, make sure that the protocol ID for all devices matches. You can change the value on the NETBuilder II bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -VIP X25ProtID = 66
```

Configuring XNS

The section provides information for configuring XNS routing for communication over an X.25 network.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your XNS LAN according to the procedures in Chapter 18.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Determine the XNS network number to be assigned to each bridge/router.
- Obtain the MAC address and X.25 address of each remote host participating in XNS routing.

Procedure

To enable the XNS Protocol to operate over an X.25 PDN, see Figure 45-14 and follow these steps:

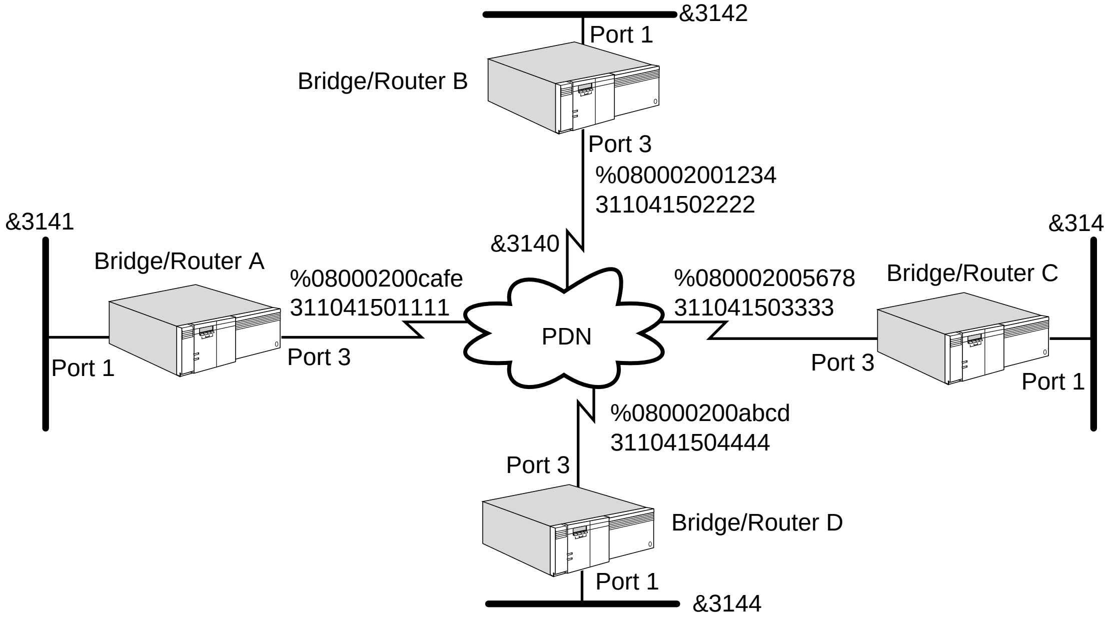


Figure 45-14 Configuring XNS over X.25

1 Assign a network number to each port on each 3Com router that is connected to the X.25 PDN.

For example, to assign &3140 as the network number to port 3 on bridge/routers A, B, C, and D, enter the following command on each router:

```
SETDefault !3 -IDP NETnumber = &3140
```

2 Set up mapping information between NETnumber and X.25 addresses for each bridge/router directly connected to the PDN.

Using Figure 45-14 as an example, the following sequence of commands specifies network number to X.25 address mapping information. The network number in each case corresponds to port 3 on the remote bridge/router.

For example, enter the following commands on bridge/router A:

```
ADD !3 -RIPXNS ADDRess %080002001234 #311041502222
ADD !3 -RIPXNS ADDRess %080002005678 #311041503333
ADD !3 -RIPXNS ADDRess %08000200abcd #311041504444
```

Enter similar commands on bridge/routers B, C, and D, specifying the MAC address and the X.25 address mapping information.

3 Prioritize XNS traffic over other protocols.

By default, the XNS Protocol does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to XNS packets over other traffic. To prioritize XNS packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to “ProfileType” on page 45-2 and to “X.25 Profiles Configuration Examples” on page 45-6 for more information.

b Assign the X.25 user profile to the IDP Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying XNS traffic over port 3. Enter:

```
SETDefault !3 -IDP X25PROFileid = 1
```

4 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all NETBuilder II bridge/routers use the hexadecimal value of 0xC0 as the XNS protocol identifier. This value ensures acceptance of an incoming call request when XNS routing is enabled.

If you have a bridge/router from another vendor that needs to receive XNS-routed packets, make sure that the protocol ID for all devices matches. You can change the value on the NETBuilder II bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -IDP X25ProtID = 77
```

You can enter a hexadecimal value between 0 and FF.

Procedure

To configure XNS to operate over an X.25 PDN when bridge/router A is running 5.0 software or later and bridge/router B is running an earlier version, see Figure 45-15 and follow these steps:

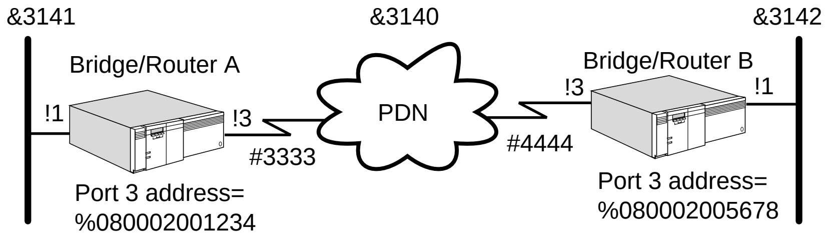


Figure 45-15 Enabling XNS Across a PDN Between Two Neighbors With Different Software Versions

On bridge/router A, follow these steps:

1 Assign a network number to the port that is connected to the X.25 PDN.

Assign &3140 as the network number to port 3 on bridge/router A by entering:

```
SETDefault !3 -IDP NETnumber = &3140
```

2 Configure bridge/router A to interoperate with software earlier than 5.0 by entering:

```
SETDefault !3 -RIPXNS CONTrol = OldNbrMap
```

3 Specify XNS-to-X.25 address mapping information for the bridge/router A port that is directly connected to the PDN by entering the following command:

```
ADD !3 -RIPXNS ADDRess %080002005678 #4444
```

On bridge/router B, follow these steps:

1 Assign a network number to the port that is connected to the X.25 PDN.

For example, to assign &3140 as the network number to port 3 on bridge/router B, enter:

```
SETDefault !3 -IDP NETnumber = &3140
```

2 Specify XNS-to-X.25 address mapping information for the bridge/router B port that is directly connected to the PDN.

For example, use the following command to specify the XNS-to-X.25 address mapping information.

```
ADD !3 -RIPXNS ADDRess &3141 #3333
```

When adding a neighbor on bridge/router B, it must use the network number of port 1 on bridge/router A.

The NETBuilder II bridge/router by default specifies addresses in canonical format, and a SuperStack II NETBuilder bridge/router model 327 or 527 by default specifies addresses in noncanonical format. When connecting the two platforms using an X.25 link running XNS, the NETBuilder II will not know that model 327 or 527 is a token ring platform. The token ring models will not know that the NETBuilder II is an Ethernet platform. You must configure each platform as a static neighbor to the other platform and specify the neighbor's address in canonical format for Ethernet and noncanonical format for token ring. Use:

```
ADD !<port> -RIPXNS ADDRess %<host> <media address>
```

When using this syntax on the NETBuilder II, you must specify the remote host address in noncanonical format to indicate that the remote host is a token ring platform (model 327). When using this syntax on model 327, you must specify the remote host address in canonical format to indicate that the remote host is an Ethernet platform (NETBuilder II).

Setting Up Bridging over X.25

This section describes how to configure your bridge to forward packets over X.25.

Bridging over X.25 requires two or more 3Com bridges to be connected over one or more X.25 PDNs to access nodes on remote LANs. The bridge will not learn from DTEs that are not preconfigured as a neighbor.

Configuring Transparent Bridging

This section describes how to configure transparent bridging.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the transparent bridging procedures in Chapter 3.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the X.25 addresses of each bridge/router participating in transparent bridging.

Procedure

To configure transparent bridging over an X.25 PDN, see Figure 45-16 and follow these steps:

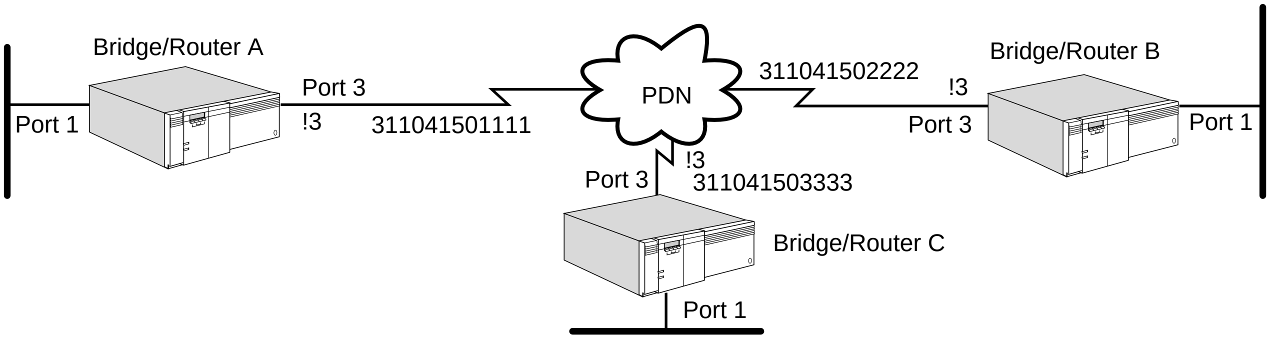


Figure 45-16 Configuring Transparent Bridging over X.25

1 Verify that transparent bridging is enabled on each 3Com bridge port that is directly connected to the X.25 PDN.

By default, transparent bridging is enabled on all NETBuilder II bridge/routers. To verify the setting, use:

```
SHow [!<port>] -BRidge TransparentBRidge
```

If transparent bridging has been disabled, you can enable it on port 3 of Bridge/router A, B, and C by entering the following command on each of these devices:

```
SETDefault !3 -BRidge TransparentBRidge = TransparentBRidge
```

2 Enable the bridge by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

3 Configure all DTEs on the PDN as neighbors that will take part in bridging over X.25.

Perform this step for nonvirtual ports only.

You can configure a maximum of eight neighbors per port.

To configure a neighbor, on bridge/router A enter:

```
ADD !3 -BRidge X25Neighbor = 311041502222
ADD !3 -BRidge X25Neighbor = 311041503333
```

Enter similar commands on bridge/routers B and C to configure the DTEs on the PDN as neighbors.

4 Prioritize bridge traffic.

By default, the BRidge Service does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to bridged packets over other traffic. To prioritize bridged packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to "ProfileType" on page 45-2 and to "X.25 Profiles Configuration Examples" on page 45-6 for more information.

b Assign the X.25 user profile to the BRidge Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying the bridged traffic over port 3. Enter:

```
SETDefault !3 -BRidge X25PROFileid = 1
```

With the current X25VCLimit default, the BRidge Service can establish more than one virtual circuit to a destination. Because the number of virtual circuits is greater than one, packets may not be received in the order in which they were sent. For some bridge-only protocols, such as local area transport (LAT), the sequence of packets needs to be maintained. If the bridged environment consists of these types of protocols, you must create an X.25 user profile with the X25VCLimit parameter set to 1, and assign this profile ID in the BRidge Service. Mnemonic filters can be used to prioritize bridged traffic over X.25. For example, you can configure mnemonic filters for IP and IPX. You can also assign user profiles that are different from the bridge profile ID. All bridged IP and IPX traffic can establish separate virtual circuits for carrying the traffic. Remaining bridged traffic uses the bridge user profile ID.

5 Specify a protocol identifier to be included in an outgoing X.25 call request.

By default, all NETBuilder II bridge/routers use the hexadecimal value of 0xDD as the transparent bridging protocol identifier. This value ensures acceptance of an incoming call request when transparent bridging is enabled.

If you have a bridge/router from another vendor that needs to receive transparent bridging packets, make sure that the protocol ID for all devices matches. You can change the value on the NETBuilder II bridge/routers by using the X25ProtID parameter. For example, to change the value on port 3 of bridge/router A, enter:

```
SETDefault !3 -BRidge X25ProtID = 11
```

You can enter a hexadecimal value between 0 and FF.

Configuring Source Route Bridging

This section provides information for configuring source route bridging over X.25. For more information about source route bridging, see Chapter 5.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the source route bridging procedures in Chapter 5.
- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the X.25 addresses of each bridge/router participating in source route bridging.

Procedure

To configure source route bridging over X.25, follow these steps:

1 Configure all DTEs on the source routing X.25 port as neighbors using this syntax:

```
ADD !<port> -BRidge X25Neighbor = <address>
```

You can configure a maximum of eight neighbors per port.

Perform this step for virtual ports only.

On bridge/router A, enter

```
ADD !3 -BRidge X25Neighbor = 311041502222
ADD !3 -BRidge X25Neighbor = 311041503333
```

Enter similar commands on bridge/routers B and C to configure the DTEs on the PDN as neighbors.

2 Assign a unique ring number to the logical ring associated with each X.25 source routing port.

The ring number can be any number in the range 1 to 4,095, and can be entered in either decimal or hexadecimal format using:

```
SETDefault !<port> -SR RingNumber = <number>(1–4095) |
 0x<number>(1–FFF)
```

You can enter the ring number in decimal or hexadecimal format. Precede the hexadecimal number with 0x.

For more information about ring numbers, refer to Chapter 56 in *Reference for NETBuilder Family Software.*

3 Prioritize bridge traffic.

By default, the BRidge Service does not have an X.25 user profile configured. Configure X.25 user profiles only if you want to assign a priority to bridged packets over other traffic. To prioritize bridged packets, follow these steps:

a Use the -PROFile ProfileType parameter to create an X.25 user profile.

Refer to “ProfileType” on page 45-2 and to “X.25 Profiles Configuration Examples” on page 45-6 for more information.

b Assign the X.25 user profile to the BRidge Service using the X25PROFileid parameter.

For example, suppose you want to use user profile 1 for carrying the bridged traffic over port 3. Enter:

```
SETDefault !3 -BRidge X25PROFileid = 1
```

With the current X25VCLimit default, the BRidge Service can establish more than one virtual circuit to a destination. Because the number of virtual circuits is greater than one, packets may not be received in the order in which they were sent. For some bridge-only protocols, such as LAT, the sequence of packets needs to be maintained. If the bridged environment consists of these types of protocols, you must create an X.25 user profile with the X25VCLimit parameter set to 1, and assign this profile ID in the BRidge Service.

Mnemonic filters can be used to prioritize bridged traffic over X.25. For example, you can configure mnemonic filters for IP and IPX. You can also assign user profiles that are different from the bridge profile ID. All bridged IP and IPX traffic can establish separate virtual circuits for carrying the traffic. Remaining bridged traffic uses the bridge user profile ID.

4 Verify that source route bridging is enabled on the wide area port using:

```
SHow !<port> -SR SrcRouBridge
```

If source route bridging is disabled, you need to enable it for your wide area port:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

5 If you want to run both source route and transparent bridging on a NETBuilder II bridge/router, skip this step and go on to step 6. If you want to run source route bridging only on a NETBuilder II bridge/router, disable transparent bridging on the wide area port using:

```
SETDefault !<port> -BRidge TransparentBRidge = NoTransparentBRidge
```

This step does not apply to model 32x and 52x SuperStack II NETBuilder bridge/router. Transparent bridging is not supported on these models.

6 Verify that bridging is enabled by entering:

```
SHow -BRidge CONFiguration
```

If bridging has been disabled, enable it for the system by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

Setting Up a Permanent Virtual Circuit Connection

This section describes how to set up permanent virtual circuits (PVC) on an X.25 interface. A fixed point-to-point connection can use a PVC to emulate a leased or private line. X.25 PVCs can be set up on routed configurations to transmit and receive data over an X.25 interface on public data networks.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Set up the X25 Service as described in "Setting Up the X25 Service" on page 45-1.
- Obtain the X.25 addresses of the destination bridge/router participating in the PVC.
- Create an X.25 user profile to assign the target DTE for a the desired routing protocol.

Procedure

To configure an X.25 PVC, follow these steps:

1 Add the port using:

```
ADD! <port> -X25 PVC <lcn1> [,lcn2] <destination dte address>
  <protocol ID> [<user profID]
```

For example, to set up a PVC on port 2 of your bridge/router, enter:

```
ADD !2 -X25 PVC 1,2 31102255731 CC
```

This command creates a PVC connection on port 2. This PVC carries IP traffic, specified by protocol ID CC, to and from DTE address 311022255731 on logical channel numbers 1 and 2.

2 To verify the X.25 PVC configuration, enter:

```
SHow !2 -X25 PVC
```

The PVCs configured on port 2 are displayed.

How X.25 Works

This section describes the X25 Service.

Fully Meshed, Partially Meshed, and Nonmeshed Topologies

A fully meshed X.25 topology is a topology where each node on a network is directly connected to all other nodes on the network. Each node is connected to the other nodes through a virtual circuit, and each virtual circuit has a DTE associated with it. Figure 45-17 shows an example of a fully meshed X.25 topology.

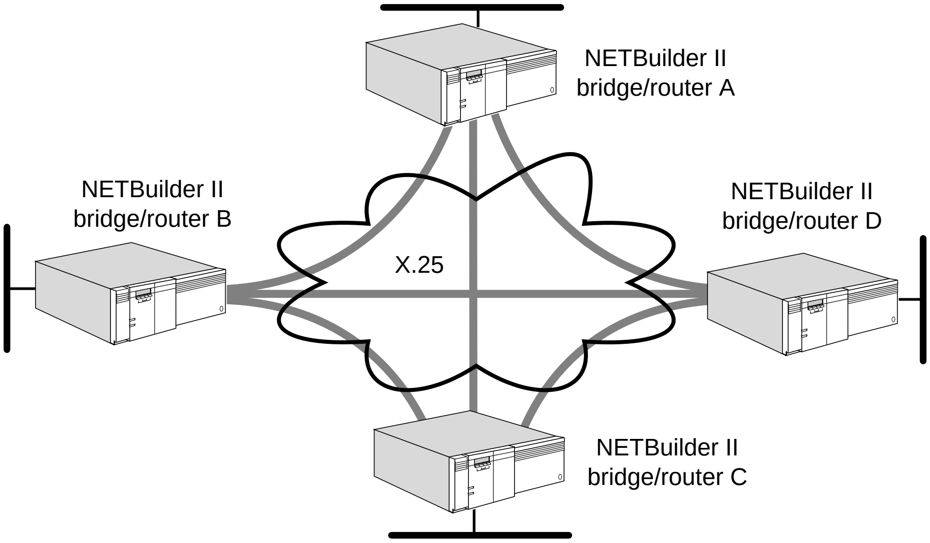


Figure 45-17 Fully Meshed X.25 Topology

The topology in Figure 45-17 consists of NETBuilder II bridge/routers. Through the established virtual circuits, bridge/router A is connected to bridge/routers B, C, and D; bridge/router B is connected to bridge/routers A, C, and D; and so on.

A nonmeshed X.25 topology is a topology where each node on a network is not necessarily connected to all other nodes on the network. Figure 45-18 shows an example of a nonmeshed X.25 topology.

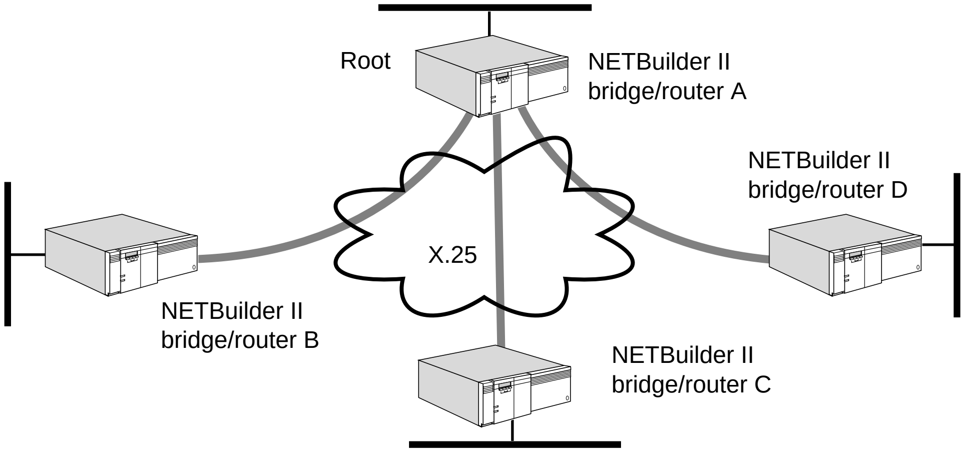


Figure 45-18 Nonmeshed X.25 Topology

The topology in Figure 45-18 consists of NETBuilder II bridge/routers. Through the established virtual circuits, bridge/router A is connected to bridge/routers B, C, and D. bridge/routers B, C, and D are connected to bridge/router A only, but not to one another.

Two possible solutions exist to work around the lack of connectivity between bridge/routers B, C, and D. If you are routing IP-RIP, IPX, or AppleTalk, these protocols offer the next-hop split horizon feature. In IP-RIP, this feature is enabled when -RIPIP CONTrol is set to NonMesh. In IPX, it is enabled by manually configuring neighbors. In AppleTalk, next-hop split horizon is configured by adding static mappings to the address mapping table.

For example, if you are routing IP-RIP, use the SETDefault !<port> -RIPIP CONTrol = NonMesh syntax. If you are routing IPX, you can configure bridge/routers B, C, and D as neighbors using the PolicyControl and AdvToNeighbor parameters in the -NRIP and SAP Services. If routing AppleTalk, you can add the address of bridge/routers B, C, and D to an address mapping table. After taking such action, bridge/router A, the root bridge/router, learns available routes from each neighbor and then updates each neighbor with available routes other than that particular neighbor's own routes. Even though bridge/routers B, C, and D are not directly connected to one another, they can still learn of routes other than their own through bridge/router A. For more information on next-hop split horizon, refer to Chapter 14, Chapter 6, and Chapter 13.

Another solution in a topology where there is a lack of connectivity is to create virtual ports. Virtual ports are supported by bridging and all routing protocols over an X.25 network. You must use virtual ports in a Boundary Routing over X.25 topology and when bridging or routing DECnet, IP-OSPF, VINES, or XNS over X.25 in a partially meshed or nonmeshed topology. Using virtual ports in all other bridging or routing scenarios over an X.25 network is optional.

For information on the number of virtual ports supported per platform, see Table 1-1 in Chapter 1.

Virtual ports allow the creation of multiple logical ports on one path. Each virtual circuit attaches a separate logical network. Figure 45-19 shows a Boundary Routing over X.25 topology where virtual ports are configured. In this topology, even though the SuperStack II NETBuilder boundary routers are not directly connected to one another, information about each of their networks can still be propagated through the NETBuilder II bridge/router.

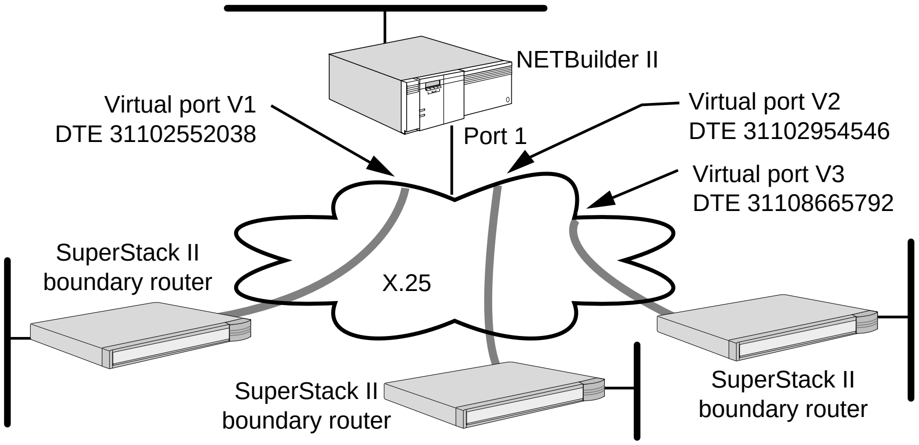


Figure 45-19 Using Virtual Ports in a Boundary Routing Over X.25 Topology

For more information on virtual ports and Boundary Routing over X.25, refer to Chapter 1 and Chapter 32, respectively.

A partially meshed X.25 topology is a topology where some nodes on a network are directly connected to all other nodes on the network (as in a fully meshed topology) and other nodes are not (as in a nonmeshed topology). Figure 45-20 shows an example of a partially meshed X.25 topology.

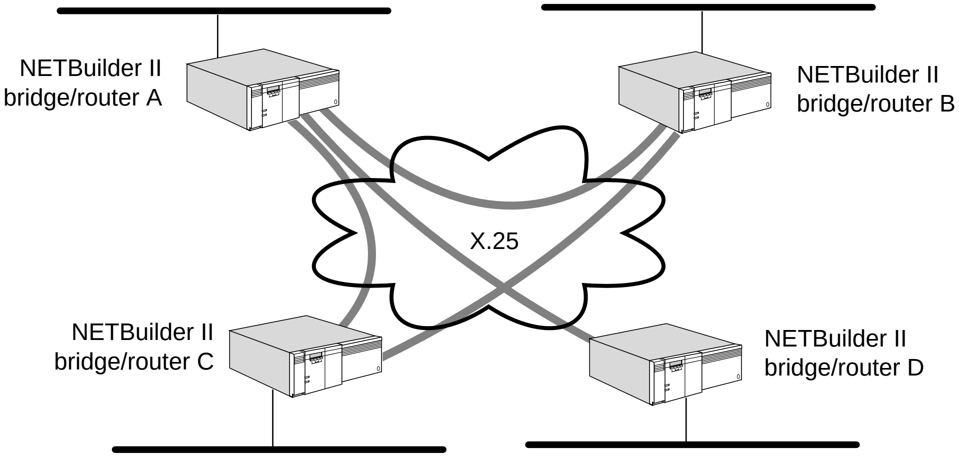


Figure 45-20 Partially Meshed X.25 Topology

The topology in Figure 45-20 is composed of four NETBuilder II bridge/routers. Through the established virtual circuits, bridge/routers A, B, and C are connected to one another, but bridge/router D is connected to bridge/router A only.

The lack of connectivity between bridge/routers B, C, and D can be worked around using the same two solutions discussed earlier in this section that apply to nonmeshed topologies.

Facilities

In addition to the basic X.25 functionality that is supported by all PDNs, another feature called *facilities* is optionally supported on some PDNs. Use of facilities is controlled at subscription time or on a call-by-call basis, depending on the facility.

The bridge/router supports the following facilities:

- Flow-control negotiation
- Throughput class negotiation
- Closed user group
- Fast select
- Fast select acceptance

46 CONFIGURING LOCAL AND GLOBAL SWITCHING

This chapter describes procedures for configuring the XSWitch Service on your bridge/router. The XSWitch Service consists of two features, local switching and global switching (X.25 tunneling over IP).

X.25 local switching allows the NETBuilder bridge/router to take an incoming call from a high-speed serial (HSS) port that is not targeted for the bridge/router itself and forward the call to its real X.25 destination by switching it over an X.25 WAN on another locally attached HSS port.

Global switching allows the bridge/router to take an incoming X.25 call that is not targeted for the bridge/router itself and, instead of switching the call to another HSS port, encapsulate and forward it through a locally attached IP Internet to another IP peer for further switching.

When a bridge/router is configured for a switched virtual circuit and switching occurs, a *switched virtual circuit* is established. The switched virtual circuit is disconnected automatically when communication is complete.

For definitions of switching terms, refer to "Switching Terms" on page 46-3.

Setting Up Local Switching on a SVC

This section describes how to configure local switching on a switched virtual circuit. Figure 46-1 shows a bridge/router using local switching to forward an X.25 call from WAN #1 to WAN #2.

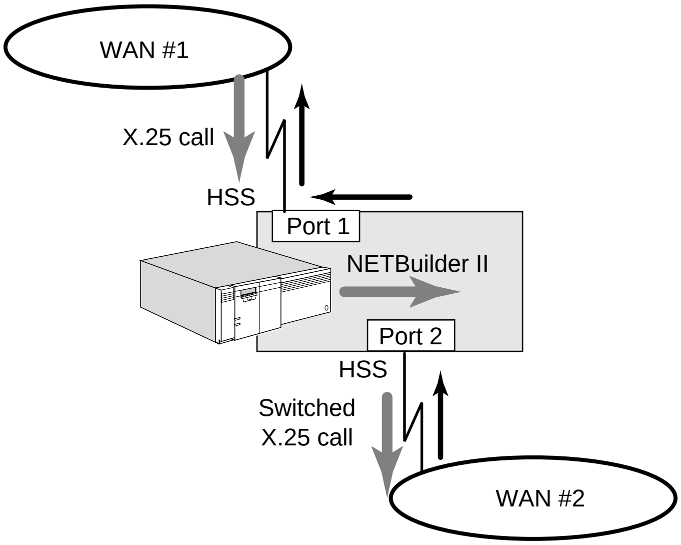


Figure 46-1 Local Switching

When the XSWitch Service receives an incoming X.25 call, it looks in the X25Prefix table to find an entry whose X.25 address prefix matches the address of the called address. When a match is found, its associated HSS port is used for switching. These X.25-prefix-to-HSS-port entries are user-configurable.

To configure local switching, following these steps:

1 Verify that local switching is enabled by entering:

```
SHow -XSWitch CONTrol
```

If local switching is not enabled, enable it entering:

```
SETDefault -XSWitch CONTrol = LoclSW
```

2 Assign X.25 prefix addresses to your HSS ports.

For example, to assign an X.25 prefix address of 5109 to port 2, enter:

```
ADD !2 -XSWitch X25Prefix 5109
```

For more information, refer to x on page 66-3 in *Reference for NETBuilder Family Software*.

Setting Up Global Switching on an SVC

This section describes how to configure global switching (X.25 tunneling over IP). Figure 46-2 shows a bridge/router using tunneling to forward an X.25 call from WAN #1 to WAN #2.

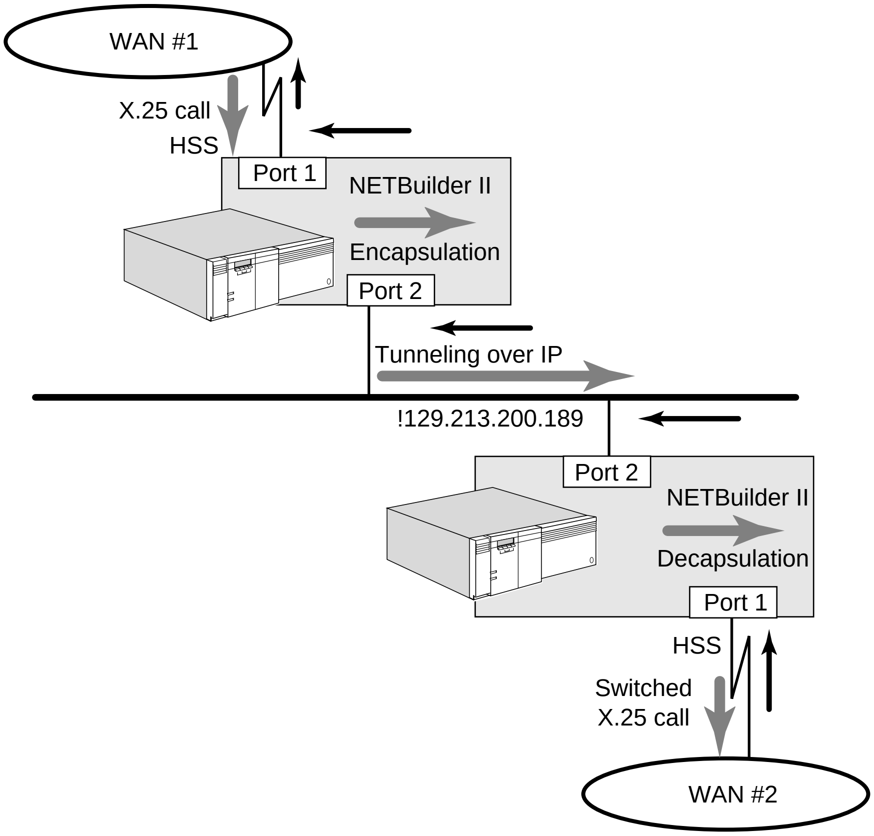


Figure 46-2 Global Switching

When the XSWitch Service receives an incoming X.25 call, it looks in the X25Prefix table to find an entry whose X.25 address prefix matches the address of the called address. When a match is found, its associated IP address is used for switching. These X.25-prefix-to-IP-address entries are user-configurable.

To configure global switching, follow these steps:

1. Verify that global switching is enabled by entering:

   ```
   SHow -XSWitch CONTrol
   ```

   If global switching is not enabled, enable it by entering:

   ```
   SETDefault -XSWitch CONTrol = GlobSW
   ```

2. Assign X.25 prefix addresses to your IP addresses.

   For example, to assign an X.25 prefix address of 5109 to an IP address of 129.213.200.189, enter:

   ```
   ADD !129.213.200.189 -XSWitch X25Prefix 5109
   ```

For additional parameters that affect global switching, refer to Chapter 66 in *Reference for NETBuilder Family Software*.

Switching Terms

The following terms are used in this chapter to explain switching:

tunneling service	A method of connecting peer internets that are not physically reachable with the X.25 Protocol. This is a generic service on NETBuilder bridge/routers. Global switching interfaces with it to set up and maintain the tunnel between two entities over the Internet.
encapsulation	Conveying an X.25 packet within a TCP data packet so it can be forwarded through a TCP connection.
decapsulation	Extracting an X.25 packet encapsulated in a TCP data packet for further forwarding through a locally attached X.25 WAN.

47 CONFIGURING INTERNETWORKING USING ATM

This chapter describes how to configure your NETBuilder II bridge/router to establish LAN, WAN, and MAN connectivity through Asynchronous Transfer Mode (ATM).

For conceptual information, refer to "How ATM Works" on page 47-11.

The bridge/router supports both bridging and routing of multiple protocols over ATM. The ATM Service allows your bridge/router to transmit and receive data over a permanent virtual circuit (PVC) with any other device on the ATM network. You can achieve multiprotocol encapsulation over ATM through PVCs by upgrading to software version 9.0 and installing the MP ATMLink module in your NETBuilder II bridge/router. In this configuration, your bridge/router supports operation over ATM adaptation layer 5 (AAL5) and router cluster topologies in meshed, partially meshed, and nonmeshed topologies.

Setting Up the ATM Service

This section describes how to configure your bridge/router to transmit and receive data over an ATM interface using PVCs with the following protocols:

- Transparent bridging
- Source Route bridging
- IP routing
- IPX routing

You must follow the steps in this section whether you are configuring for bridging or for routing. After you have completed these steps, proceed to "Configuring Transparent Bridging" on page 47-5, "Configuring Source Route Bridging on page 47-6, "Configuring IP Routing" on page 47-7, or "Configuring IPX Routing" on page 47-9.

For detailed descriptions of all commands, see *Reference for NETBuilder Family Software*.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your bridge/router ports, virtual ports, and paths according to Chapter 1.
- Obtain the ATM addresses (VPI.VCI) from your ATM service provider or the ATM switch.
- Determine if you have a partially meshed or nonmeshed topology.

If you plan to enable the Internet Protocol-Routing Information Protocol (IP-RIP) or the Internetwork Packet Exchange (IPX) routing protocol, you need to make certain that the next-hop split horizon feature is enabled. If you have a partially meshed or nonmeshed topology, and you plan to enable Open Shortest Path First (OSPF), make sure that you set -OSPF CONTrol to NonMesh to enable the point-to-multipoint interface. For information on meshed, partially meshed, and nonmeshed topologies, next-hop split horizon, and virtual ports, refer to "How ATM Works" on page 47-11. For instructions on setting up virtual ports, refer to Chapter 1.

Procedure To transmit and receive data over an ATM network, refer to Figure 47-1 and follow these steps on both ends of the link:

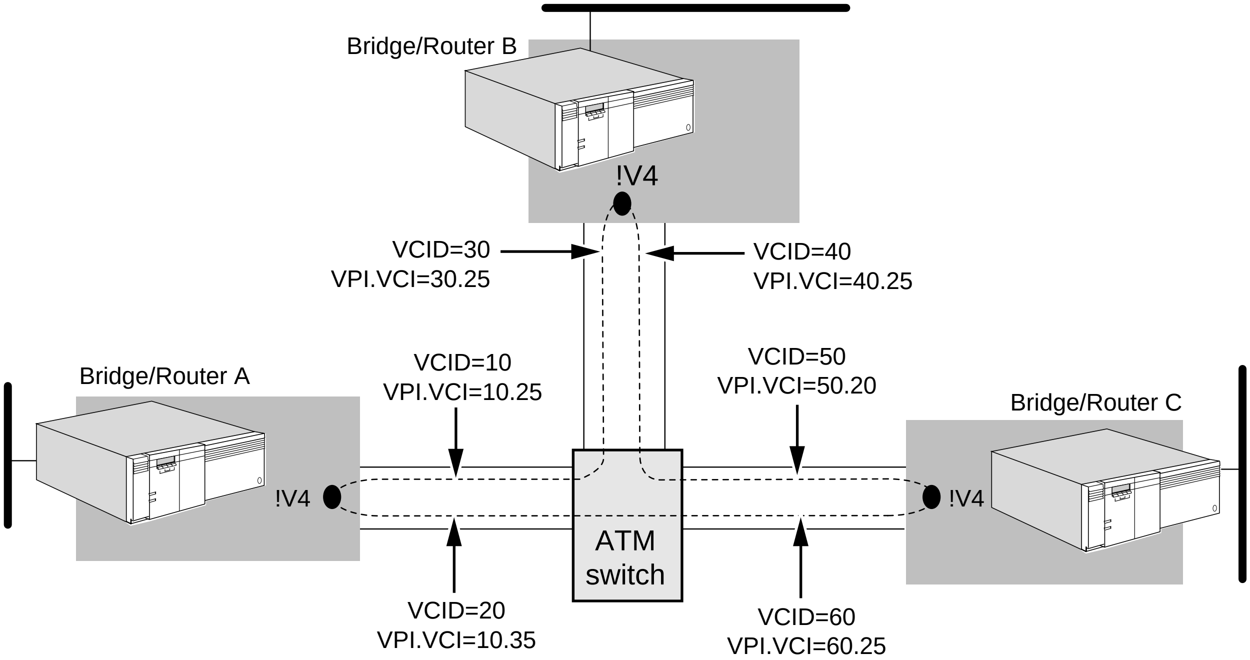


Figure 47-1 Configuring the ATM Service

1 Verify that the port owner setting is ATM by entering:

```
SHow -PORT OWNer
```

The NETBuilder II bridge/router automatically sets the port owner to ATM if the MP ATMLink module is installed. If the setting for the port is not correct, use the SETDefault command.

For example, to set the owner on port 4 to ATM on bridge/router A, enter:

```
SETDefault !4 -PORT OWNer = ATM
```

2 Create a virtual port for each remote network that is attached to the ATM network using:

```
ADD !<port> -PORT VirtualPort <path> MPATM
```

For example, to configure a virtual port for path 4 on bridge/router A, enter:

```
ADD !V4 -PORT VirtualPort 4 MPATM
```

Enter similar commands on bridge/routers B and C.

Each ATM virtual port has a unique media access control (MAC) address, and virtual ports are limited to 64 per physical interface.

3 For outbound traffic, configure a traffic shaper to control the traffic flow using:

```
SETDefault -ATM TrafficShaper = <id>(1–14) <peak>(1–155,000)
  <avg>(1–155,000) [<burst>(1–255)] [High | Low]
```

Based on the user applications, configure the traffic shaper options. For software version 9.0, only AAL5 data-application traffic (not voice and video application traffic) is supported.

a Supply the ID of the shaper to be modified. Valid IDs are from 1 to 14.

b Specify the peak rate and average rate in kilobits per second. Valid rates are from 1 to 155,000.

c Specify the burst count in 53-byte cells. Valid numbers are from 1 to 255. The default burst count is 32.

d Specify the priority level. Valid priorities are High or Low.

e Virtual circuit traffic associated with a high-priority shaper are serviced first. The default priority is High. If several traffic shapers have the same priority, they are serviced in a round-robin process and considered to be equal priority.

For example, to configure shaper 9 with a 10 kbps peak rate, an 8 kbps average rate, a burst count of 64 53-byte cells, and a high priority, enter:

```
SETDefault -ATM TrafficShaper = 9 10 8 64 High
```

For conceptual information about traffic shaping, refer to “Quality of Service” on page 47-13 and to “Traffic Shapers” on page 47-14.

4 Add a permanent virtual circuit for the virtual port, and map its unique virtual circuit identifier (VCID) to the service provider’s VPI.VCI using:

```
ADD !<port> -ATM PermVirCircuit <vcid> <vpi.vci> [LLCSNAP | [NULL
  | IP | IPX]] [<shaper_id>]
```

a Supply a VCID between 1 and 1024; enter the VPI.VCI number supplied by the ATM service provider.

b Supply an encapsulation type. Use LLCSNAP to allow multiple protocol types to be carried within a single ATM virtual circuit. Use NULL and the keyword IP or IPX when only one protocol is configured to run on the virtual circuit.

c Select a traffic shaper ID between 1 and 14 for outgoing traffic that was previously configured in step 3.

For example, to assign VCIDs of 10 and 20 to the VPI.VCIs of 10.25 and 10.35 on virtual port !V4 with LLCSNAP encapsulation using traffic shaper 9 on bridge/router A, enter:

```
ADD !V4 -ATM PermVirCircuit 10 10.25 LLCSNAP 9
ADD !V4 -ATM PermVirCircuit 20 10.35 LLCSNAP 9
```

Enter similar commands on bridge/routers B and C, making sure to use the same encapsulation type.

5 If necessary, adjust the size of the VPI and VCI bits to match the size supported by the ATM switch using:

```
SETDefault !<port> -ATM VPIBits = <vpi_bits>(1–8)
SETDefault !<port> -ATM VCIBits = <vci_bits>(1–16)
```

By default, VPI is set to 6, and VCI is set to 10. Valid VPI numbers range from 0 to 255, valid VCI numbers range from 0 to 65,535 when the full range of bits is used. VPI.VCIs from 0.0 to 0.32 are reserved virtual circuits and are not allowed as user virtual circuits. The VPI and VCI values must be compatible with the configured value for the VPIbits and VCIbits parameters.

6 If you adjust the VPIBits and VCIBits parameters, re-enable the path using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

Verifying the Configuration

To verify your ATM configuration, following these steps:

1 Display current ATM configuration information by entering:

```
SHow -ATM CONFiguration
```

Verify that your ports and paths, and the PVC are correctly configured.

2 Obtain ATM distributed protocol module (DPM) statistics using:

```
SHow -SYS [!<port | slot>] DpmSTATistics [POrt | SLot] [SRc |
  DEst] [<SUmmary | ALl | BRidge | IP>]
```

This display shows per-slot or per-port statistics for IP or bridge data sent or received on the ATM interface.

3 Obtain virtual ports statistics for IPX by entering:

```
SHow -SYS STATistics -IPX
```

This display shows IPX per port statistics for data sent or received over ATM virtual ports.

For detailed statistic information, refer to Appendix H.

Monitoring the Network

If you are experiencing connectivity problems, monitor the virtual circuit and the network connectivity status by following these steps:

1 Specify the time interval at which the interface is checked to determine whether it is connected to the ATM network using:

```
SETDefault !<port> -ATM KeepAliveTime = <seconds>(1–60)
```

The default setting is 2 seconds.

2 Re-enable the path to make the changes to the KeepAliveTime parameter effective using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

3 Determine if the interface is connected to the ATM network using:

```
SETDefault !<port> -ATM LoopMode = AssumeConnected | DetectFraming
  | LoopBack
```

The DetectFraming option determines whether the interface is connected to the ATM network if successful framing of received data has occurred.

The LoopBack option determines whether the interface is connected to the ATM network if F4 loopbacks to the ATM switch are successful.

4 Obtain end-to-end connection status by performing end-to-end loopback testing for all virtual circuits associated with the specified virtual port using:

```
SETDefault !<port> -ATM VirCirLoopTime = <seconds>(1–60)
SETDefault !<port> -ATM VirCirLoopMode = ENabled
```

By default, the VirCirLoopTime parameter is set to 5 seconds. It specifies the time interval in seconds to initiate the F5 loopback to determine the end-to-end connection status.

Configuring Transparent Bridging

This section describes how to configure transparent bridging over ATM using PVCs.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the transparent bridging instructions in Chapter 3.
- Set up the ATM Service as described in "Setting Up the ATM Service" on page 47-1.
- Obtain the virtual circuit identifier (VCID) of the PVCs for each bridge/router participating in transparent bridging.

Transparent bridging does not correctly operate in some nonmeshed topologies. For more information, refer to "Fully Meshed, Partially Meshed, and Nonmeshed Topologies" on page 47-18.

Procedure

To configure transparent bridging over ATM, see Figure 47-2 and follow these steps:

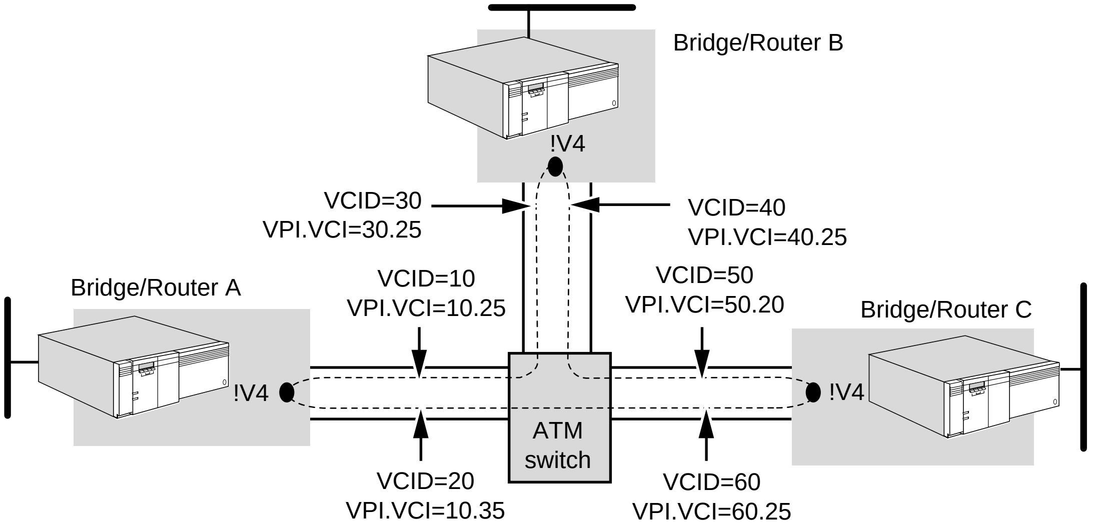


Figure 47-2 Configuring Transparent Bridging over ATM

1 Verify that transparent bridging is enabled on each bridge port that is directly connected to the ATM switch.

By default, transparent bridging is enabled. To verify the setting, on each device use:

```
SHow -BRidge TransparentBRidge
```

If transparent bridging has been disabled, you can enable it on virtual port 4 of bridge/routers A, B, and C. On each of these devices enter:

```
SETDefault !V4 -BRidge TransparentBRidge = TransparentBRidge
```

2 Enable the bridge by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

3 Specify the local VCID of the PVCs connecting to bridge neighbors that are participating in bridging over ATM.

For example, to specify bridge/router A's local VCIDs of the PVCs connecting to bridge/routers B and C and map them to virtual port !V4, on bridge/router A enter:

```
ADD !V4 -BRidge ATMNeighbor = 10
ADD !V4 -BRidge ATMNeighbor = 20
```

Enter similar commands on bridge/routers B and C to configure ATM for their neighbors. You can configure up to 256 neighbors on a virtual port.

This completes the procedure for configuring bridging over an ATM switch.

Configuring Source Route Bridging

This section provides information for configuring source route bridging over ATM.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 5.
- Set up the ATM service as described in "Setting Up the ATM Service" on page 47-1.
- Assign a unique ring number for each remote network.
- Assign a bridge number for the bridge.

Procedure

To configure source route bridging over ATM, follow these steps:

1 Assign each wide area port of each bridge/router that is attached to the ATM network the ring number (hexadecimal) of the network it accesses using:

```
SETDefault !<port> -SR RingNumber = <number> (1-4095) | 0x<number>
  (1-FFF)
```

You can enter the ring number in decimal or hexadecimal format. Precede the hexadecimal number with 0x.

2 Verify that source route bridging is enabled on the wide area port using:

```
SHow !<port> -SR SrcRouBridge
```

If source route bridging is disabled, you need to enable it for your wide area port using:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

3 If you want to run source route and transparent bridging on a NETBuilder II bridge/router, skip this step and go on to step 4. If you want to run source route bridging only on a NETBuilder II bridge/router, disable transparent bridging on the wide area port using:

```
SETDefault !<port> -BRidge TransparentBRidge = NoTransparentBRidge
```

This step does not apply to model 32x and 52x SuperStack II bridge/router. Transparent bridging is not supported on these models.

4 Specify the local VCID of the PVCs connecting to bridge neighbors that are participating in bridging over ATM using:

```
ADD !<port> -BRidge ATMNeighbor = <VCID>
```

This completes the procedure for configuring source route bridging over an ATM switch.

5 Verify that bridging is enabled by entering:

```
SHow -BRidge CONFiguration
```

If bridging has been disabled, enable it for the system by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

Configuring IP Routing

This section describes how to configure IP routing over ATM using PVCs.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your LAN according to the procedures in Chapter 6.
- Set up the ATM Service as described in "Setting Up the ATM Service" on page 47-1.
- Determine the IP addresses for each port of your bridge/router that is attached to the ATM switch.
- Obtain the IP address and the VCID of the PVCs for each bridge/router that is attached to the ATM switch and participating in IP routing.

Procedure

To enable IP to operate over an ATM switch, see Figure 47-3 and follow these steps:

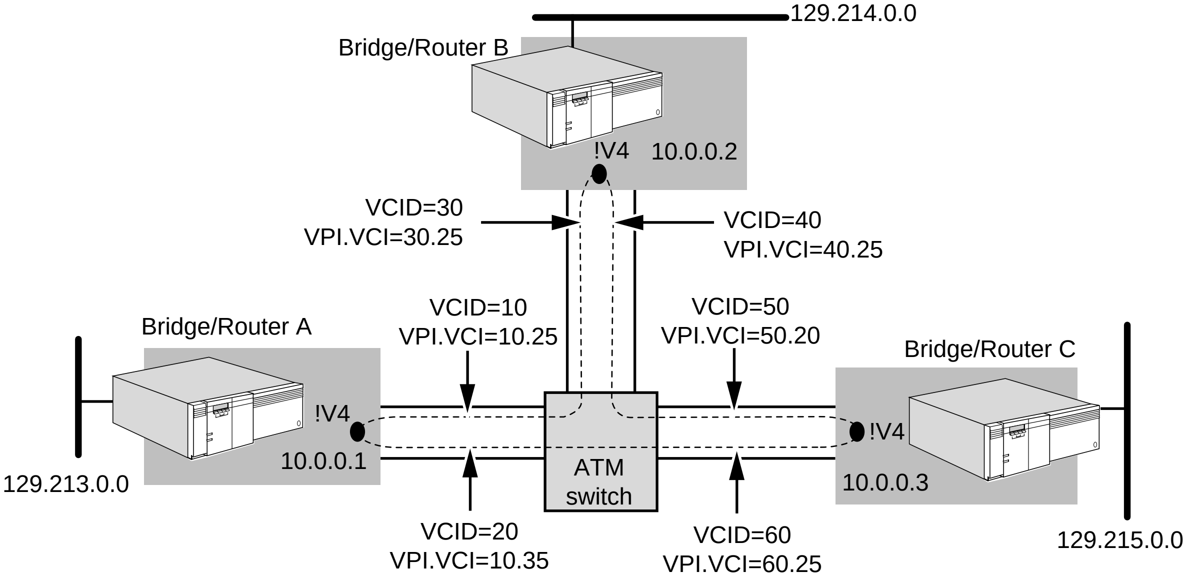


Figure 47-3 Configuring IP over ATM

1 Assign an IP address to each virtual port on each bridge/router that is directly attached to the ATM switch.

For example, the following command assigns the address 10.0.0.1 to virtual port 4 on bridge/router A:

```
SETDefault !V4 -IP NETaddr = 10.0.0.1
```

2 Specify IP-to-ATM address mapping information for all neighbors.

The following sequence of commands specifies IP-to-ATM address mapping information for the bridge/routers directly attached to the ATM switch. In the examples that follow, ATM can be used in place of the and sign (&) when specifying the VCID of the PVC. The VCID has local significance and is mapped to the PVC associated with the neighbor.

For example, on bridge/router A (IP address 10.0.0.1) enter:

```
ADD -IP ADDRess 10.0.0.2 &10
ADD -IP ADDRess 10.0.0.3 &20
```

Enter similar commands on bridge/router B (IP address 10.0.0.2) and bridge/router C (IP address 10.0.0.3), specifying the IP address and the local VCID.

3 Enable the dynamic routing protocols using Routing Information Protocol-Internet Protocol (RIPIP) or Open Shortest Path First (OSPF) for each virtual port.

- To learn routes dynamically on virtual port 4 using RIPIP, determine if the ATM network is fully meshed or partially meshed. If it is fully meshed, then enter:

  ```
  SETDefault !V4 -RIPIP CONTrol = (TAlk, Listen, FullMesh)
  ```

  If it is partially meshed, enter:

  ```
  SETDefault !V4 -RIPIP CONTrol = (TAlk, Listen, NonMesh)
  ```

  Setting the CONTrol parameter to the TAlk and Listen values allows the router to send and receive routing information with other routers using RIP.

The RIPIP Service CONTrol parameter enables or disables RIPIP routing for the specified port. Neighbor learning is enabled by default (DynamicNbr) which causes new addresses to be learned through the Inverse Address Resolution Protocol (InARP) and dynamically updates the RIP AdvToNeighbor list. If NoDynamicNbr is specified, RIP's AdvToNeighbor list is not updated with new addresses and the neighbors list must be manually configured.

- To enable routes dynamically on virtual port 4 using OSPF, determine whether the ATM network is fully meshed or partially meshed.

  If the network is fully meshed, enter:

  ```
  SETDefault !V4 -OSPF CONTrol = (Enable, FullMesh)
  ```

  If the network is partially meshed, enter;

  ```
  SETDefault !V4 -OSPF CONTrol = (Enable, NonMesh)
  ```

  All of the OSPF neighboring routers must be configured with the same mode: FullMesh or NonMesh. NonMesh is the default setting for this parameter.

  After OSPF operation has been enabled, the router exchanges routing information with other routers using OSPF.

The OSPF Service CONTrol parameter enables or disables OSPF routing for the specified port. Neighbor learning is enabled by default on nonbroadcast multi-access (NBMA) interfaces, which means that neighbor lists are automatically created and OSPF operates correctly without static neighbor information. Neighbor learning can be disabled (NoDynamicNbr) for security reasons so that only those statically configured neighbors exchange routing information.

4 If the port is configured with neighbor learning disabled, manually specify neighbors for the routing protocols.

a If your network is running RIP, specify a list of neighbor addresses to which RIP will send update packets.

For example, on bridge/router A, add the IP addresses of neighboring bridge/routers B and C, enter:

```
ADD !V4 -RIPIP AdvToNeighbor 10.0.0.2
ADD !V4 -RIPIP AdvToNeighbor 10.0.0.3
```

Enter similar commands on bridge/routers B and C.

b If your network is running OSPF, specify a list of neighbor addresses to which OSPF will send update packets.

For example, on bridge/router A, add the IP addresses of neighboring bridge/routers B and C, enter:

```
ADD !V4 -OSPF Neighbor 10.0.0.2
ADD !V4 -OSPF Neighbor 10.0.0.3
```

Enter similar commands on bridge/routers B and C.

5 Verify that IP routing is enabled on each bridge/router that is attached to the ATM switch by entering:

```
SHow -IP CONFiguration
```

If IP routing has been disabled, enable it by entering:

```
SETDefault -IP CONTrol = ROute
```

This completes the procedure for configuring IP routing over an ATM switch.

Configuring IPX Routing

This section describes how to configure IPX routing over ATM using PVCs.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Configure your IPX LAN according to the procedures in Chapter 13.
- Set up the ATM Service as described in “Setting Up the ATM Service” on page 47-1.
- Determine the IPX network number to be assigned to each port attached to the ATM network.
- Obtain the MAC addresses for each remote host participating in IPX routing.

Procedure

To enable IPX to operate over an ATM switch, see Figure 47-4 and follow these steps:

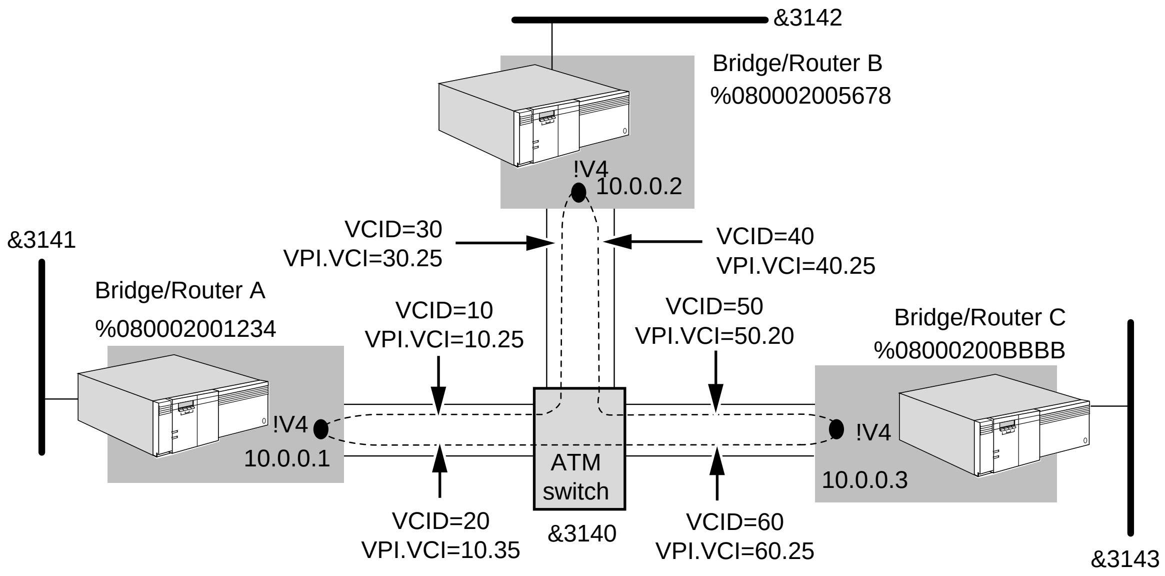


Figure 47-4 Configuring IPX over ATM

1 Assign a network number to each virtual port on each 3Com bridge/router connected to the ATM switch.

For example, assign &3140 as the network number to virtual port 4 on bridge/routers A, B, and C by entering the following command on each bridge/router:

```
SETDefault !V4 -IPX NETnumber = &3140
```

2 Specify IPX network number to ATM VCID mapping information for each bridge/router directly connected to the ATM switch.

For example, to map bridge/router A's local VCID to the neighbor's MAC address, enter:

```
ADD !V4 -IPX ADDRess &10 %080002005678
ADD !V4 -IPX ADDRess &20 %08000200BBBB
```

Enter similar commands on bridge/routers B and C using their local VCIDs and the neighbor's MAC address. To obtain the physical MAC address of neighbors, enter:

```
SHow -SYS ADDRess
```

3 If you have a partially meshed topology and you are operating on a non-NBMA network, specify which neighbors on each interface receive route reachability information.

For example, on bridge/router A, specify that bridge/router B receives route reachability information by entering:

```
ADD !V4 -NRIP AdvToNeighbor &3140%080002005678
ADD !V4 -SAP AdvToNeighbor &3140%080002005678
```

The dynamic neighbor learning feature is the default on ports on NBMA networks, such as X.25 and Frame Relay. This option is not displayed for non-NBMA networks. When dynamic neighbor learning is enabled, the neighbor list is automatically created and NRIP/SAP operates correctly without requiring you to manually configure static neighbor information as shown in the example in this step.

4 Enable the use of policy parameters by entering:

```
SETDefault !V4 -NRIP PolicyControl = AdvToNbr
SETDefault !V4 -SAP PolicyControl = AdvToNbr
```

5 Verify that IPX routing is enabled on each bridge/router that is attached to the ATM switch by entering:

```
SHow -IPX CONFiguration
```

If routing has been disabled on bridge/router A, enable it by entering:

```
SETDefault !V4 -IPX CONTrol = ROute
```

Enable routing on bridge/routers B and C.

In this example, bridge/routers A, B, and C are running software version 9.0 or later.

6 If you are using NRIP and SAP as your routing protocols, verify that routing is enabled on each port of each bridge/router that is attached to the ATM switch by entering:

```
SHow -NRIP CONTrol
```

To verify that Auto, or Talk and Listen are set, enter:

```
SHow -SAP CONTrol
```

7 If you are using NLSP as the routing protocol, follow these steps:

a Make sure the NLSP routing protocol is enabled by entering:

```
SHow -NLSP CONTrol
```

b Specify the local VCID of the PVC that is associated with neighbors that will be taking part in routing over ATM using:

```
ADD !<port> -NLSP Neighbors &<VCID>
```

For example, on bridge/router A enter the local VCIDs of the PVCs:

```
ADD !V4 -NLSP Neighbors &10
ADD !V4 -NLSP Neighbors &20
```

To allow the bridge/routers B and C to accept the adjacency, you must configure the Neighbors parameter on each of them and supply the local VCID.

c Display the NLSP adjacencies by entering:

```
SHow -NLSP ADJacencies
```

This completes the procedure for configuring IPX routing over an ATM switch.

How ATM Works

ATM transmits voice, video, and data across LANs, MANs, and WANs. ATM is an international standard defined by the American National Standards Institute (ANSI) and the International Telecommunications Union–Telecommunications Standards Sector (ITU-TSS), formerly CCITT. ATM is the result of research and the development of the Broadband Integrated Services Digital Network (B-ISDN).

ATM implements a high-speed, connection-oriented, cell-switching and multiplexing technology provides you with bandwidth up to 155 Mbps (3Com's offering). In ATM, all information is formatted into small, fixed-length units called *cells*. Each cell contains 53 octets divided into a 48-octet information field (or payload) and a 5-octet header. By using small fixed-length cells with switching technology, ATM can provide minimal delays for voice and video applications. The switch processes each cell more quickly, and the switch throughput increases. Small cells are not delayed by large cells because all the cells are the same size, which greatly reduces network delays.

ATM operates in a connection-oriented mode. A connection-oriented service requires that a virtual connection be established between the source and destination nodes before data can be transmitted. All connections are virtual in the sense that bandwidth is not permanently assigned to the connection; instead, the network provides the required bandwidth when cells are transmitted. Connections can be established at subscription time as PVCs or on demand as switched virtual circuits (SVCs) using a signaling protocol. In software version 9.0, only PVCs are supported.

Network Interfaces

Software version 9.0 supports the ATM Forum's ATM User-Network Specification, version 3.0 and 3.1. In this specification, two types of interfaces are defined for ATM networks and are shown in Figure 47-5:

- User-to-network interface (UNI)
- Network node-to-network node interface (NNI)

The UNI defines the interface between a user and the network, and includes both private and public interfaces. In Figure 47-5, the private UNI (1) defines the interface between an ATM user device and a private ATM switch owned by a private organization. The public UNI (2) defines the interface between an ATM user device or a private ATM switch and an ATM switch used in a public service provider's network.

The NNI defines a switch-to-switch interface, also known as an inter-switching system interface (ISSI), and includes both private and public interfaces. In Figure 47-5, a private ISSI (3) defines an interface between private ATM switches. A public ISSI (4) defines an interface between public switches. The NNI does not include the interface between a private switch and a public switch, which is considered part of the public UNI.

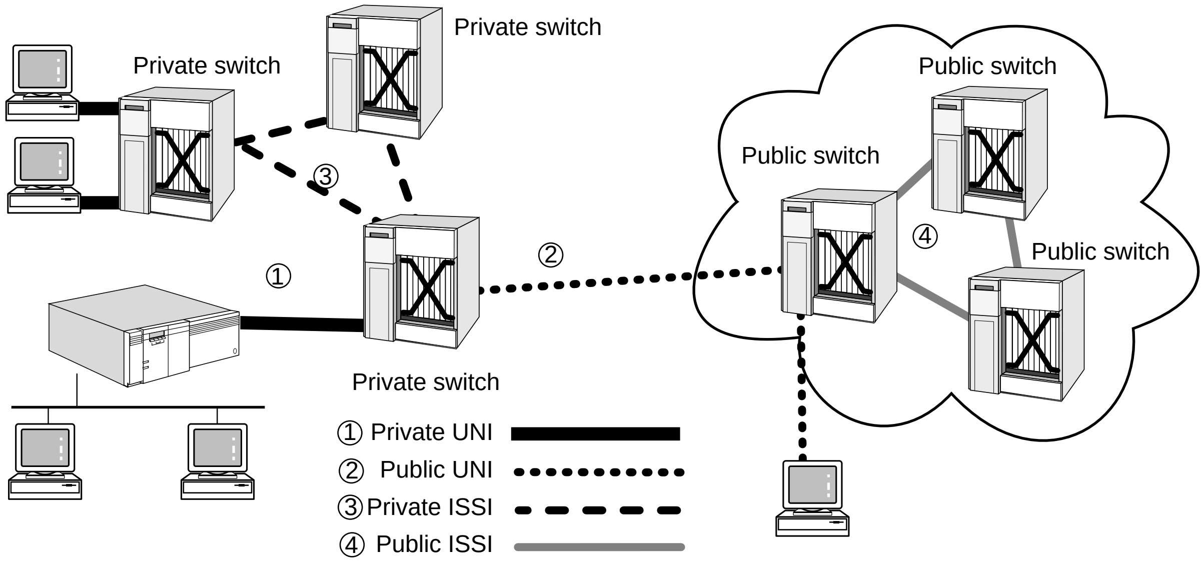


Figure 47-5 ATM Network Interfaces

ATM Addressing, Virtual Paths, and Virtual Channels

The header of each ATM cell contains addressing information like traditional LAN packets. Instead of a specific destination address, each cell contains two fields, an 8-bit VPI and a 16-bit VCI, that specify the PVC over which the cell should be forwarded. The VPI and VCI fields define a routing field that provides an ATM switch with the information that it needs to route each cell. The PVC is usually represented in VPI.VCI format, where VPI is a decimal number between 0 and 255 and VCI is a decimal number between 0 and 65,535.

A virtual channel (VC) is a communications circuit that transports ATM cells between two or more endpoints. The endpoints of a VC can be a user-to-user connection, a user-to-network connection, or a network-to-network connection. When multiple VCs on the same transmission path are headed for the same destination, they can be grouped into a virtual path, which is a collection of VCs. A VP performs the same functions as a trunk line in a telephone network; the VP allows a number of virtual channels to be bundled together for transport between two ATM devices. Figure 47-6 shows the relationship between virtual channels and virtual paths.

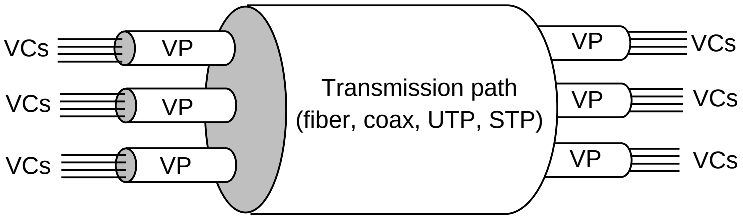


Figure 47-6 Virtual Channels and Virtual Paths

When configuring your bridge/router for ATM, to configure a PVC use:

```
ADD !<port> -ATM PermVirCircuit <vcid> <vpi.vci> [LLCSNAP | [NULL
  | IP | IPX]] [<shaper_id>]
```

Encapsulation Types

In software version 9.0, multiprotocol encapsulation over ATM AAL5 (MPATM) is supported using PVCs as defined in RFC 1483. The following encapsulation formats are supported for transparent bridging, and IP and IPX routing:

- MPATM logical link control/Subnetwork Access Protocol (LLC/SNAP)

  Use LLC/SNAP encapsulation to allow multiple protocol types to be carried within a single ATM connection (virtual circuit). The type of the encapsulated packet is indicated by a standard LLC/SNAP header.

- NULL encapsulation

  Use NULL encapsulation when only one protocol is configured to run on a VC. In this situation, no encapsulation is required. This type of encapsulation is not supported with transparent bridging.

For detailed descriptions of the encapsulation formats, refer to RFC 1483.

Quality of Service

Different types of applications require different levels of service from a network. For example, voice and video applications are very sensitive to delay and variations in delay, but are not insensitive to minimal cell loss. Data applications are not insensitive to delay or variation in delay, but extremely sensitive to cell loss.

To meet the specific service requirements of each application, the node requesting the connection informs the network about the desired characteristics of each connection request. Some of the information in a connection request includes the following:

- Called party number
- Average bandwidth requirements
- Peak bandwidth requirements
- Maximum acceptable percentage of cell loss
- Maximum acceptable variation in network delay

The network uses this information to select the individual physical links that support the virtual circuit across the network as shown in Figure 47-7. For example, when selecting a specific physical link, the network makes sure that it can support all virtual circuits assigned to the physical link and still maintain the quality of service requirements for each individual virtual circuit. When the network and user agree on the characteristics of the connection, the network establishes the virtual circuit across the network. If the network cannot support the desired quality of service for a connection request, it rejects the connection.

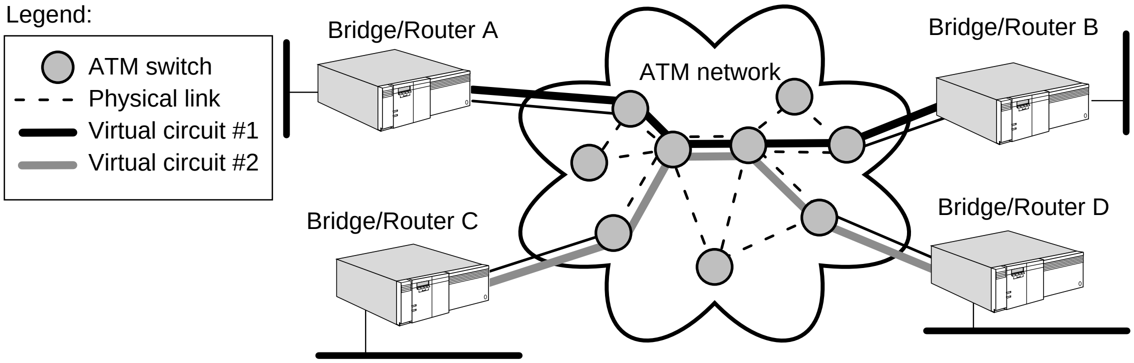


Figure 47-7 Physical Links and Virtual Circuits

After the connection is established, the nodes at each end of the connection exchange information by transmitting cells across the UNI. The cells are relayed from switch to switch until they arrive at the UNI of the destination node. When there is no more data to be transmitted, the connection is terminated and the previously allocated network resources can be used by other connections.

Traffic Shapers

A traffic shaper defines the attributes that allow the outbound traffic of attached virtual circuits to be transmitted based on the following items:

- Priority level
- Average and peak rate in kilobits per second
- Burst count

The peak rate specifies the maximum data rate at which a virtual circuit can transmit, which determines the maximum bandwidth available to all of the virtual circuits attached to the traffic shaper. You configure traffic-shaping attributes using the -ATM TrafficShaper parameter. You must associate every virtual circuit with one traffic shaper using the -ATM PermVirCircuit parameter.

A traffic shaper activates only when one or more of the attached virtual circuit connections becomes active. Each active traffic shaper consumes a fixed portion of the total bandwidth available on the associated ATM interface, as specified by the peak rate, regardless of the number of VCs that are attached to the traffic shaper.

The combined peak rates of all active traffic shapers should not exceed the maximum bandwidth available on the ATM interface. If the maximum bandwidth is exceeded, some traffic shapers and the associated virtual circuit traffic are not serviced because of the limitation in available bandwidth. For example, suppose you configure three active traffic shapers with the same priority and a peak rate of 75 Mbps and all of the attached virtual circuits are transmitting. The VCs attached to one of the traffic shapers will not be adequately serviced because the traffic shaper is selected one at a time in a round-robin process until the maximum bandwidth of 155 Mbps is reached.

The software displays the following message if the total peak rate for all active shapers exceeds the maximum bandwidth:

```
WARNING: ATM traffic shapers configured for !<path> exceeds
  155Mbps.
```

The software services traffic shapers configured with a high priority ahead of the shaper with low priority. If VCs attached to the high-priority shapers use up the available bandwidth, the VCs associated with the low-priority shapers are not serviced.

The software provides 14 traffic shapers with predefined initial values. You can reconfigure each traffic shaper to meet the traffic control requirements of the attached virtual circuits. The new traffic-shaping attributes do not take effect for the attached VCs until the associated ATM interface is reset (the path must be re-enabled). To display the predefined initial values of the traffic-shaping attributes, enter:

```
SHow -ATM TrafficShaper
```

Each of the 14 traffic shapers has a peak bit rate, average bit rate, burst cell rate, and a priority. Each virtual channel connection (VCC) present on the module must be mapped to a shaper for it to effectively carry data. When more than one VCC is mapped to a shaper, each VCC has the bandwidth defined by the shaper. The aggregate bandwidth of all the VCCs mapped to all the active shapers should not exceed the total bandwidth of the link. Shapers available on the ATMLink module provide the following features:

- Outbound data traffic control
- Bandwidth reservation
- Prioritization of traffic among VCCs of the same or different protocols

Examples of these features are shown in the following pages. For all examples, a maximum bandwidth of 50 Mbps full duplex is assumed.

Outbound Data Traffic Control

Where data is known to be of a variable rate and bursty in nature, the traffic shapers moderate and limit the traffic rate to a predefined shaper value. The following example illustrates outbound data traffic control.

Example To limit IP traffic going from router A to router B to a peak rate of 15 Mbps, an average rate of 10 Mbps, and a maximum number of back-to-back cells at the peak rate to 32 cells, follow these steps:

1 Define the shaper by entering:

```
SETDefault -ATM TrafficShaper = 3 15 10 32 H
```

2 Define the PVC and map it to the shaper:

```
ADD !V1 -ATM PermVirCircuit 1 10.20 null IP 3
```

Bandwidth Reservation

You can use bandwidth reservation where there are multiple protocols running and when bandwidth must be reserved for some protocols in a predetermined ratio.

Example Suppose IP and IPX protocols are running on the same UNI interface, and you want to reserve 35 Mbps for IP and 15 Mbps for IPX. Follow these steps:

1 Define a shaper for 30 Mbps average and peak rate by entering:

```
SETDefault -ATM TrafficShaper = 3 35 35 32 H
SETDefault -ATM TrafficShaper = 4 15 15 32 H
```

2 Define a second shaper for 25 Mbps average and peak rate by entering:

```
ADD !V1 -ATM PermVirCircuit 1 10.20 null IP  3
ADD !V1 -ATM PermVirCircuit 2 10.21 null IPX 4
```

Prioritization of Traffic among VCCs of the Same Protocol

When there are multiple VCCs for a given protocol, you can use prioritization between VCCs.

Example Suppose there are two VCCs defined to carry IP traffic, but you want the traffic on one VCC to be higher than the traffic on the other VCC. Follow these steps:

1 Define a set of traffic shaping attributes associated with each PVC by entering:

```
SETDefault -ATM TrafficShaper = 3 30 30 32 H
SETDefault -ATM TrafficShaper = 4 30 30 32 L
```

2 Add the PVCs on the virtual ports by entering:

```
ADD !V1 -ATM PermVirCircuit 1 10.20 null IP   3
ADD !V2 -ATM PermVirCircuit 2 10.21 null IP   4
```

Prioritization of Traffic among VCCs of Different Protocols

When there are multiple protocols, you can prioritize one protocol over the other.

Example Suppose there are two VCCs defined to carry IP and IPX traffic, but you want IP traffic to be a higher priority than IPX. Follow these steps:

1 Define a set of traffic-shaping attributes associated with each PVC by entering:

```
SETDefault -ATM TrafficShaper = 3 30 30 32 H
SETDefault -ATM TrafficShaper = 4 30 30 32 L
```

2 Add the PVCs on the virtual ports by entering:

```
ADD !V1 -ATM pvc 1 10.20 null IP   3
ADD !V2 -ATM pvc 2 10.21 null IPX  4
```

Network Management

When you connect to an ATM network using an ATM adapter on the user side to an ATM switch on the network side, your user-to-network connection is managed by an ATM UNI Management Entity (UME).

The UMEs exist on both sides of the interface and support an exchange of management information between them. UMEs are used in any device that transmits data in ATM cells across an ATM public or private UNI as shown in Figure 47-8. Typical devices containing UMEs include workstations, bridges, routers, Frame Relay switches, and ATM network switches.

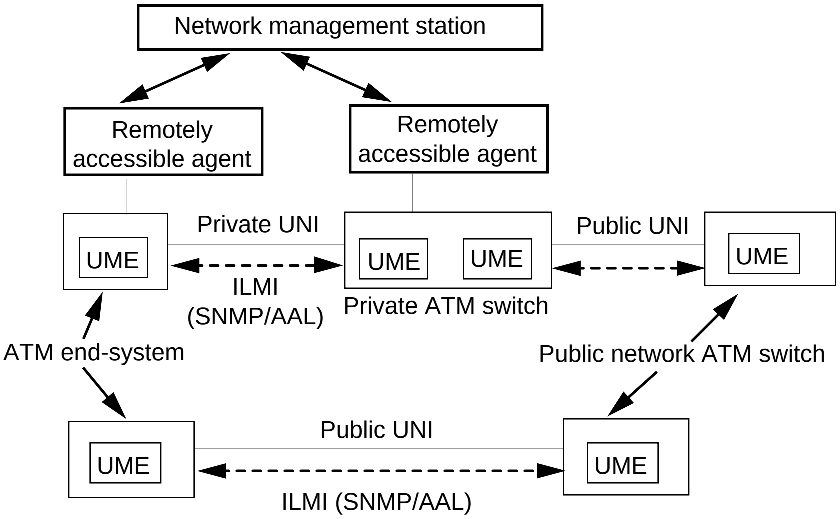


Figure 47-8 Interim Local Management Interface Definition

The two UMEs (one on each side of the UNI) have the same management information base (MIB) defined as the ATM UNI Interim Local Management Interface (ILMI) MIB by the UNI specification, and support seven groups of management information with respect to the user-to-network interface.

UMEs communicate using the ILMI Protocol, which uses SNMP version 1 PDUs encapsulated in AAL5. The ILMI provides status, configuration, and control information about the virtual path and virtual channel connections available at the UNI. You can obtain statistics about the status and operation of the UNI to facilitate performance monitoring and troubleshooting. By default, all ILMI communication takes place over the VCC with VPI = 0 and VCI = 16.

The key functions of the UME in the bridge/router 9.0 software are as follows:

- Provides the SNMP agent on the NETBuilder II bridge/router access to all supported objects on the ATM UNI ILMI MIB groups (except for the network prefix group).

  Access by the agent to the ATM UNI ILMI MIB on the switch is not supported. Access to other MIBs on the NETBuilder II bridge/router through the ILMI from the switch is also not supported.

- Provides a management station on the switch side access to all objects of the ATM UNI ILMI MIB as well as the “system” group.

Fully Meshed, Partially Meshed, and Nonmeshed Topologies

A fully meshed ATM topology (Figure 47-9) is a topology in which each node on a network is directly connected to all other nodes on the network. Each node is connected to the other nodes through a virtual circuit.

The topology in Figure 47-9 consists of NETBuilder II bridge/routers. Using virtual circuits, bridge/router A is connected to bridge/routers B, C, and D; bridge/router B is connected to bridge/routers A, C, and D; and so on. This type of topology can provide basic connectivity for campus backbones at 155 Mbps and also can construct sophisticated router clusters around one or more ATM switches.

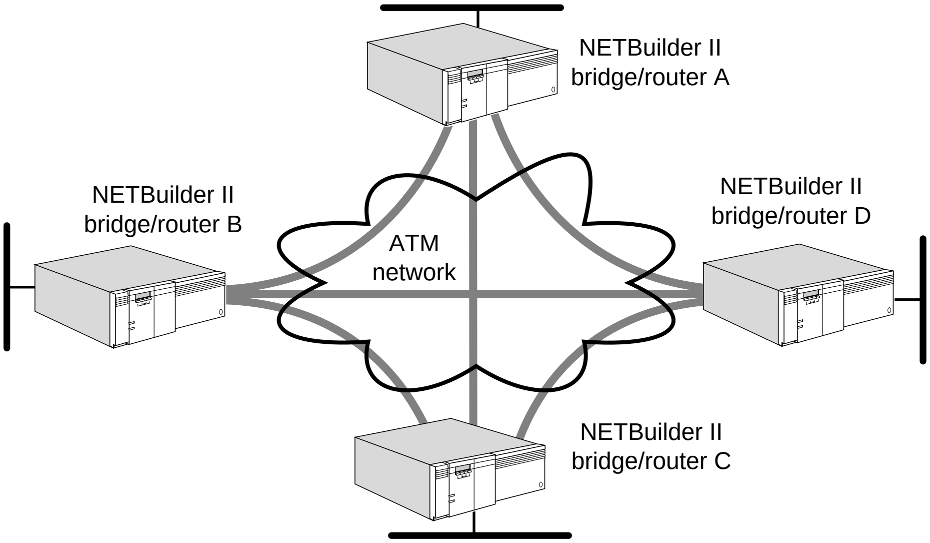


Figure 47-9 Fully Meshed ATM Topology

A nonmeshed ATM topology (Figure 47-10) is a topology where each node on a network may not be connected to all other nodes on the network.

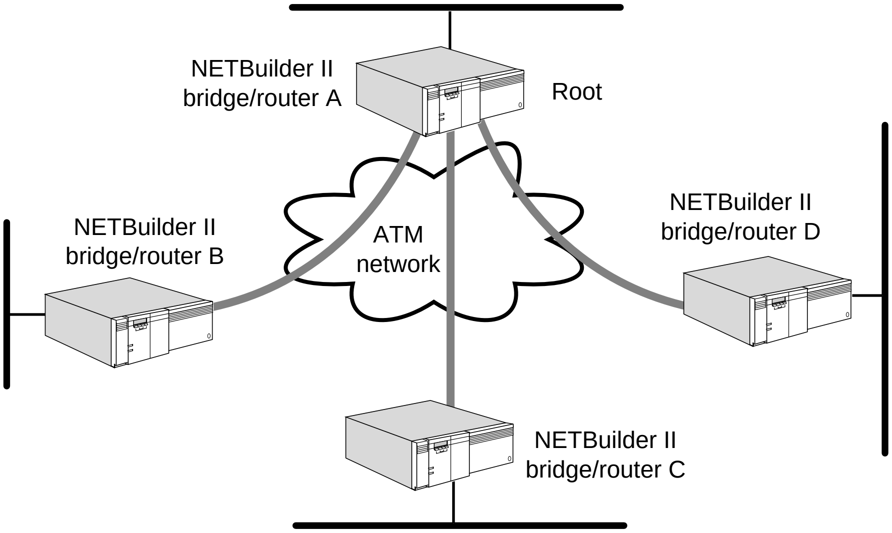


Figure 47-10 Nonmeshed ATM Topology

The topology in Figure 47-10 consists of NETBuilder II bridge/routers. Through the established PVCs, bridge/router A is connected to bridge/routers B, C, and D. bridge/routers B, C, and D are connected to bridge/router A only, but not to one another.

Nonmeshed topologies are supported but not recommended for use with ATM. Because each router is not connected to all other routers, traffic may have to cross the ATM switch twice. In Figure 47-10, traffic from bridge/router B to bridge/router C must pass through the ATM switch to bridge/router A, which sends the traffic through the ATM switch again to bridge/router C. Because the traffic passes through the switch twice, the nonmeshed topology reduces the effectiveness of a high-speed ATM campus backbone.

Transparent bridging does not correctly operate in some nonmeshed topologies. For example, in Figure 47-11, the transparent bridge properly forwards traffic received on !v1 to !v2. However, traffic received from one of its remote connections on !v3 is not properly forwarded to the other two remote connections on !v3; therefore, do not configure transparent bridging in this type of nonmeshed topology. The flooding algorithm floods packets on a per-port basis, not on a neighbor-per-port basis.

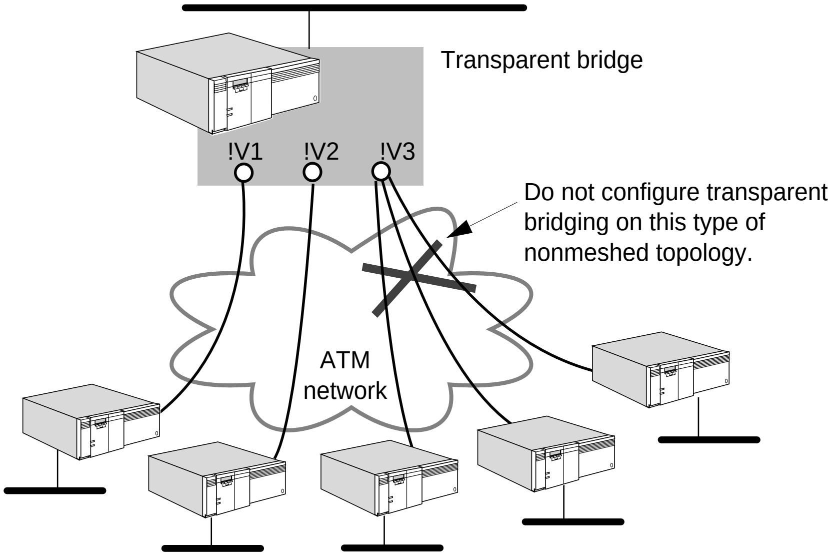

Transparent bridge
!V1
!V2
!V3
Do not configure transparent bridging on this type of nonmeshed topology.
ATM network

Figure 47-11 Transparent Bridging in Nonmeshed ATM Topologies

A partially meshed ATM topology is a topology where some nodes on a network are directly connected to nodes on the network (as in a fully meshed topology) and other nodes are not directly connected (as in a nonmeshed topology). Figure 47-12 is an example of a partially meshed ATM topology.

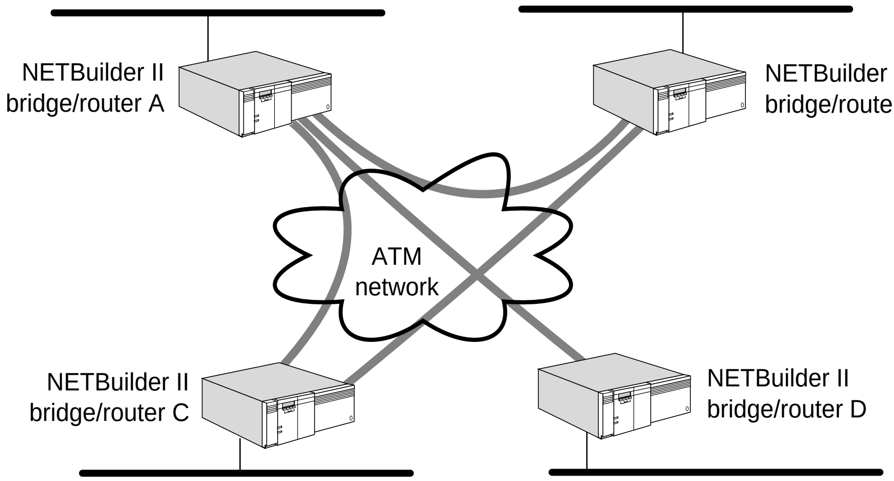


Figure 47-12 Partially Meshed ATM Topology

The topology in Figure 47-12 consists of four NETBuilder II bridge/routers. Through the established PVCs, bridge/routers A, B, and C are connected to one another but bridge/router D is connected to bridge/router A only.

The lack of connectivity among bridge/routers B, C, and D in partially meshed and nonmeshed topologies can be worked around using next-hop split horizon and virtual ports. If you are routing IP-RIP or IPX, these protocols offer the next-hop split horizon feature. In IP-RIP, set -RIPIP CONTrol to NonMesh to enable next-hop split horizon. In IPX, next-hop split horizon is enabled by manually configuring neighbors.

For example, if you are routing IP-RIP and you set -RIPIP CONTrol to NonMesh, a list of neighbors containing bridge/routers B, C, and D will be generated by the system, or you can configure them as neighbors using the -RIPIP AdvToNeighbor parameter. For more information about these parameters, refer to Chapter 47 in *Reference for NETBuilder Family Software*.

If you are routing IPX, you can configure bridge/routers B, C, and D as neighbors using the -NRIP PolicyControl and -NRIP AdvToNeighbor parameters. For more information on next-hop split horizon, refer to Chapter 6 and Chapter 13.

Virtual ports are supported by bridging and all routing protocols, and must be used when configuring ATM for fully meshed, partially meshed, and nonmeshed topologies. For information on the number of virtual ports supported per platform, refer to Table 1-1 on page 1-4.

ATM Terms

The following terms are used in this chapter to explain ATM:

Asynchronous Transfer Mode (ATM)	A transmission protocol that segments user traffic into small, fixed sized cells. Cells are transmitted to their destination where the original traffic is reassembled.
ATM Adaptation Layer (AAL)	Layer 3 of the ATM architecture that adapts user traffic into or from ATM 48-byte payloads.

AAL5 supports variable bit rate, delay tolerant, connection-oriented data traffic requiring minimal sequencing or error detection support.

Interim Local Management Interface (ILMI)
Refers to ATM Forum-defined interim specifications for network management functions between an end user and a public or private network, and between a public network and a private network. It is based on a limited subset of SNMP capabilities.

permanent virtual circuit (PVC)
A virtual channel connection that has been established by manual or semi-automated methods. It is similar to a leased or dedicated real circuit.

switched virtual circuit (SVC)
A virtual channel connection that has been dynamically established in response to a signaling request message.

UNI Management Entity (UME)
The code residing in ATM devices at each end of a UNI circuit that implements the management interface to the ATM network.

user-to-network interface (UNI)
ATM Forum-developed specifications for the procedures and protocols between a user DTE and the ATM network to effectively use ATM services and capabilities.

virtual channel connection (VCC)
Virtual channels in two or more sequential physical circuits concatenated to create an end-to-end connection. A VCC is a specific instance of a switched virtual circuit (SVC) or a permanent virtual circuit (PVC).

virtual channel identifier (VCI)
The 16-bit number in an ATM cell header identifying the specific virtual channel on which the cell is traversing on the current physical circuit.

virtual circuit identifier (VCID)
A user-assigned identifier or alias for a PVC representing the circuit characteristics. The VPI.VCI is analogous to the DLCI of a Frame Relay PVC.

virtual path identifier (VPI)
The 8-bit number in an ATM UNI cell header identifying the specific virtual path on which the cell is traversing on the current physical circuit.

48 CONFIGURING INTERNETWORKING USING ATM AND LAN EMULATION

This chapter describes how to configure a NETBuilder II bridge/router to establish LAN, WAN, and MAN connectivity through Asynchronous Transfer Mode (ATM) with LAN emulation.

For conceptual information, refer to "How ATM and LAN Emulation Work" on page 48-5.

Setting Up the ATMLE Service

This section describes how to configure your bridge/router to transmit and receive data over an ATM interface using LAN emulation.

For detailed descriptions of all commands, refer to *Reference for NETBuilder Family Software*.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Configure your bridge/router ports, virtual ports, and paths according to Chapter 1.
- Make sure the ATM switch is connected and signalling is present on the interface.

Procedure

To perform LAN Emulation over an ATM network, see Figure 48-1 and follow these steps:

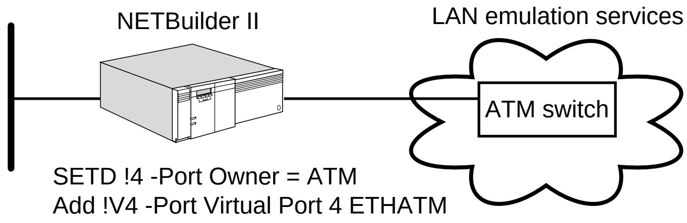

NETBuilder II
LAN emulation services
ATM switch
SETD !4 -Port Owner = ATM
Add !V4 -Port Virtual Port 4 ETHATM

Figure 48-1 Enabling a Port for LAN Emulation

1 To verify that the port owner setting is ATM, enter:

```
SHow -PORT OWNer
```

The NETBuilder II bridge/router automatically sets the port owner to ATM if the ATMLink module is installed. If the setting for the port is not correct, use the SETDefault command.

For example, to set the owner on port 4 to ATM on bridge/router A, enter:

```
SETDefault !4 -PORT OWNer = ATM
```

2 Create a virtual port for a LAN emulation client (LEC) to be attached to the ATM network using:

```
ADD !<port> -PORT VirtualPort <path> ETHATM
```

For example, to configure a virtual port for path 4 on bridge/router A, enter:

```
ADD !V4 -PORT VirtualPort 4 ETHATM
```

Each ATM virtual port has a unique MAC address, and virtual ports are limited to 75 per physical interface.

When the virtual port is added to the configuration, the ATM address for the virtual port is constructed using the MAC address as the ATM address end-system identifier.

Verifying the Configuration

To verify your ATM LAN emulation configuration, display current ATM configuration information by entering:

```
SHowDefault !V4 -ATMLE CONFiguration
```

Verify that your ATM LAN emulation configuration parameters are configured correctly.

Monitoring ATM LAN Emulation Status

To monitor the status of the LEC, enter:

```
SHow -ATMLE STATUS Verbose
```

A display appears showing the state of the LEC. The following example display shows the LEC status during initialization:

```
LAN Emulation Client !V1
LEC State - CONFIG
                The LEC has established the Configure Direct VCC with
                the LECS. The LEC has sent a Configuration Control Frame
                and is waiting for the Configuration Response Frame.
LECID - 0
Emulated LAN - *none*
Local ATM Address:- 47000000003C0000A000000008000206902400
Local MAC Address - %0800020690242
LECS -       Configuration Direct VCC
              CN - 3 VPI - 0 VCI - 89
              ATM Address - 4700790000000000000000000000A03E000000100
LES -        Control Direct VCC
              CN - 0 VPI - 0 VCI - 0
              ATM Address - *none*
             - Control Distribute VCC
              CN - 0 VPI - 0 VCI - 0
              NC - 0 VPI - 0 VCI - 0
              ATM Address - *none*
BUS          -Multicast Send VCC
              CN - 0 VPI - 0 VCI - 0
              ATM Address - * none*
             - Multicast Forward VCC
              CN - 0 VPI - 0 VCI - 0
              ATM Address - *none*
```

The following display shows the state of LEC configuration when initialization is complete:

```
LAN Emulation Client !V1
LEC State - ACTIVE
       The LEC is operational.
LECID - 22
Emulated LAN - elan791_0
Local ATM Address:- 47000000003C0000A000000008000206902400
Local MAC Address - %0800020690242
LECS - Configuration Direct VCC
     CN - 0 VPI - 0 VCI - 0
     ATM Address - *none*
LES - Control Direct VCC
     CN - 4 VPI - 0 VCI - 90
     ATM Address - 470000000003C0000A000000000C0DA60031700
    - Control Distribute VCC
     CN - 5 VPI - 0 VCI - 66
     NC - 0 VPI - 0 VCI - 0
     ATM Address - 470000000003C0000A000000000C0DA60031700
BUS - Multicast Send VCC
     CN - 6 VPI - 0 VCI - 91
     ATM Address - 470000000003C0000A000000000C0DA60031700
    - Multicast Forward VCC
     CN - 7 VPI - 0 VCI - 67
     ATM Address - 470000000003C0000A000000000C0DA60031700
```

Field Descriptions

The ATMLE status screen contains the following fields:

LAN Emulation Client. Identifies the virtual port upon which the ATM LAN emulation is configured.

LEC State. The current operating state of the LEC. The following LEC State values may be displayed:

IDLE The LEC is not operational and is waiting for an event to start the LECS Connect Phase.

SETUP_L1 The LEC is requesting UME to retrieve all the LAN emulation configuration server (LECS) ATM addresses.

SETUP_L2 The LEC is connecting to the LECS ATM address returned from the UME.

SETUP_L3 The LEC is connecting to the "well-known" LECS ATM address.

SETUP_L4 The LEC is connecting to the LECS PVC (VPI=0, VCI=17).

CONFIG The LEC has established the Configure Direct VCC with the LECS. The LECS has sent a configuration control frame and is waiting for the configuration response frame.

JOIN_1 The LEC has received the configuration response frame from the LECS. The LEC is connecting to the LAN emulation server ATM address.

JOIN_2 The LEC has established the control direct VCC with the LES. The LEC has sent a join control frame to the LES and is waiting for the join response frame.

SETUP_B1 The LEC has received the join response frame from the LES. The LEC has sent a LE_ARP Request Frame for the Broadcast and Unknown Server (BUS) ATM address.

SETUP_B2 The LEC has received the LE_ARP reply frame from the LES. The LEC is connecting to the BUS ATM address.

ACTIVE The LEC is operational

LECID. The identification number of the LEC.

Emulated LAN. The name of the emulated LAN successfully joined by the LEC.

LECS. The LAN emulation configuration server connection information.

Configuration Direct VCC is a bidirectional point-to-point virtual channel connection set up by the LEC to the LECS.

LES. The LAN emulation server information. Possible states include the following:

Control Direct VCC is a bidirectional point-to-point VCC set up by the LEC to the LES.

Control Distribute VCC is a unidirectional VCC set up from the LES back to the LEC. This is a point-to-multipoint connection.

BUS. The Broadcast and Unknown Server connection information. Possible states include the following:

Multicast Send VCC is a bidirectional point-to-point VCC set up by the LEC to the BUS.

Multicast Forward VCC is a unidirectional VCC set up from the BUS back to the LEC. This is a point-to-multipoint connection.

VCI . A 16-bit virtual channel identifier identifying the specific virtual channel on which the cell is traversing on the current physical circuit.

VPI . An 8-bit virtual path identifier identifying the specific virtual path on which the cell is traversing on the current physical circuit.

CN. Connection number used to identify a specific SVC.

Controlling Initialization

During initialization the LEC can either rely on the ATM switch unit management entity (UME) to determine the ATM address of the LEC's LAN emulation configuration server or configure the ATM address of a specific LECS.

The LECSATMAddr parameter specifies the ATM address of the LECS. When the LEC is in "manual" mode, and the LECSATMAddr parameter is configured, the LEC uses the configured ATM address to connect to the specified LECS. When the LEC is in "automatic" mode, it uses the UME to retrieve the LECS ATM address that will be used during initialization.

To specify which LECS address to use during initialization, follow these steps:

1 Specify the ATM address of the LECS to be used during initialization using:

```
SETDefault !<vport> -ATMLE LECSAddr <atm address>
```

For example, to assign the LECS with the ATM address `47007900000000000000000000A03E000000100` as the LECS to be used during initialization, enter:

```
SETDefault !v4 -ATMLE LECSAddr 47007900000000000000000000A03E000000100
```

2 Set the LEC to manual mode using:

```
SETDefault !<vport> -ATMLE CONTrol = [MANual | AUTOmatic]
```

For example, to enable manual mode on the LEC, enter:

```
SETDefault !v4 -ATMLE CONTrol = MANual
```

3 Reset the NETBuilder II bridge/router.

The LECS specified by the LECSATMAddr parameter will be used during initialization.

How ATM and LAN Emulation Work

ATM transmits voice, video, and data across LANs, MANs, and WANs. ATM is an international standard defined by the American National Standards Institute (ANSI) and the International Telecommunications Union–Telecommunications Standards Sector (ITU-TSS), formerly CCITT. ATM is the result of research and the development of the Broadband Integrated Services Digital Network (B-ISDN).

ATM implements a high-speed, connection-oriented, cell-switching, and multiplexing technology that provides bandwidth up to 155 Mbps (3Com's offering). In ATM, all information is formatted into small, fixed-length units called cells. Each cell contains 53 octets divided into a 48-octet information field (or payload) and a 5-octet header. By using small fixed-length cells with switching technology, ATM can provide minimal delays for voice and video applications. The switch processes each cell more quickly, and the switch throughput increases. Small cells are not delayed by large cells because all the cells are the same size, which greatly reduces network delays.

ATM operates in a connection-oriented mode. A connection-oriented service requires that a virtual connection be established between the source and destination nodes before data can be transmitted. All connections are virtual in the sense that bandwidth is not permanently assigned to the connection; instead, the network provides the required bandwidth when cells are transmitted. Connections can be established at subscription time as permanent virtual circuits (PVCs) or on demand as switched virtual circuits (SVCs) using a signaling protocol.

Network Interfaces

Software version 9.1 supports the ATM Forum's ATM LAN Emulation User Network Specification version 1.0.

The interface for interoperability with legacy LANs and protocols is the LAN emulation user network interface (LUNI) shown in Figure 48-2. The LUNI protocols allow ATM-attached end systems and LAN/ATM conversion devices to control the virtual connections required for transmission and to emulate the connectionless nature of a LAN or LAN emulation.

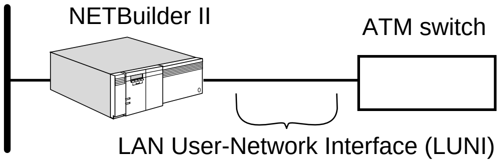


Figure 48-2 LAN Emulation User Network Interface (LUNI)

The main objective of the LAN emulation specification is to enable existing applications to access an ATM network through protocol stacks such as APPN, NetBIOS, and IPX as if they were running over traditional LANs.

LAN emulation works at the media access control (MAC) layer and enables legacy Ethernet, token ring, or FDDI traffic to run over ATM with no modifications to applications network operating systems, or desktop adapters. Legacy end stations can use LAN emulation to connect to other legacy systems as well as to ATM-attached servers, routers, hubs, and other networking devices.

ATM Addressing

The header of each ATM cell contains addressing information like traditional LAN packets. Instead of a specific destination address, each cell contains two fields, an 8-bit VPI and a 16-bit VCI, that specify the PVC or SVC over which the cell should be forwarded. The VPI and VCI fields define a routing field that provides an ATM switch with the information that it needs to route each cell. The PVC or SVC is usually represented in VPI.VCI format, where VPI is a decimal number between 0 and 255 and VCI is a decimal number between 0 and 65,535.

LAN Emulation

LAN emulation is a method for carrying network layer packets across an ATM network. The function of the LAN emulation protocol is to emulate LAN while transporting the packets over an ATM network. The LAN emulation protocol defines the service interface for higher layer network protocols. This interface presents an identical appearance to the existing LANs, and data sent across the ATM network is encapsulated in appropriate LAN MAC packet format. The MAC protocol of the specific LAN is not emulated, whether the MAC protocol is either token passing for 802.5 network types or CSMA/CD for Ethernet types.

LUNI Components and Connections

An emulated LAN on an ATM network consists of the elements shown in Figure 48-3.

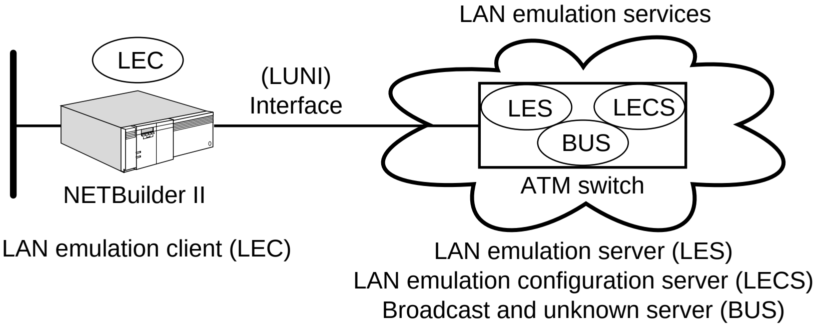

LAN emulation services
LEC
(LUNI)
Interface
LES
LECS
BUS
ATM switch
NETBuilder II
LAN emulation client (LEC)
LAN emulation server (LES)
LAN emulation configuration server (LECS)
Broadcast and unknown server (BUS)

Figure 48-3 LAN Emulation Entities

LAN Emulation Client

The LEC is a process in the NETBuilder II software that operates as an end system. The LEC forwards data, resolves addresses, and performs control functions for a single end-system. A LEC also provides a standard LAN service interface to any higher layer process that interfaces to the LEC.

Each LEC is identified by a unique ATM address, and is associated with one or more MAC addresses reachable through that ATM address.

LAN Emulation Configuration Server

The LECS is a process that assigns individual LAN emulation clients to particular emulated LANs by directing them to the LES that corresponds to the ELAN. There is logically one LECS per administrative domain, which serves all ELANS within that domain.

LAN Emulation Server

The LES provides the control functions for a particular emulated LAN. There is only one logical LES per emulated LAN, and to belong to a particular emulated LAN means to have a control relationship with that emulated LAN's particular LES. Each LES is identified by an ATM address. The LES ATM address is supplied to the LEC by the LECS or configured through the user interface.

Broadcast and Unknown Server

The Broadcast and Unknown Server (BUS) is a multicast server that is used to flood unknown destination address traffic and forward multicast and broadcast traffic to clients within a particular ELAN. Each LEC is associated with only a single BUS per ELAN, but there may be multiple BUSs within a particular ELAN. The BUS to which a LEC connects is identified by a unique ATM address. The BUS ATM address is supplied to the LEC by the LES.

Operation

The operation of a LAN emulation system consisting of the components described above consists of three main phases:

- Initialization and configuration
- Joining and registration
- Data transfer

Initialization and Configuration

When the interface becomes active, the LEC must get its ATM address. The LEC then sets up a configuration-direct connection to the LECS. The LEC must find the location of the LECS. The LECS address may be configured in the LEC and the NETBuilder II bridge/router set to Manual so that the LEC sets up the configuration-direct connection with the specified LECS. The LEC also can rely on the UME of the ATM switch to determine an appropriate LECS address.

After finding the location of the LECS, the LEC establishes a configuration-direct VCC to the LECS. When successfully connected, the LECS uses a configuration protocol to inform the LEC of the information it requires to connect to its target ELAN. This information includes the ATM address of the LES, the type of LAN being emulated, the maximum packet size on the emulated LAN, and the emulated LAN name, which consists of a text string. Network management usually configures the LECS with this information.

Joining and Registration

When the LECS gets the LES address, it sets up the control-direct VCC to the LES. When this setup is complete, the LES assigns the LEC with a unique LEC Identifier (LECID). The LEC then registers its own MAC and ATM address with the LES.

The LES then sets the control distribute VCC back to the LEC by adding the LEC as a leaf to a point to multipoint connection. The control direct and distribute VCCs can then be used by the LEC for the LAN emulation ARP (LE_ARP) procedure for requesting the ATM address that corresponds to a particular MAC address. To do this, the LEC formulates a LE_ARP request and sends it to the LES. If the LES recognizes this mapping, it may choose to reply directly on the control-direct VCC. If it does not, it forwards the request on the control-distribute VCC to solicit a response from a LEC that knows the requested MAC address.

If a LEC can respond to the LE_ARP request because it is proxying for that address, the LEC responds to the LES on the control direct VCC. The LES then forwards this response either only to the requesting LEC, or, optionally, on the control-distributed VCC to all LECs. All LECs then can learn and cache the particular address mapping (and save future LE_ARPs).

To complete registration, a LEC uses this LE_ARP mechanism to determine the ATM address of the BUS. The LEC determines the address by sending an LE_ARP for the MAC broadcast address to the LES, which responds with the BUS ATM address. The LEC then sets up the multicast-send VCC to the BUS. The BUS, then sets up the multicast forward VCC back to the LEC by adding the LEC as a leaf to a point-to-multipoint connection. The LEC is now ready to transfer data.

Data Transfer

When a LEC receives a unicast data frame for transmission, it checks its local tables to see whether it knows the ATM address associated with the MAC address. If it does not know the address, the LEC forwards the frame to the BUS to keep the data moving. The BUS responds by forwarding the frame to every client.

Simultaneously, the LEC sends a LE_ARP request to the LES, trying to resolve the unknown MAC address. The LE_ARP message includes the source ATM address of the LEC making the request. The LES searches its database of MACAddr-to-ATM address mappings and returns the ATM address if known through an LE_ARP response. However, in most implementations the LES forwards the LE_ARP to all clients.

The target client recognizes the MAC address and sends an LE_ARP response to the LES, which includes both its own ATM address and the source ATM address for the LEC originating the LE_ARP request. The server forwards the response message with the target ATM address to all the LECs in broadcast fashion. The cycle ends when the originating LEC recognizes its own ATM address contained in the response. At this point, it has learned the ATM address associated with the unknown MAC address and can set up a data-direct connection to the target LEC. When the source LEC receives subsequent frames with the newly learned MAC address, it immediately forwards them down the data-direct VCC.

Each LEC builds up its own table of MAC addresses, ATM addresses, and VCC bindings. If a particular MAC address has not been active for some time. The LEC eventually drops it from its cache. When there are no more MAC addresses associated with a data-direct VCC, the connection will eventually be released due to inactivity.

ATM LAN Emulation Terms

The following terms are used in this chapter to explain ATM:

Asynchronous Transfer Mode (ATM)
A transmission protocol that segments user traffic into small, fixed-sized cells. Cells are transmitted to their destination where the original traffic is reassembled.

ATM Adaptation Layer (AAL)
Layer 3 of the ATM architecture that adapts user traffic into or from ATM 48-byte payloads.

AAL5 supports variable bit rate, delay-tolerant, connection-oriented data traffic requiring minimal sequencing or error-detection support.

Broadcast and Unknown Server (BUS)
BUS defines the set of functions that provide ELAN or LAN emulation transmission support while a switched virtual circuit connection is being established. It also supports LAN emulation broadcast services.

Interim Local Management Interface (ILMI)
Refers to ATM forum-defined interim specifications for network management functions between an end user and a public or private network, and between a public network and a private network. It is based on a limited subset of SNMP capabilities.

LAN emulation
Refers to the emulation of the connectionless nature of a LAN over connection-oriented ATM circuits.

LAN Emulation Client
Defines the set of functions implemented in a DTE to interface with an ATM network in support of LAN emulation.

LAN Emulation Configuration Server
Defines the set of functions that provide LECs with information regarding the location of the LAN emulation servers (LES and BUS).

LAN Emulation Server
Defines the set of functions that support ELAN registration and address resolution.

LAN Emulation User Network Interface (LUNI)
Protocols allowing ATM-attached end systems and LAN/ATM conversion devices to control the virtual connections required for transmission and to emulate the connectionless nature of a LAN.

Permanent Virtual Circuit (PVC)
A virtual channel connection that has been established by manual or semi-automated methods. It is similar to a leased or dedicated real circuit.

Switched Virtual Circuit (SVC)
A virtual channel connection that has been dynamically established in response to a signaling request message.

UNI Management Entity (UME)
The code residing in ATM devices at each end of a UNI interface that implements the management interface to the ATM network.

User-to-network Interface (UNI)
ATM forum-developed specifications for the procedures and protocols between a user DTE and the ATM network to effectively utilize ATM services and capabilities.

Virtual Channel Connection (VCC)
Virtual channels in two or more sequential physical circuits concatenated to create an end-to-end connection. A VCC is a specific instance of a switched virtual circuit (SVC) or a permanent virtual circuit (PVC).

Virtual Channel Identifier (VCI)
The 16-bit number in an ATM cell header identifying the specific virtual channel on which the cell is traversing on the current physical circuit.

Virtual Path Identifier (VPI)
The 8-bit number in an ATM UNI cell header identifying the specific virtual path on which the cell is traversing on the current physical circuit.

49 CONFIGURING CONNECTIONS FOR OUTGOING CALLS

This chapter describes how to configure your bridge/router to function as an X.25 connection service gateway for outgoing calls. The gateway allows end users to make connections from IP Internet-attached Telnet clients, raw Transmission Control Protocol (TCP) clients, and Open Systems Interconnection (OSI) Virtual Terminal Protocol (VTP) clients to X.25-attached hosts that support the X.29 Protocol. Procedures in this chapter include how to make outgoing automatic (one-step) and extended (two-step) connections.

The NETBuilder II bridge/router supports 128 connection service sessions.

For conceptual information, refer to "How the Outgoing Connection Service Works" on page 49-14.

Setting Up the Gateway for Outgoing Telnet Connections

This section describes how to configure the bridge/router gateway to handle outgoing connections for Telnet clients.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the system's local and wide area ports and paths according to the procedure in Chapter 1.
- Configure the X.25 interface.

After completing the procedure for local area and wide area paths and ports, make sure you configure X.25 as the owner of each wide area interface to be used in the outgoing connection service using:

```
SETDefault !<port> -PORT OWNer = X25
```

To configure the X.25 interface, refer to Chapter 45.

Before making outgoing automatic or extended connections, you must provide a list of connection addresses as well as connection and disconnection commands to end users of the connection service gateway. For information on making outgoing connections, refer to "Making Outgoing Connections" on page 49-9.

Procedure

After configuring ports, paths, and the X.25 interface, you need to configure the gateway for outgoing automatic and extended Telnet connections. Use Figure 49-1 and the following procedure to configure the gateway.

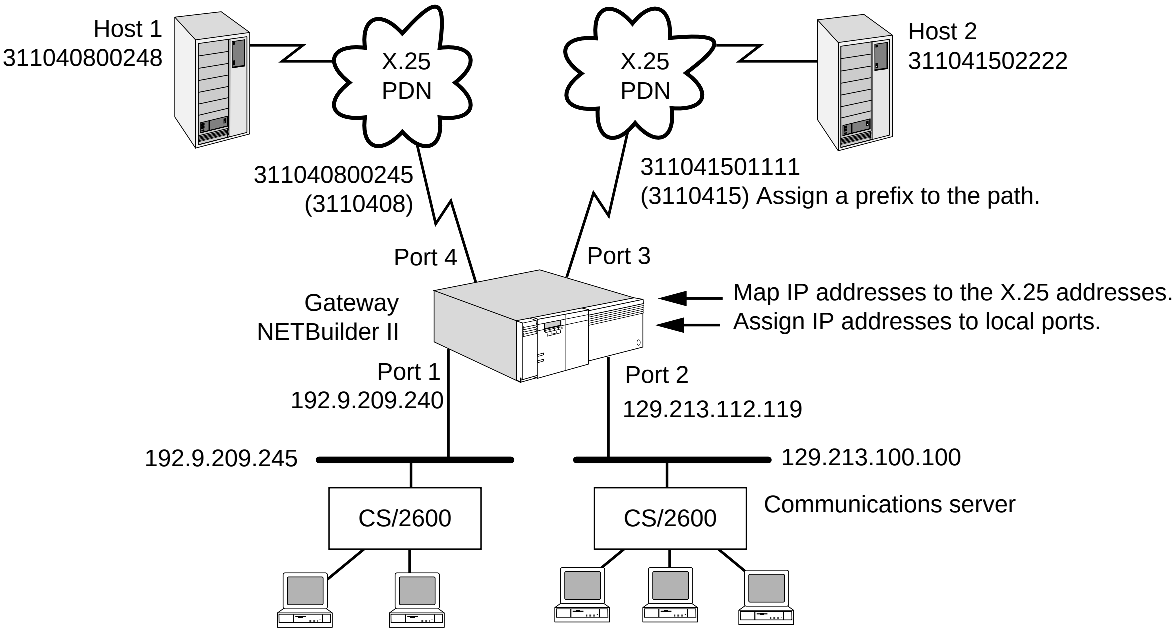


Figure 49-1 Connection Service Configuration Overview (Telnet)

To configure the gateway for outgoing automatic and extended Telnet connections, follow these steps:

1 Before configuring the gateway for outgoing Telnet connections, display address information for directly connected IP networks by entering:

```
SHow -IP NETaddr
```

The following display appears:

```
---------------------IP Directly Connected
  Networks-----------------
IP Address       Port  Subnet Mask    Status   MTU  Broadcast Format
129.213.112.119  2     255.255.255.0  Up      1500  129.213.112.255
```

You need to configure an IP address for each Ethernet interface used for connection service purposes. For example, to add the address 192.9.209.204 for port 1, enter:

```
SETDefault !1 -IP NETaddr = 192.9.209.240 255.255.255.0
```

For the gateway to accept a raw TCP connection, you need to add a listen port using the -TCPAPPL LIStenerPorts parameter.

2 Create a table that maps assigned IP addresses to X.25 addresses of the hosts to which users will want to connect using:

```
ADD [!<configfile>] -Gateway IPX25Map <IPaddress> {X25 addr
  string> | PAD}
```

a Select a configuration file, if necessary.

When creating the table, you can specify a configuration file to initialize the port and session before outgoing connections are made. If you do not specify a configuration file, then configuration file 2 is used as the default.

You can usually use configuration file 2 without modification; the default settings of the TERM Service parameters are acceptable for most outgoing connections. If you require different settings than the defaults already provided, use one of the configuration files numbered 3 through 32.

Configuration file 1 is the default for incoming connections and must not be used for outgoing connections. If you use an odd-numbered configuration file for an outgoing connection, make sure you change the DeVice parameter from Terminal to Host by entering the SETDefault !configfile -TERM DeVice = Host command, or the connection attempt will fail.

For information on TERM Service parameters specifically needed for outgoing connections, refer to Chapter 61 in *Reference for NETBuilder Family Software*. For information on how to map TERM Service parameters to X.3 parameters for outgoing connections, refer to Appendix L.

b Configure an IP address that is on the same network or subnetwork to which the gateway is attached.

An IP address assigned to an X.25 address for establishing an automatic outgoing connection must be valid on some IP subnet to which the gateway is attached. For example, if the gateway has two LAN ports and is configured to route IP packets between these two ports, the gateway will be attached to two IP subnets, and an IP address assigned to an IPX25Map entry must be derived from one of these subnets.

For example, the IP address for port 1 is 192.9.209.240. An IP address with a subnet mask of 255.255.255.0 used with the IPX25Map parameter can have the network portion 192.9.209; you can assign the host portion of the address a subnet between 1 and 254 that has not already been assigned. Similarly, the IP address for port 2 is 129.213.112.119 with a subnet mask of 255.255.255.0, and the network portion is 129.213.112; you can assign the host portion of the address a subnet between 1 and 254 that has not already been assigned. For information about Internet addressing, refer to Appendix D.

c Select either an X.25 information string or the keyword PAD.

The IPX25Map parameter requires that an X.25 address string (used for automatic connections) or the keyword PAD (used for extended connections) follow the IP address. Table 49-1 summarizes the X.25 address strings that can be used with the IPX25Map parameter.

Table 49-1 X.25 Information Strings for Automatic Connections

To Specify	Options	X.25 Address String Example*
X.25 host information	No options	311040800248
	Call user data:	
	Display the data string on the destination terminal	311040800248DHELLO†
	Hide (protect) the string on the destination terminal	311040800248PHELLO†
	Facilities:	
	Reverse charge request (R- or R* can be used)	R-311040800248 R*311040800248

(continued)

Table 49-1 X.25 Information Strings for Automatic Connections (continued)

To Specify	Options	X.25 Address String Example*
	Closed user group (G09- or G09* can be used)	G09-311040800248 G09*311040800248
	Reverse charge request and closed user group (R, G09- or R, G09* can be used)	R,G09-311040800248 R,G09*311040800248
	Facilities and call user data:	
	Reverse charge request and call user data	R*311040800248DHELLO
	Closed user group and call user data	G09*311040800248PHELLO
	Reverse charge, closed user group, and call user data	R, G09-311040800248DHELLO
Private line	No options	L (The gateway uses the lowest path enabled for connection service.)
	A path	L3 (directly selects line 3)
	A path and call user data	L4DHELLO (directly uses line 4 with call user data)

* P, D, R, G, L and call user data can be entered in upper- or lowercase.

† P and D distinguish the end of the X.25 address from the call user data.

d Configure address mappings for automatic connections.

The following three examples show IP-to-X.25 address mappings for automatic connections:

- To map the IP address 192.9.209.100 to the X.25 host address 311040800248 with reverse charging and to initialize the port and session with configuration file 2, enter:

```
ADD -Gateway IPX25Map 192.9.209.100 R*311040800248
```

- If no configuration file is specified in the command, as shown in this example, the gateway automatically uses configuration file 2 as the default.
- To map the IP address 192.9.209.101 to the X.25 host address 311041502222 with reverse charging and closed user group facilities, and to initialize the port and session with configuration file 4, enter:

```
ADD !4 -Gateway IPX25Map 192.9.209.101 R,G09*311041502222
```

- To map the IP address 129.213.112.120 to the X.25 host address 311041502222 with reverse charging and to send call user data, and to initialize the port and session with configuration file 4, enter:

```
ADD !4 -Gateway IPX25Map 129.213.112.120 R-311041502222DHELLO
```

With automatic connections, the gateway automatically places a call to the destination X.25 host address, supports the named facility, reverse charge and/or closed user group, and sends call user data as requested. For more information, refer to "Making Outgoing Connections" on page 49-9 and "Automatic Connections" on page 49-10.

e Configure address mappings for extended connections.

The following example shows how to make an IP-to-X.25 address map for an extended connection. To place the caller who makes a connection request into PAD emulation mode and to initialize the port and session with configuration file 4, enter the PAD keyword after the IP address as follows:

`ADD !4 -Gateway IPX25Map 129.213.112.121 PAD`

PAD emulation mode allows the caller to select a different profile, alter PAD parameters, and establish virtual calls to X.25 hosts. For information on PAD emulation mode, refer to "Making Outgoing Connections" on page 49-9 and "Extended Connections" on page 49-10.

f Display the IP address-to-X.25 address mappings by entering:

`SHow -Gateway IPX25Map`

The following display appears:

```
Config File  IP Address            X.25 Information
-------------------------------------------------------------
! 2          192.9.209.100         R*311040800248
! 4          192.9.209.101         R,G09*311041502222
! 4          129.213.112.120       R-311041502222DHELLO
! 4          129.213.112.121       (PAD mode)
```

3 Configure the X25Prefix parameter in the XSWitch Service so that the gateway can select the wide area path for reaching the X.25 host.

Make sure to configure this parameter for each wide area path that is enabled for connection service.

The prefix is a series of numbers that match the destination X.25 address in part or completely. For example, to configure the gateway's wide area port 3 with a prefix that identifies the X.25 host with address 311041502222, enter:

`ADD !3 -XSWitch X25Prefix 3110415`

When a connection request is made, the gateway scans the prefix table for a prefix that matches the target X.25 address. If no match exists for the target X.25 address, the connection request is denied. To prevent a connection denial because of no matching prefix, you can select one default port for the gateway. For example, you can select port 4 to be the default port by using the default option in the following command:

`ADD !4 -XSWitch X25Prefix Default`

To display the prefix-to-port mapping, enter:

`SHow -XSWitch X25Prefix`

The following display appears:

```
Port #/IPAddr     X.25 Prefix
--------------------------------------
!4                Default
!3                3110415
!4                3110408
```

4 Verify that the gateway paths are configured for outgoing automatic and extended connections.

For example, to verify path 3 is configured, enter:

`SHow !3 -Gateway CONTrol`

The following display appears:

```
Path !3 CONTrol = (Enable, InExt, OutExt, InAuto, OutAuto, DDXP,
  SubAddr, NoDSA, NoTrace)
```

By default, the CONTrol parameter is disabled; however, it automatically becomes enabled when the X.25 path comes up.

If the display is incorrect, you can configure the path for the type of operation desired. For example, to enable the gateway on wide area path 3 for both outgoing automatic and extended connections, enter:

```
SETDefault !3 -Gateway CONTrol = (Enable, OutExt, OutAuto)
```

After configuring the X25 Service and Gateway Service parameters, refer to the next section for information on making outgoing automatic and extended connections.

Setting Up the Gateway for Outgoing VTP Connections

This section describes how to configure the bridge/router gateway to handle outgoing connections for OSI VTP clients.

Prerequisites

Before beginning this procedure, complete the following steps:

- Log on to the system with Network Manager privilege.
- Set up the local and wide area ports and paths using the procedure in Chapter 1.
- Configure the X.25 interface.

After completing the procedure for local area and wide area paths and ports, make sure you configure X.25 as the owner of each wide area interface to be used in the outgoing connection service. For example, for each wide area interface use:

```
SETDefault !<port> -PORT OWNer = X25
```

Procedure

After configuring ports, paths, and the X.25 interface, you need to configure the gateway for outgoing automatic and extended VTP (OSI) connections. Use Figure 49-2 and the following procedure to configure the gateway.

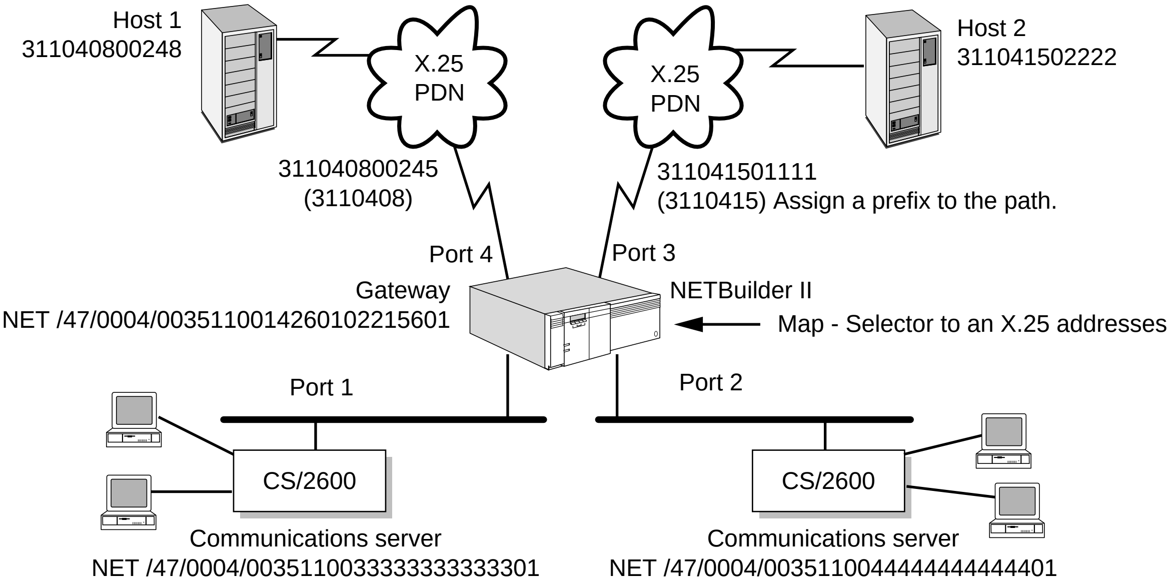


Figure 49-2 Connection Service Configuration Overview (OSI)

To configure the gateway for outgoing automatic and extended VTP (OSI) connections, follow these steps:

1 Before configuring the gateway for outgoing OSI connections, display the Network Entity Title (NET) by entering:

```
SHow -CLNP NetEntityTitle
```

The following display appears:

```
NetEntityTitle = /49/005308000212345600
```

For more information about configuring the NetEntityTitle, refer to the AreaAddress parameter in Chapter 32 in *Reference for NETBuilder Family Software*.

2 Create a table that maps assigned P-Selector to X.25 addresses of the host to which users will want to connect.

Use:

```
SETDefault !<P-Sel> -Gateway PSelX25Map {[!<config file>] (<x.25
  addr string> | PAD ) | None}
```

The P-Selector in the Presentation Address must be 2 octets in length and the value of the first octet must be 0. When you want to make a connection using Telnet profiles, and 4 is used, 0 or 4 is used for the X.3 profiles. As a result, the mapping is only for the second octet of the P-Selector. The X.3 profile can only be applied for outgoing automatic connections.

a If you are using Telnet profiles, select a configuration file if necessary.

When creating the table, you can specify a configuration file to initialize the port and session before outgoing connections are made. If you do not specify a configuration file, then configuration file 2 is used as the default. You can usually use configuration file 2 without modification; the default settings of the TERM Service parameters are acceptable for most outgoing connections. If you require different settings from the defaults already provided, use one of the configuration files numbered 3 through 32.

Configuration file 1 is the default for incoming connections and must not be used for outgoing connections. If you use an odd-numbered configuration file for an outgoing connection, make sure you change the DeVice parameter from Terminal to Host by entering the SETDefault !configfile -TERM DeVice = Host command, or the connection attempt will fail.

For information on TERM Service parameters specifically needed for outgoing connections, refer to Chapter 61 in *Reference for NETBuilder Family Software*. For information on how to map TERM Service parameters to X.3 parameters for outgoing connections, refer to Appendix L.

b Configure the second octet of the P-Selector to map to an X.25 address.

Use:

```
SETDefault !<P-Sel> -Gateway PSelX25Map
```

Select either an X.25 address string or the keyword PAD.

The PSelX25Map parameter requires either an X.25 address string (used for automatic connections) or the keyword PAD (used for extended connections). For example, to map P-Selector 4 to the X.25 address 311040800248, enter:

```
SETDefault !4 -Gateway PSelX25Map = 311040800248
```

To set P-Selector 2 to PAD for extended connections, enter:

```
SETDefault !2 -Gateway PSelX25Map = PAD
```

Table 49-1 summarizes the X.25 address strings that can be used with the PSelX25Map parameter.

c To display the P-Selector-to-X.25 address mappings, enter:

```
SHow -Gateway PSelX25Map
```

The following display appears:

```
Config File     P-Selector     X.25 Information
-----------------------------------------------
!2              01             311040800248
!2              02             (PAD Mode)
```

3 Configure the X25Prefix parameter in the XSWitch Service so that the gateway can select the wide area path for reaching the X.25 host.

Make sure you configure this parameter for each wide area path that is enabled for connection service.

The prefix is a series of numbers that match the destination X.25 address in part or completely. For example, to configure the gateway's wide area port 3 with a prefix that identifies the X.25 host with address 311041502222, enter:

```
ADD !3 -XSWitch X25Prefix 3110415
```

When a connection request is made, the gateway scans the prefix table for a prefix that matches the target X.25 address. If no match exists for the target X.25 address, the connection request is denied. To prevent a connection denial because of no matching prefix, you can select one default port for the gateway. For example, you can select port 4 to be the default port by using the Default option when you enter:

```
ADD !4 -XSWitch X25Prefix Default
```

To display the prefix-to-port mapping, enter:

```
SHow -XSWitch X25Prefix
```

The following display appears:

```
Port #/IPAddr       X.25 Prefix
------------------------------------
!4                  Default
!3                  3110415
!4                  3110408
```

4 Verify that the gateway paths are configured for outgoing automatic and extended connections.

For example, to verify path 3 is configured, enter:

```
SHow !3 -Gateway CONTrol
```

The following display appears:

```
Path !3 CONTrol = (Enable, InExt, OutExt, InAuto, OutAuto, DDXP,
  SubAddr, NoDSA, NoTrace)
```

By default, the CONTrol parameter is disabled. However, it automatically becomes enabled when the X.25 path comes up.

If the display is incorrect, you can configure the path for the type of operation desired. For example, to enable the gateway on wide area path 3 for both outgoing automatic and extended connections, enter:

```
SETDefault !3 -Gateway CONTrol = (Enable, OutExt, OutAuto)
```

The gateway also supports the X.3 VT profile. From the VT client, issue a connection to the gateway, setting the first octet of the P selector to 4; the second octet may be any value. The X.25 address is part of the VT-profile parameters and is carried on the connect request, so no configuration on the gateway is required.

After configuring the X25 Service and Gateway Service parameters, refer to "Making Outgoing Connections" page 49-9 for information on making outgoing automatic and extended connections.

Making Outgoing Connections

Before making automatic or extended outgoing connections, you must provide a list of connection addresses as well as connection and disconnection commands to end users of the connection service gateway. Configure IP connection addresses with the IPX25Map parameter or OSI connection addresses with the PSelX25Map parameter.

The following is an example of IP-to-X.25 address mappings:

```
Config File   IP Address         X.25 Information
-------------------------------------------------------------
! 2           192.9.209.100      R*311040800248
! 2           192.9.209.101      R,G09*311041502222
! 2           129.213.112.120    R-311041502222DHELLO
! 2           129.213.112.121    (PAD mode)
```

The following is an example of P-Selector-to-X.25 address mappings:

```
Config File   P-Selector    X.25 Information
--------------------------------------------
!2            01            311040800248
!2            02            (PAD Mode)
```

Users can make Telnet and VTP connections using the list of connection addresses. The connection command used depends on the commands that are available on the device from which the connection is made. For example, if a user initiates a connection request from a terminal connected to a 3Com communications server, then communications server commands such as the Connect command can be used. When connecting from another device, consult the documentation that ships with that device for information on commands that can be used, and make sure you provide the appropriate commands to users of the connection service gateway.

During outgoing connection establishment, the gateway selects a port through which the connection is made. These ports are not physical ports, but virtual ports. The gateway selects the next available port, and initializes the port and session with the specified configuration file (if none is specified, configuration file 2 is used), except OSI connections using X.3 profiles, which do not require the configuration file. No correlation exists between the selection of the port and the configuration file that initializes it. For example, the gateway could select port 8 and initialize it with configuration file 2. On the next connection to the same destination, the gateway could select port 60 and initialize with configuration file 2.

For information on making automatic connections, read the next section. For information on making extended connections, refer to "Extended Connections" page 49-10.

Automatic Connections

When you initiate a connection request to the IP address of 192.9.209.101, for example, the gateway receives the request, locates the matching entry in the address mapping table, and uses the destination X.25 address and other information to place the call. In this example, the gateway expects to find a reverse charge facility offered at destination address 311040800248. The gateway also examines the following prefix table:

```
Port #/IPAddr     X.25 Prefix
-----------------------------
!4                Default
!3                3110415
!4                3110408
```

The gateway finds that prefix 3110408 can be reached on port 4. The gateway places the call to the destination X.25 host on port 4, and initializes the port and session with the parameter settings in configuration file 2. When the connection is established, the host prompt appears.

If the gateway cannot match the destination address with an address in the address mapping table and the prefix in the prefix table (no default path has been defined), the gateway rejects the connection and displays a message similar to "connection refused." The exact wording of the message is Telnet-client or VTP-client-dependent. If the gateway can match a destination address, but not a prefix, and a default port is defined for the X25Prefix parameter, the gateway uses the default port to place the call.

When you complete the session with the host, you need to end the session. The command that you use depends on the host.

Extended Connections

This section describes to an end user the packet assembler/disassembler (PAD) emulation mode features and how to make an extended connection. When you initiate a connection request to the IP address of 129.213.112.121, for example, the gateway receives the request and locates the matching entry in the address mapping table that has no X.25 address. In this example, the gateway finds a destination match and places you into PAD mode, which is indicated by the NB-PAD> prompt.

When you establish an X.25 virtual call from PAD mode and the connection is established, the gateway displays on-screen messages indicating that you are connected. If the virtual call is rejected, the on-screen message is "CLR 0 0." For more information about establishing virtual calls, refer to "Establishing a Virtual Call" on page 49-11.

In PAD emulation mode, you can perform the following actions:

- Select individual PAD parameter values.
- Request the current values of PAD parameters to be transmitted by the PAD to the host.
- Establish and clear a virtual call.

The PAD emulation user interface also supports the use of call user data and facilities with the command issued to establish a virtual call. Facilities include reverse charge requests and basic closed user groups.

The PAD emulation user interface provided by the X.25 connection service has limited functionality and only supports some of the capabilities described in CCITT Recommendation X.28. These supported capabilities and the command syntax for invoking them are described in the following sections.

Selecting Individual PAD Parameters

After selecting a default PAD profile, you can assign new values to individual parameters (overriding the default values) by using the SET command.

For example, to set the values of parameter 2 to 0, parameter 3 to 2, and parameter 9 to 4, at the NB-PAD> prompt, enter:

```
SET 2:0, 3:2, 9:4
```

To set the values of parameter 2 to 0, parameter 3 to 2, parameter 9 to 4, and to read the set values back, at the NB-PAD> prompt, enter:

```
SET? 2:0, 3:2, 9:4
```

Requesting Current Values of PAD Parameters

You can read the values currently assigned to individual PAD parameters by using the PAR? command. To read the current values for parameters 2, 3, and 9, at the NB-PAD> prompt, enter:

```
PAR? 2,3,9
```

Establishing a Virtual Call

You can establish a virtual call to an X.25 destination address by supplying the following information:

- The X.25 address

  To establish a virtual call to a host whose X.25 address is 311040800248, enter the X.25 address at the NB-PAD> prompt as follows:

  ```
  311040800248
  ```

- The X.25 address with optional call-user data

  To establish a virtual call to a host whose X.25 address is 311040800248 and to transmit a call-user data string “HELLO” with the call request, enter one of the following strings at the NB-PAD> prompt:

  ```
  311040800248DHELLO
  311040800248PHELLO
  ```

  The D and P distinguish the end of the address from the call-user data. Use D when you want the gateway to display the data string it is sending as call-user data on the call. Use P if you want to hide (protect) the data, for example, when sending passwords to the host.

- The X.25 address with optional facilities and with optional call-user data

  The connection service supports two facility requests: reverse charge request and basic closed user group selection.

  To establish a virtual call to a host whose X.25 address is 311040800248 and to request reverse charging, enter one of the following strings at the NB-PAD> prompt:

  ```
  R-311040800248
  R*311040800248
  ```

  Either R- or R* can be used to indicate reverse charging.

  To establish a virtual call to a host whose X.25 address is 311040800248 and to request a closed user group selection, enter one of the following strings at the NB-PAD> prompt:

  ```
  G09-311040800248
  G09*311040800248
  ```

  Either G09- or G09* can be used to indicate closed user group.

  To establish a virtual call to a host whose X.25 address is 311040800248 and to request both the reverse charging and closed user group, enter one of the following strings at the NB-PAD> prompt:

  ```
  R,G09-311040800248
  R,G09*311040800248
  ```

  To establish a virtual call to a host whose X.25 address is 311040800248, to request both the reverse charging and closed user group, and to specify that the user data “HELLO” be transmitted as call user data with the call request, enter one of the following strings at the NB-PAD> prompt:

  ```
  R,G09-311040800248DHELLO
  R,G09*311040800248DHELLO
  R,G09*311040800248PHELLO
  ```

- The path on the gateway to be used for establishing a connection

  To establish a virtual call to a host that is connected over a private line and is not identified by an X.25 address, and to select the gateway path 3 (on which X.25 connection service is enabled) to be used for the connection, enter the following string at the NB-PAD> prompt:

  ```
  L3
  ```

  To specify that the user data “HELLO” be passed as call user data with the call request, enter one of the following strings at the NB-PAD> prompt:

  ```
  L3DHELLO
  L3PHELLO
  ```

  To allow the gateway to select a line for establishing the call, enter the following string at the NB-PAD> prompt:

  ```
  L
  ```

  The gateway selects the lowest numbered path that is enabled for connection service.

When you establish a virtual call by using one of the previously described methods, the host displays a greeting or prompt, the appearance and format of which is host-dependent.

When you complete the session with the host, you need to end the session. The command used depends on the host. You can escape from the X.25 host by entering the PAD recall character (usually [Ctrl] + P) to return you to the NB-PAD> prompt, and return back to the X.25 host by entering another PAD recall character.

Clearing a Virtual Call

You can exit from the PAD mode prompt back to the original Telnet or VTP initiator by entering:

```
CLeaR
```

Troubleshooting Outgoing Connections

If you encounter problems with the connection service gateway, verify that the settings in the PATH, PORT, X25, LAPB (if used), and Gateway Services are correct as follows:

- To verify the control, state, baud, connector, and clock settings, enter:

  ```
  SHow -PATH CONFiguration
  ```

- To verify that the owner of the wide area ports used in the connection service is X.25, enter:

  ```
  SHow -PORT CONFiguration
  ```

- To verify the interface type, the X.25 address, and the PDN network type, enter:

  ```
  SHow -X25 CONFiguration
  ```

- To verify the settings of the LAPB Service parameters, enter:

  ```
  SHow -LAPB CONFiguration
  ```

  For additional information, refer to Chapter 33 in *Reference for NETBuilder Family Software*.

- To verify the settings for the path used in the connection service, enter:

  ```
  SHow -Gateway CONTrol
  ```

If connection requests continue to fail, enable the X.25 trace feature by using the SETDefault !<path> -X25 Trace = (Data, Control) syntax. Use SHow !<path> -X25 Trace to display data and/or control information for the specified X.25 path at the network layer. For more information about the Trace parameter, refer to Chapter 65 in *Reference for NETBuilder Family Software*.

Make sure to turn Trace off after you are finished using it because it slows down the performance of your bridge/router.

You can enable tracing by using the SETDefault !<path> -Gateway CONTrol = Trace syntax to obtain additional information. After setting the CONTrol parameter, establish (or attempt to establish) a connection with the X.25 host. The screen displays information that can be used for debugging.

You can also display active session information such as the source and destination address by entering:

```
SHow -Gateway PadSession.
```

For more information on the CONTrol and PadSession parameters, refer to Chapter 26 in *Reference for NETBuilder Family Software*.

How the Outgoing Connection Service Works

The X.25 connection service gateway allows IP Internet-attached Telnet clients and OSI VTP clients to connect to X.25-attached hosts that support the X.29 Protocol. The Telnet or OSI VTP clients can be PCs or workstations running Telnet client software or VTP client software, or asynchronous dumb terminals connected to a communications server that supports the Telnet and/or VTP protocol. LAN-to-WAN connections are also referred to as *outgoing connections* and are controlled by the outgoing connection service of the gateway. Figure 49-3 is an example of outgoing LAN-to-WAN connections.

Login is not supported on outgoing calls.

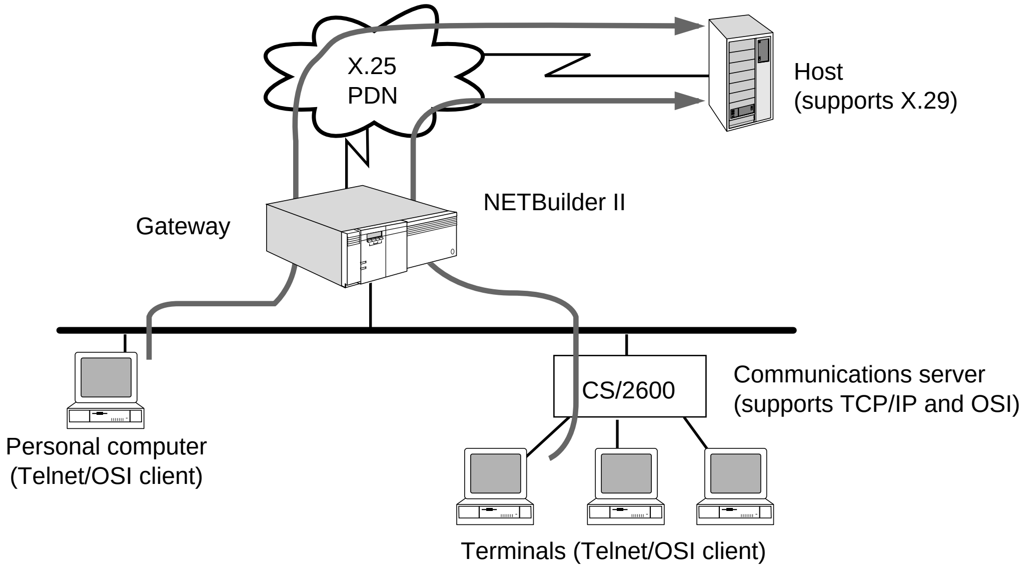


Figure 49-3 LAN-to-WAN Connections (Outgoing)

The X.25 connection service can also be used to front-end a host that does not support a LAN interface, but has an X.25 interface and supports the X.29 Protocol as shown in Figure 49-4. This configuration is very similar to the one shown in Figure 49-3, except that neither the connection service gateway nor the host is connected to an X.25 public data network (PDN). The gateway and the host instead are connected directly to each other back-to-back, using X.25 for terminal connections.

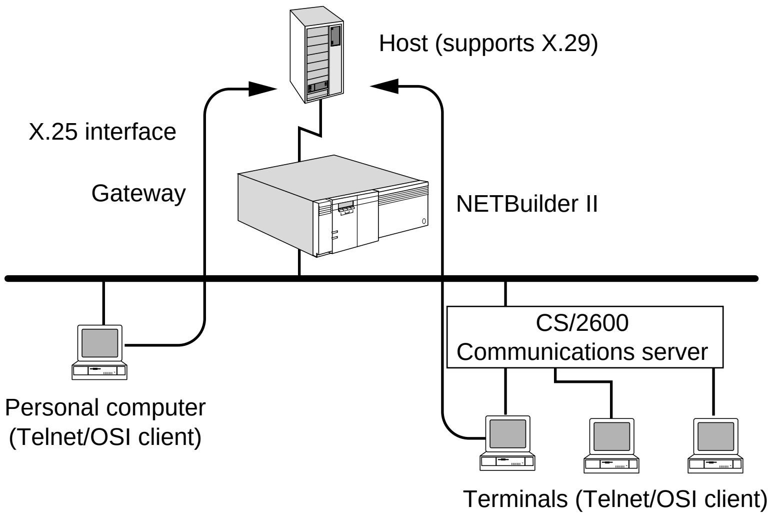


Figure 49-4 Host Front-End Connections (Outgoing)

The X.25 connection service gateway offers two types of outgoing connections:

- Automatic (one-step)

  End users can enter a connection command from the Telnet client or OSI VTP client and the gateway automatically establishes the link to the X.25 host.

- Extended (two-step)

  End users can enter a connection command from the Telnet client or OSI VTP client and establish a connection with the gateway's PAD emulation user interface. Once in PAD emulation mode, users can execute a connection command to the desired host by providing the appropriate information.

With outgoing connections, you are limited to connecting to a single host with each Telnet or VTP connection.

50 CONFIGURING CONNECTIONS FOR INCOMING CALLS

This chapter describes how to configure your bridge/router to function as an X.25 connection service gateway for incoming calls. The gateway allows end users to make connections from X.25 packet assembler/disassembler (PAD)-attached terminals to IP Internet-attached Telnet, Rlogin servers, or Rlogin hosts. This chapter describes procedures for making incoming automatic (one-step) and extended (two-step) connections, configuring name services for Transmission Control Protocol/Internet Protocol (TCP/IP) connections, configuring Rlogin connections, and selecting the name service for Open System Interconnection (OSI) connections.

For conceptual information, refer to "How the Incoming Connection Service Works" on page 50-23.

Configuring the Gateway for Incoming Connections

This section describes how to configure the bridge/router gateway to handle incoming connections.

Prerequisites

Before beginning this procedure, complete the following tasks:

- Log on to the system with Network Manager privilege.
- Set up the system's local and wide area ports and paths by referring to Chapter 1.
- Configure the X.25 interface.

After completing the procedure for configuring local area and wide area paths and ports, configure X.25 as the owner of each wide area interface to be used in the incoming connection service using:

```
SETDefault !<port> -PORT OWNer = X25
```

To configure the X.25 interface, refer to Chapter 65. You may also want to use the data compression feature; for detailed information, refer to Chapter 39.

You must configure the gateway paths for incoming automatic and extended connections before you configure the bridge/router. Refer to "Making Incoming Connections" on page 50-2.

Procedure

To configure the gateway paths for incoming automatic and extended connections, follow these steps:

1 Verify that the gateway paths are configured for incoming automatic and extended connections.

For example, to display information for path 3, enter:

```
SHow !3 -Gateway CONTrol
```

The following display appears:

```
Path !3 CONTrol = (Enable, InExt, OutExt, InAuto, OutAuto, DDXP,
  SubAddr, NoDSA, NoTrace)
```

By default, the CONTrol parameter is disabled. However, it automatically becomes enabled when the X.25 path comes up.

2 If the display is incorrect, configure the path for the desired type of operation using:

```
SETDefault !<path> -Gateway CONTrol = (Enable, InAuto, InExt)
```

After configuring the X25 Service and the Gateway Service parameters, refer to the next section for information on making incoming automatic and extended connections.

Making Incoming Connections

To initiate a connection request from an X.25 PAD-attached terminal, end users must use commands that are supported by the PAD service provider. Examples provided in this section use a general command syntax and consist of elements that are generally made available by PAD service providers.

During incoming connection establishment, the gateway selects a port through which the connection is made. These ports are not physical ports, but virtual ports, and range in number from 0 to 127 on the NETBuilder II system. The gateway selects the next available port, and initializes the port and session with the specified configuration file (if none is specified, configuration file 1 is used). No correlation exists between the selection of the port and the configuration file that initializes it. For example, the gateway could select port 3 and initialize it with configuration file 1. On the next connection establishment to the same destination, the gateway could select port 7 and initialize it with configuration file 1. For information on configuration files, refer to "Using Configuration Files" on page 50-3 and "Creating Port-Initialization Macros" on page 50-6.

For information on making incoming automatic connections, refer to the next section. For information on making incoming extended connections, refer to "Extended Connections" on page 50-4.

Automatic Connections

When making an automatic connection request from the PAD-attached terminal to a Telnet, Rlogin, or OSI server, you must identify the X.25 address of the gateway in addition to the destination server. You can specify the destination server as X.25 Call User Data (data to be sent to the gateway along with the call request) in one of the following ways:

- Host address (IP address)
- Host name (IP or OSI)
- Configuration file number

Automatic incoming connections are also supported for subaddress mapping. For more information on configuring a subaddress map, refer to the description for the SubAddrMap parameter in Chapter 26 in *Reference for NETBuilder Family Software*.

Using Addresses

To connect a PAD-attached terminal user to the Telnet server whose IP address is 129.213.112.009:

```
<connect> 311040800245 D 129213112009
```

You must substitute a connection command for <connect> that is supported by the PAD service provider.

This command specifies the gateway's X.25 address of 311040800245; X.25 Call User Data follows and specifies the destination IP address of 129.213.112.009. When you supply the IP address as Call User Data, be sure to include zeros. For example, do not write 129213112009 as 1292131129. The letter D separates the X.25 address from the Call User Data.

The gateway uses the addressing information to automatically place the call to the destination. The gateway selects the next available port, and initializes the port and session with configuration file 1. To disconnect the session, use an exit command. The specific command that is entered is host-dependent.

Using Names

To connect a PAD-attached terminal user to the gateway's X.25 address and to a server named "marketing" use:

```
<connect> 311040800245 D marketing
```

You must substitute a connection command for <connect> that is supported by the PAD service provider.

When you use a name to make an incoming automatic connection, the name can be no longer than 12 characters to conform to X.25 Call User Data limitations.

This command specifies the gateway's X.25 address of 311040800245; Call User Data follows with a name. The name "marketing" identifies the host; the gateway resolves the name through the IP (Domain name or IEN116) name resolver, or the OSI Name Server (X.500 or File) and automatically places the call to the destination. The gateway selects the next available port, and initializes the port and session with configuration file 1. To disconnect from the session, use an exit command. The specific command that is entered is host-dependent. For network manager information on configuring name services, refer to "Name Service for TCP/IP Connections" on page 50-9 or "Name Service for OSI Connections" on page 50-13.

Using Configuration Files

To automatically connect a PAD-attached terminal user to the gateway's X.25 address and to a server whose address or name is specified in a port-initialization macro called by a configuration file use:

```
<connect> 311040800245 D 03
```

You must substitute a connection command for <connect> that is supported by the PAD service provider.

This command connects a PAD-attached terminal user to the gateway's X.25 address of 311040800245; Call User Data follows with a configuration file number. The configuration file calls a port-initialization macro, which contains a TELnet, RLOGin, or VTp connection command to a server.

During connection establishment, the gateway selects the next available port, initializes the port and session with the settings in the specified configuration file, and automatically places the call to the destination.

The gateway supports only one session with incoming automatic connections. Access control must be disabled for automatic connections using the configuration file.

To disconnect the session, use an exit command (host-dependent).

For network manager information about creating port-initialization macros, refer to "Creating Port-Initialization Macros" on page 50-6.

Extended Connections

An extended connection request occurs in two steps. First, you must identify the X.25 address of the gateway in the connection command and make a connection to the gateway. Second, you can connect to Telnet or Rlogin servers on the IP Internet, or to an OSI host. You also can configure and manage the 3Com bridge/router.

To make an extended connection, follow these steps:

1 Make an X.25 connection to the gateway using:

```
<connect> 311040800245
```

You must substitute a connection command for <connect> that is supported by the PAD service provider.

If access control is disabled, you are placed into the management/configuration interface, indicated by the NETBuilder> prompt. The interface is the same interface that is seen when you connect to the system through a local console, through a Telnet connection, or through an OSI VTP connection.

If access control is enabled, you must supply a valid user name and password assigned to you by the network manager at the Netlogin prompt. If the name is invalid, the gateway rejects the connection; otherwise, you are placed into the bridge/router's user interface. For network manager information on configuring access control, refer to Chapter 51.

When you have the NETBuilder> prompt, you can make connections to Telnet, Rlogin, or VTP hosts as described in step 2. If you have Network Manager privilege and want to manage or configure the bridge/router, proceed to step 3.

2 From the NETBuilder> prompt, make connections to TCP/IP or OSI hosts by using the gateway connection service commands.

You can use the TELnet, RLOGin, or Connect commands to connect to a TCP/IP host. With each of the connection commands, you can use a name or an IP address. For example, you can enter one of the following commands:

```
Telnet 129.213.112.9
Connect host1
```

You can use the VTp or Connect commands to connect to an OSI host. With these commands, you can use a name or an OSI address. For example, you can enter one of the following commands:

```
VTp /47/0004/003511003C3C3C5C3C3C01 !9
Connect /47/0004/003511003C3C3C5C3C3C01 !9
```

N-selectors, T-selectors, and S-selectors in the PSAP address are host-dependent. For more information about PSAP addressing, refer to Appendix E.

If you use a name, the gateway performs the name resolution through the IP (Domain name or IEN116) name resolver, or the X.500 DUA. For network manager information on configuring name services, refer to "Name Service for TCP/IP Connections" on page 50-9 or "Name Service for OSI Connections" on page 50-13.

You can also make connections to multiple destinations by entering ECM with the Connect, TELnet, RLOGin, and VTp commands. For more information, refer to Chapter 52.

3 From the NETBuilder> prompt, manage or configure the bridge/router.

You must have Network Manager privilege and not be restricted through access control. For network manager information on configuring access control, refer to Chapter 51.

4 When you complete the extended session with the host, end the session.

The commands used to exit or logout are provided by the host. After entering an exit command, you are returned to the NETBuilder prompt.

5 To disconnect your session from the gateway and return to the PAD terminal prompt, enter one of the following commands:

```
LIsten
LOGout
```

If you have Network Manager privilege, you can disconnect another user's session by specifying their port number. The following command disconnects the user's session on port 3 and puts the port into listen mode:

```
LOGout !3
```

For additional information about these commands, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

Troubleshooting Incoming Connections

If you encounter problems with the connection service gateway, verify that the settings in the PATH, PORT, X25, LAPB (if used), and Gateway Services are correct as follows:

- To verify the control, state, baud, connector, and clock settings, enter:

  ```
  SHow -PATH CONFiguration
  ```

- To verify that the owner of the wide area ports used in the connection service is X.25, enter:

  ```
  SHow -PORT CONFiguration
  ```

- To verify the interface type, the X.25 address, and the public data network (PDN) network type, enter:

  ```
  SHow -X25 CONFiguration
  ```

- To verify the settings of the LAPB Service parameters, enter:

  SHow -LAPB CONFiguration

- For additional information, refer to Chapter 33 in *Reference for NETBuilder Family Software*.
- To verify the settings for the path used in the connection service, enter:

  SHow -Gateway CONTrol

If connection requests continue to fail, enable the X.25 trace feature using:

```
SETDefault !<path> -X25 Trace = (Data, Control)
```

To display data and control information for the specified X.25 path at the network layer, use the SHow !<path> -X25 Trace syntax For more information about the Trace parameter, refer to Chapter 66 in *Reference for NETBuilder Family Software*.

Enable tracing using the SETDefault !<path> -Gateway CONTrol = Trace syntax to obtain additional information. After setting the CONTrol parameter, establish (or attempt to establish) a connection with the X.25 host. The screen displays information that can be used for debugging.

To display a history of the status of the last few sessions, use the SHow !<path> -Gateway ConnHistory syntax. Additional information on the CONTrol and ConnHistory parameters can be found in Chapter 26 in *Reference for NETBuilder Family Software*.

Customizing the Incoming Connection Service

This section describes how you can customize the incoming connection service.

Creating Port-Initialization Macros

You can specify a configuration file as Call User Data to be used with an incoming automatic connection request (for example, connect 311040800245 D 01). After receiving the connection request, the gateway uses the configuration file to call a port-initialization macro previously assigned to a configuration file by the network manager through the -TERM InitMacro parameter. The macro must contain a connection command to a Telnet, Rlogin, or OSI host to automate the incoming connection. During connection establishment, the gateway selects the next available port, initializes the port and session with the contents of the configuration file, and makes the connection to the host through the macro.

Although configuration files numbered 1 through 32 are valid for incoming automatic connections using configuration files, you should use configuration file 1 as the default. You can usually use configuration file 1 without modification; the default settings of the TERM Service parameters are acceptable for most incoming connections. If you require different settings from the defaults already provided, use one of the configuration files numbered 3 through 32.

Configuration file 2 is the default for outgoing connections and must not be used for incoming connections. If you use an even-numbered configuration file *for an incoming connection, make sure you change the DeVice parameter from Host to Terminal using the SETDefault !configfile -TERM DeVice = Terminal command, or the connection attempt will fail.*

This section describes how to:

- Create and manage macros.
- Assign a macro to a configuration file number.

Table 50-1 lists commands that are used for creating and managing macros, and assigning the macro to a configuration file number. You can find more detailed macro information in Appendix G.

Table 50-1 Commands for Creating and Managing Macros

Procedure	Command	Function
Creating macros	`DEFine <macroname> = <text>`	Defines a macro.
	`PAuse`	Makes the system pause for one second.
	`PAuse <seconds>`	Makes the system pause for the specified number of seconds during macro execution.
	`Echo "<string>"`	Displays the specified string on the terminal during macro execution.
Assigning a macro to a configuration file number	`SETDefault !<configfile> -TERM InitMacro = "<macroname>"`	Defines port-initialization macros.
Managing macros	`SETDefault !<configfile> -TERM InterActTerm = NoMacroEcho`	Suppresses the display as the macro is executed.
	`SETDefault !<configfile> -TERM InterActTerm = NoMacroBreak`	Prevents macro termination with the Break key.
	Press Break key	Stops execution of the macro.
	`UNDefine <macroname>`	Deletes the macro.
	`FLush -SYS MACros`	Removes all macros from the macro cache.
	`SHow -SYS MACros`	Displays all macro names.
	`SHow -SYS MACros <macroname>`	Displays contents of specified macros.

Creating Macros

A macro is a file that contains a series of individual commands that automates the incoming automatic connection. You can create a macro if you have Network Manager privilege. The macro can consist of a connection command or a series of connection commands in a menu-driven interface, as well as X.3-type parameter settings in the TERM Service that are used to initialize the session with the host.

After the macro is created, you can assign the macro as a port-initialization macro and give it a configuration file number. Each time the X.25 PAD-attached terminal user specifies the configuration file number as Call User Data in the incoming automatic connection request, the gateway automatically executes the port-initialization macro that contains the connection command.

You use the DEFine command to create a macro file and specify its contents. When a new macro is created with the same name as an existing macro, the new macro contents replace the old macro contents.

A single macro can contain up to 256 characters. Macro contents must begin with a left parenthesis. If the definition requires more than one line, press the Return key after the opening parenthesis. After you press the Return key, the Macro: prompt appears on the next line. All characters entered between the opening and closing parentheses are part of the macro. Nested parentheses in balanced pairs are allowed.

To create a macro, follow these steps:

1 Create the macro using:

```
DEFine <macroname> = (
```

Macro names can be up to 14 characters long; the first character must be alphabetic. Names longer than 14 characters are truncated. The macro service does not distinguish between upper- and lowercase letters in macro names.

For example:

```
DEFine start = (
```

The name of this macro is "start", the left parenthesis indicates the beginning of the macro.

After pressing the Return key, the Macro: prompt appears on the next line.

2 Enter the desired commands at the Macro: prompt.

For example, the following commands request a connection to "host1," pause for a second to give time for the host to respond with a login request, transmit the user's name as the login name, and transmit the password:

```
Echo “connection”
Connect host1 ECM
PAuse 1
TRansmit “terry”
PAuse 1
TRansmit “<password>”
RESume
```

The text of the macro must conform to the conventions for assigning strings described in *New Installation for NETBuilder II Software*.

The Break key or the character specified by the -TERM BReakChar parameter can be used to cancel the DEFine command at any time before the terminating right parenthesis is entered.

3 Complete the macro by entering the right parenthesis.

The normal NETBuilder command prompt returns.

4 Suppress the display of the macro as the macro is executed, and make sure the user can terminate the macro execution with the Break key using:

```
SETDefault !<config file> -TERM InterActTerm = (NoMacroEcho,
  MacroBreak)
```

NoMacroEcho and MacroBreak are the default settings of the -TERM InterActTerm parameter and may not need to be set.

Assigning the Macro to a Configuration File

After defining the macro, you need to assign it as a port-initialization macro and give it a configuration file number. To assign the macro to a configuration file, use:

```
SETDefault !<config file> -TERM InitMacro = “start”
```

Valid configuration files numbers are 1 and 3 through 32. Configuration file 2 is the default for outgoing connections and must not be used for incoming connections.

This command assigns the macro named “start” as a port-initialization macro to a configuration file. When the PAD-attached terminal user initiates an incoming automatic connection using the specified configuration file, the gateway selects the next available port, initializes the port and session with the contents of configuration file, changes the port from listen mode to command mode, executes the initialization macro, and makes the connection request to host1.

Managing Macros

Use the following commands to delete and display macros, and to flush the macro cache:

- To delete a macro, use:

```
UNDefine <macroname>
```

In this command, <macroname> is the name of the macro to be deleted.

- To display all the defined macros on the gateway, enter:

  `SHow -SYS MACros`

- To display the contents of a specific macro, use:

```
SHow -SYS MACros <macroname>
```

- To remove all macros from the macro cache, enter:

  `FLush -SYS MACros`

Name Service for TCP/IP Connections

Because users more easily remember names instead of addresses, the X.25 connection service software allows you to assign names to IP addresses. Using names also helps users associate a network resource with its function and allows users to connect to resources without knowing their network addresses.

Name and corresponding address information is maintained in a database provided by the name service. This service maintains and updates information regarding resource names and addresses, and responds to queries regarding names. The network manager decides which one of the following two name services to use:

- IEN116

  Allows you to use a database maintained on the gateway disk to add, remove, and change the names of network resources. The IEN116 name service can be stored on any gateway or terminal server with an internal diskette.

- Domain

  Allows the gateway to use, but not provide, the Domain name service. The network must include a Domain name server that responds to Domain name requests from the gateway. The Domain name service is more widely used than IEN116.

If your gateway is installed on a network that is already in operation, the name service probably has already been defined. Set up your gateway to use the existing name service. If you are setting up a new network, you need to select the name service for it.

If you plan on using names in incoming automatic or extended connection requests to TCP/IP hosts, you need to:

- Select either the IEN116 or the Domain name service.
- Select the address of the primary and/or secondary name servers.
- Assign names to network addresses or resources.

To configure the gateway for IEN116 name service, refer to the next section. To configure the gateway for Domain name service, refer to "Domain Name Service" on page 50-11.

IEN116 Name Service

The IEN116 name service allows you to use a database maintained on the gateway diskette to add, remove, and change the names of network resources from your gateway. The IEN116 name service can be stored on any server with an internal diskette and is called the primary name server. If your gateway boots from an internal diskette, the default name server is the gateway itself. You can optionally specify a secondary name server to be used if the primary name server is unavailable.

Configuring the Gateway. To configure the gateway to use the IEN116 name service, follow these steps:

1 Configure IEN116 as the name service type by entering:

```
SETDefault -IPName NameServiceType = Ien
```

The default value for the NameServiceType parameter is Domain.

2 Specify the addresses to be used as the primary (and optionally, the secondary) name servers using:

```
SETDefault -IPName PrimaryNameServer = <primarynameserver address>
```

To specify the address of the secondary name server, use:

```
SETDefault -IPName SecondaryNameServer = <secondarynameserver
  address>
```

3 Assign Internet names to network addresses using:

```
ADD -IPName NAME <Internet name> <network address>
```

Internet names consist of simple character strings. The maximum length of an Internet name is 40 alphanumeric characters. The first character must be alphabetic, and subsequent characters can be either alphabetic or numeric. The only nonalphanumeric characters allowed are the underscore (_), the period (.), and the dash (–). Internet names assigned with the IEN116 name service are case-sensitive; if capital letters are used as part of the name, a user must enter the name exactly as assigned to connect to that name successfully.

To add names, the gateway must boot from its own diskette, and the gateway itself must be used as either the primary or the secondary name server. Otherwise, the ADD -IPName NAME command has no effect.

4 Confirm that the new name has been successfully added.

For example, to display the name "Finance", enter:

```
SHow -IPName NAME Finance
```

The newly created name "Finance" should be displayed, along with its corresponding Internet address.

Displaying a List of Names. To display a list of existing network names, enter:

```
SHow -IPName NAME
```

The gateway queries both the primary and secondary name servers, and displays the names available on each.

Displaying the Internet Address for a Specific Name. To display the Internet address assigned to a specific name, use:

```
SHow -IPName NAME <Internet name>
```

Deleting an Internet Name. To delete an Internet name, use:

```
DELete -IPName NAME <Internet name>
```

To delete names, the gateway must boot from its own diskette, and the gateway itself must be used as either the primary or the secondary name server. Otherwise, the DELete -IPName NAME command has no effect.

Domain Name Service

Because the gateway does not implement the server side of the Domain name service (but does implement the client side), your network must include a Domain name server. The Domain name server responds to Domain name requests from the gateway. All names must be added and removed from the Domain name database on the Domain name server.

If you select Domain name service, you must specify a primary name server if your gateway boots from an internal diskette. You can optionally specify a secondary name server to be used when the primary name server is unavailable.

Configuring the Gateway. To configure the gateway to use the Domain name service, follow these steps:

1 Confirm that the NameServiceType parameter is set to the default value, Domain, by entering:

```
SHow -IPName NameServiceType
```

If the value is not set to Domain, set the value by entering:

```
SETDefault -IPName NameServiceType = Domain
```

2 Specify the string for the DomainName parameter using:

```
SETDefault -IPName DomainName = “<name>”
```

The domain name can be no more than 128 characters. The domain name resolver appends the domain string to each name request that does not include a period.

3 Specify the primary name server address using:

```
SETDefault -IPName PrimaryNameServer = <address>
```

You can specify the secondary name server address using:

```
SETDefault -IPName SecondaryNameServer = <address>
```

4 Assign names to the resources on your network from the name server.

Consult the name server documentation for details on how to do this.

5 Confirm that a new name has been successfully added using:

```
SHow -IPName NAME <value>
```

Using the Name Cache. The gateway uses a storage area in memory, called the *name cache*, where some previously used names and their corresponding addresses are stored. The name cache allows the gateway to recall domain names quickly and reduces the data traffic load on the network. When a domain name request is made, the gateway searches its name cache first. If the information is not found, the resolver refers the name request to the primary name server.

If the Network Manager modifies any of the domain names or the information database is otherwise corrupted, discrepancies can exist between the information stored in the local cache and in the Domain name server. To correct these discrepancies by deleting cache information, enter:

```
FLush -IPName CAChe
```

This command deletes all entries in the cache. In addition, the cache is cleared automatically whenever the DomainName parameter is changed.

To display the contents of the name cache, enter:

```
SHow -IPName CAChe
```

Configuring Rlogin Connections

The RLOGin command can be used in incoming extended TCP connections to a specified IP Internet-attached host that is using the Rlogin protocol. Because Rlogin supports passing on additional information during session establishment, you may need to do more configurations.

The 3Com implementation of Rlogin supports the Rlogin client only. The client username (username on the client side), server username (username to be used for login on the Rlogin server side), terminal type, and baud rate are communicated to the server during the connection setup. The number of rows and columns also can be communicated to the server if the server requests the information.

To configure for Rlogin connections, obtain Network Manager privilege follow these steps. Depending on your configuration, some changes may be optional.

1 Set the terminal type to be used for the Rlogin connection using:

```
SETDefault !<configfile> -TERM TERMType = “<string>”
```

Use configuration file 1 for incoming connections. If you need to customize a configuration file for a specific host, use configuration files 3 through 32. You must not use configuration file 2 because it is the default for outgoing connections. If you use an even-numbered configuration file for an incoming connection, make sure that you change the DeVice parameter from Host to Terminal using the SETDefault !configfile -TERM DeVice = Terminal command or the connection attempt will fail.

The gateway transmits the string to the Rlogin server when the RLOGin command is used. The TELnet command also uses this string, which has a maximum of 40 characters.

2 To set the number of columns and rows for the terminal, use:

```
SETDefault !<configfile> -TERM COLumns = <number> (1-255)
SETDefault !<configfile> -TERM ROW = <number> (1-255)
```

3 To specify that the gateway send an empty string for the client username to the destination during the connection negotiation, enter:

```
SETDefault -TCPAPPL RLogSendName = No
```

With this setting, the user is usually prompted for a password to log on to the remote host.

The default for this value is "yes." If this value is used, the client username is automatically sent to the destination host.

Name Service for OSI Connections

Using names in connection requests helps users to associate a network resource with its function. Also, connections to resources can occur without users knowing the network addresses of the resource. Because incoming automatic connections to OSI hosts are identified by names and not addresses, you must configure the gateway to assign names to presentation service access point (PSAP) addresses to use OSI name services. However, extended connections and incoming automatic connections (which use the configuration file) can use addresses.

Name and corresponding address information is maintained in a database provided by the name service or directory service. The name service maintains and updates information regarding resource names and their corresponding addresses and responds to queries regarding names. The gateway supports the following two types of name services:

- X.500 directory service

  Allows you to use a database called the Directory Information Tree (DIT) which is maintained on a computer that runs the X.500 protocol. With this database, you can add, remove, and show the names and addresses of network resources from the gateway.

- File-based name service

  Stores the database on the gateway diskette. By default, a gateway that boots from its own diskette stores the database on the gateway diskette.

Figure 50-1 shows these two name services.

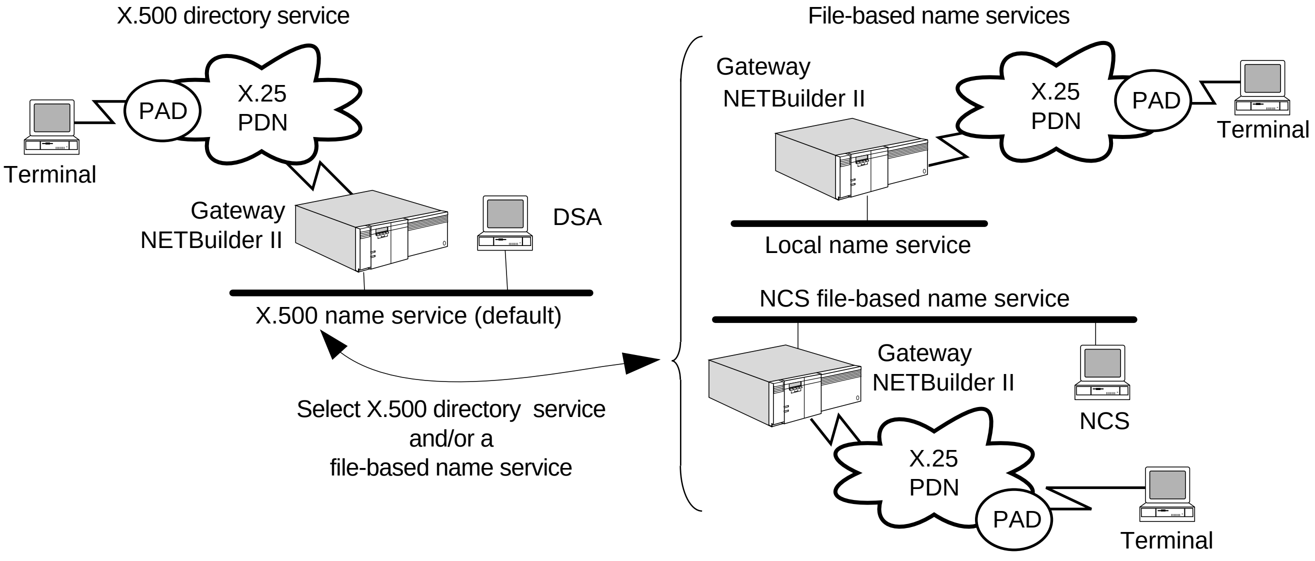


Figure 50-1 Name Services Supported for Incoming OSI Connections

You can use one or both name services. If you use both name services, one name service first attempts to resolve a name request from the gateway and if that fails, the other name service attempts to resolve it. You can configure the order of this operation by entering:

```
SETDefault -OSIAPPL NameSourceOrder
```

When selecting the name service, remember that if you use a name service that has its database stored on another computer, the same database can be used for multiple servers. If you store the name service on the local diskette, a separate name service must be set up for each server.

If you have a computer that can support X.500 directory service, use the X.500 directory service.

To configure name services for incoming OSI connections, you need to:

- Select either the X.500 or the file-based name service, or both.
- Determine the name resolution order, if both name services are used.
- Assign names to resources.

To configure the gateway to use the X.500 directory service, refer to the next section. To configure the gateway to use the file-based name service, refer to “File-Based Name Service” on page 50-22.

X.500 Directory Service

If you do not have a computer on your network that supports the X.500 protocol, skip this section and continue to “File-Based Name Service” on page 50-22.

The X.500 directory service allows you to use a database called the Directory Information Tree (DIT), which is maintained on a computer that runs the X.500 protocol. With this database, you can add, remove, and show the names and addresses of network resources from the gateway. Additional information about the DIT is provided in “Adding Names to Resources with the Directory Manager” later in this chapter.

The computer running X.500 contains the Directory System Agent (DSA) and the Directory Information Base (DIB) database. The DSA maintains the DIB and interfaces with the Directory User Agent (DUA) that runs on the gateway. The DSA processes the DUA operation requests, such as “add a name” or “delete a name” as shown in Figure 50-2.

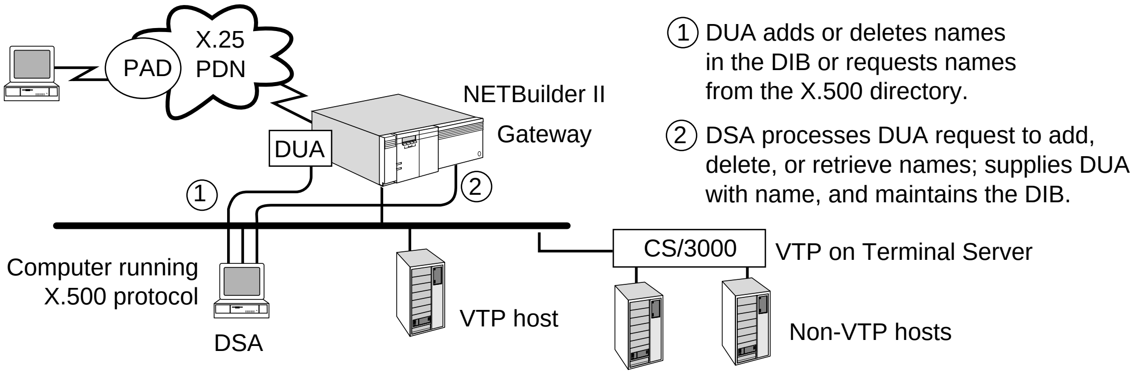

X.25
PAD
PDN
NETBuilder II
Gateway
DUA
1
2
Computer running
X.500 protocol
DSA
VTP host
CS/3000
VTP on Terminal Server
Non-VTP hosts
1 DUA adds or deletes names in the DIB or requests names from the X.500 directory.
2 DSA processes DUA request to add, delete, or retrieve names; supplies DUA with name, and maintains the DIB.

Figure 50-2 DUA and DSA Interaction

The DUA sends a request to the DSA when it needs the DSA to resolve a name to a presentation address during a VTp <name> or Connect <name> command (available in incoming extended connections), or when you want to access the DIB using the DirectoryManage command.

Configuring the Gateway. To configure the gateway for X.500 directory service, follow these steps:

1 Confirm that the NameSourceOrder parameter includes X.500 in its values by entering:

```
SHow -OSIAPPL NameSourceOrder
```

If you want to use both name services, include both names and the order in which name requests should be resolved.

For example, to include both X.500 and file-based name services and to specify that the X.500 directory be queried first, enter:

```
SETDefault -OSIAPPL NameSourceOrder = X500 File
```

If you want to use only the X.500 directory service, enter:

```
SETDefault -OSIAPPL NameSourceOrder = X500
```

2 Specify the address of the DSA using:

```
SETDefault -OSIAPPL DSAAddress = <PSAP address>
```

You may need to change the address of the DSA first. To accomplish this, disconnect the gateway from its current DSA by entering:

```
UnBindDSA
```

Set the address of the new DSA using:

```
SETDefault -OSIAPPL DSAAddress = <PSAP address>
```

The DUA-DSA connection is transparent to the user. The connection occurs by either an operation request to the DSA or an incoming extended connection attempt made with the VTp <name> or the Connect <name> command.

3 Select a DSA vendor by entering:

```
SETDefault -OSIAPPL DSAType = Standard
```

4 Use the DirectoryManage command to add names of resources on your network.

The DirectoryManage command uses a menu system to add directory names. For information on using DirectoryManage and completing the configuration procedure, refer to the next section.

Managing Entries in the DIB. The menu-driven DirectoryManage command allows you to add, remove, and show entries in the DIB. Each entry in the DIB is made up of attributes. These attributes depend on the object class the entry describes. Examples of object classes are “Country” or “Person.” Attributes of the object class “Person” could be “Name,” “Social Security Number,” and “Address.” For example, a typical entry belonging to the object class “Person” could be:

{Name = John Doe, SS# = 543-45-4333, Address = 324 Bayfront Ave., Santa Clara}

The gateway supports the following object classes: Country, Organization, OrganizationalUnit, ApplicationProcess, and ApplicationEntity. Their respective attributes are CountryName, OrganizationName, OrganizationUnitName, CommonName, and PresentationAddress.

Entries in the DIB are arranged in the DIT. Figure 50-3 and Figure 50-4 show the tree structure and how it applies to a directory name. The position of the object classes in the tree reflects their hierarchical relationship. Country is highest in the tree, followed by Organization, OrganizationalUnit (up to 3 levels are allowed), ApplicationProcess, and ApplicationEntity. This hierarchy must always be respected when configuring a DUA operation.

A leaf entry, which is an entry without any entries below it, is the only type of entry that can be added or deleted. For example, in the entry {CountryName US, OrganizationName 3Com}, US is not a leaf entry because 3Com is below it; therefore, it cannot be deleted. 3Com is a leaf entry and can be deleted.

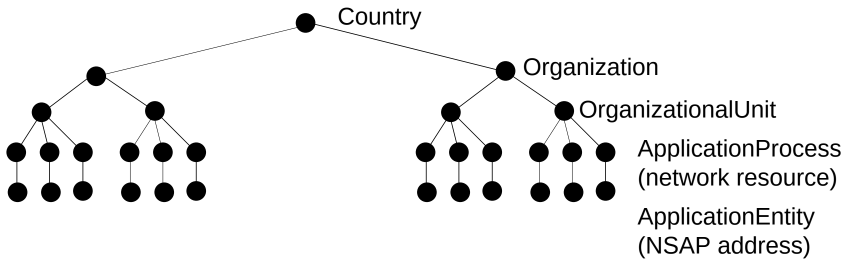


Figure 50-3 Directory Information Tree (DIT)

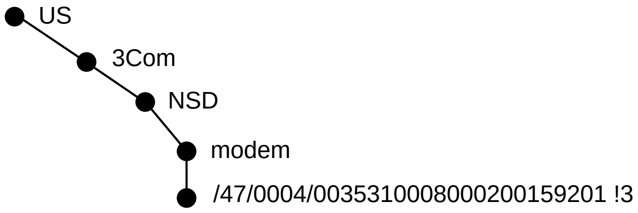


Figure 50-4 Directory Name in DIT Format

The object classes are defined as follows:

- Country

  Common to all directory names in the same directory. In the example in Figure 50-4, country is defined as US, so this is the first part of all names in this directory. The country name must consist of two characters.

- Organization

  A maximum of 14 characters is allowed. In the example, 3Com is the organization name.

- OrganizationalUnit

  Up to three levels of organizational units can exist in the DIB. In the example, NSD is an organizational unit. The OrganizationalUnit name cannot be more than 14 characters. Figure 50-5 shows a directory with three levels of organizational units.

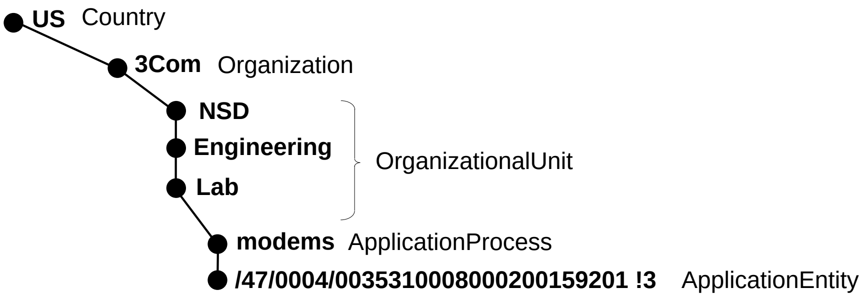

US Country
3Com Organization
NSD
Engineering
Lab
OrganizationalUnit
modems ApplicationProcess
/47/0004/0035310008000200159201 !3 ApplicationEntity

Figure 50-5 Directory Name with Three Organizational Units

- ApplicationProcess

  CommonName is the attribute of the ApplicationProcess object class; it refers to the name of the network resource. In Figure 50-5, modems is an application process.

- ApplicationEntity

  The address that corresponds to the resource name, or presentation address, is the attribute of the ApplicationEntity object class. See Figure 50-5 for an example of an application entity.

Adding Entries. To add an entry to the X.500 directory, select the Add Name option from the Directory Manager Menu, and then enter the directory name.

To assign the name “C=US O=3Com OU=NSD CN=modems” to the PSAP address of the gateway port to which modems are attached, follow these steps. It assumes that no country name has been defined in the directory name database.

1 Specify US as the country name by doing the following:

a To invoke the Directory Manager menu, enter:

`DirectoryManage`

The following main menu is displayed:

```
-------------Directory Manager Menu------------
1.- Add name
2.- Delete name
3.- List allnames
4.- Print one VT name
5.- Set user name and password
6.- Set default DN
```

b Select option 1, Add name.

The following submenu is displayed:

```
-------------Directory Manager Menu------------
No Default Distinguished Name
No UserName
1.- Country
2.- Organization
3.- OrganizationalUnit
4.- ApplicationProcess
5.- ApplicationEntity
6.- Do Add Request
```

Remember to follow the hierarchy and only add leaf entries. You must add one level at a time, starting from the top. For example, to add the entry {CountryName US, OrganizationName 3Com, OrganizationalUnitName Engineering}, you must first add the entry {CountryName US} to the DIB. You then add the entry {CountryName US, OrganizationName 3Com}, and finally the entry {CountryName US, OrganizationName 3Com, OrganizationalUnitName Engineering}.

c Select 1 to specify the country name.

d For the country name, enter:

```
US
```

e Select 6, Do Add Request, to add the country name.

If a country name has already been specified in the database, a message appears.

2 Specify the organization name, 3Com, by following these steps:

a Select 1 from the Directory Manager submenu and when prompted for the country name, enter:

```
US
```

b Select 2 from the Directory Manager submenu.

c For the organization, enter:

```
3Com
```

Up to 14 characters can be entered for the organization name.

d Select 6, Do Add Request, to add the name C=US O=3Com.

3 Specify the organizational unit, NSD, by following these steps:

a Select 1 from the Directory Manager submenu and enter `US` when prompted for the country name.

b Select 2 from the Directory Manager submenu and when prompted for the organization name, enter:

```
3Com
```

c Select 3 from the Directory Manager menu.

d For the organizational unit name, enter:

```
NSD
```

e Select 6, Do Add Request, to add the name C=US O=3Com OU = NSD.

4 Specify the Application Process, modems, by following these steps:

a Select 1 from the Directory Manage submenu and when prompted for the country name, enter:

```
US
```

b Select 2 from the Directory Manager submenu and when prompted for the organization, enter:

```
3Com
```

c Select 3 from the Directory Manager submenu and when prompted for organizational unit, enter:

```
NSD
```

d Select 4 from the Directory Manager submenu.

e At the `CommonName` prompt, enter:

```
modems
```

A maximum of 14 characters can be specified for the CommonName attribute.

f Select 6, Do Add Request, to add the name C=US O=3Com OU=NSD CN=modems.

5 Specify the Application Entity by following these steps:

a Select 1 from the Directory Manager submenu and when prompted for the country name, enter:

```
US
```

b Select 2 from the Directory Manager submenu and when prompted for the organization name, enter:

```
3Com
```

c Select 3 from the Directory Manager submenu and when prompted for the organizational unit name, enter:

```
NSD
```

d Select 4 from the Directory Manager submenu and enter when prompted for the CommonName of the application process.

```
modems
```

e Select 5 from the Directory Manager submenu.

f Enter the PSAP address for modems that is connected to gateway 2:

```
/47/0004/0035310008000200159201
```

g Select option 6, Do Add Request, to add the name C=US O=3Com OU =NSD CN=modems PA=/47/0004/0035310008000200159201.

Displaying Directory Names. Use the DirectoryManage command to either display all names or display the address of a particular name.

To display all names, follow these steps:

1 Select option 3, List all names, from the Directory Manager main menu.

2 Specify Country, Organization, OrganizationUnit and Filtered Application Process, if necessary.

If you are using a Default Distinguished Name, refer to "Setting Up the Default Distinguished Name" section on page 50-21.

3 Select option 5, Do list request.

To display the address of a particular name, follow these steps:

1 Select option 4, Print one VT name.

2 Specify the parts of the name.

3 Select option 5, Do print request.

Deleting Entries. Only leaf entries are allowed to be deleted, meaning that the DSA deletes a DIT entry from bottom to top. For example, to delete {CountryName US, OrganizationName 3Com, OrganizationalUnitName Engineering}, first provide {CountryName US, OrganizationName 3Com, OrganizationalUnitName Engineering} to the DSA, and the DSA will delete OrganizationalUnitName Engineering. The DIT now contains only {CountryName US, OrganizationName 3Com}. Next, have the DSA delete {CountryName US, OrganizationName 3Com}, and the DSA will delete the leaf entry OrganizationName 3Com. You then provide the entry {CountryName US} to be deleted.

To delete a name from the directory name database, follow these steps:

1. Select option 2, Delete name, from the Directory Manager main menu.

   For example, to delete the name C=US O=3Com OU=NSD, select option 2, Delete name, from the Directory Manager main menu. The following menu is displayed:

```
-------------Directory Manager Menu------------
No Default Distinguished Name
No UserName
1.- Country = US
2.- Organization = 3Com
3.- OrganizationalUnit = NSD
4.- ApplicationProcess
5.- ApplicationEntity
6.- Do Delete request
```

2. Select option 6, Do Delete request.

   The name C=US O=3Com OU=NSD is deleted from the database.

Setting the User Name and Password. To set up the user name and password when you do a DSA operation request, follow these steps.

1. Select option 5, Set user name and password, from the Directory Manager main menu.

   The following submenu is displayed:

```
-------------Directory Manager Menu------------
No Default Distinguished Name
No UserName
1.- Country
2.- Organization
3.- OrganizationalUnit
4.- Person
5.- Save user name and password
```

2. Specify Country, Organization, and OrganizationalUnit names by selecting options 1, 2, and 3, respectively.
3. Select option 4, Person.

   The Common Name prompt is displayed.

```
CommonName:
```

4. Enter the user name. A maximum of 14 characters is allowed.

   For example, enter the user name John. The User Password prompt is displayed:

```
UserPassword:
```

5 Enter the password.

For example, enter the password Guest.

6 Select option 5, Save user name password.

The new user name and password is displayed at the top of the screen as:

```
UserName: CN = John        Password = Guest
```

Setting Up the Default Distinguished Name. The following procedure describes how to set up a default distinguished name (DN) so that when you do a DSA operation request, or delete or list names, you do not have to retype certain fields of names that remain constant.

For example, the Country, Organization, and OrganizationalUnit of a directory name are often common to all devices in the same network or subnetwork. To avoid having to define them every time you access the DSA, you can specify a default name called a default DN that contains all three of them. After you define a default DN, only the parts not defined in the DN need to be defined whenever a new name is added.

The default DN must first be added to the database before it can be used as the default distinguished name for all name requests. For example, if you want to define a DN in which US is the CountryName, 3Com is the OrganizationName, and Finance is the OrganizationalUnitName, you must first add this name to the directory. Save US3ComFinance as the default DN. You need to then specify only the Application Entity and Application Process attributes when accessing the DSA, and the default DN is automatically added to these attributes.

Follow these steps to set the name you just added to the database as the default DN. To record the default DN on the gateway only, follow these steps:

1 Select 6, Set Default DN, from the main menu.

The following submenu is displayed:

```
-------------Directory Manager Menu------------
No Default Distinguished Name
No UserName
1.- Country
2.- Organization
3.- OrganizationalUnit
4.- Save default DN
```

2 Enter the name you just added to the database:

a Select 1 from the menu and specify **`US`** as the country.

b Select 2 from the menu and specify the organization by entering:

`3Com`

c Select 3 from the menu and specify the organizational unit, by entering:

`Finance`

d Select 4 to save the default DN and press the Return key.

The following information is displayed:

```
-------------Directory Manager Menu------------
Default DN: Country=US Org=3Com OrgUnit=Finance
No UserName
1.- Country = US
2.- Organization = 3Com
3.- OrganizationalUnit = Finance
4.- Save default DN
The new default DN is displayed at the top of the screen.
```

File-Based Name Service

You can use a file-based name service in addition to, or instead of, the X.500 directory service. A file-based name service stores the name service on the gateway diskette.

Configuring the Gateway for File-Based Name Service. To configure the gateway to use the file-based name service, follow these steps:

1. Confirm that the NameSourceOrder parameter includes file-based in its values by entering:

   ```
   SHow -OSIAPPL NameSourceOrder
   ```

   If you want to use both the file-based and X.500 name services, include both names and the order in which name requests should be resolved.

   For example, to include both file-based and X.500 name services, and to specify that the file-based name service be queried first, enter:

   ```
   SETDefault -OSIAPPL NameSourceOrder = File X500
   ```

2. Add names to physical addresses in the file-based name database.

   For example, to assign the name “gate” to the PSAP address /47/0004/00351100080002013C3701!1.128, enter:

   ```
   ADD -OSIAPPL NAme gate /47/00004/00351100080002013C3701!1.128
   ```

   You can use this command to add names to the database on the gateway diskette.

   The name can be no more than 14 characters and must start with a letter. Characters that can follow the first letter are a letter, a digit, or one of the following symbols: underscore (_), the period (.), or the at sign (@). All other characters are ignored.

   Optionally, you can delete names using the DELete -OSIAPPL NAme <name> syntax.

3. To confirm that the new name has been successfully added, use:

   ```
   SHow -OSIAPPL NAme <name>
   ```

Displaying Names. To display names currently stored on the gateway diskette in the filename database, enter:

```
SHow -OSIAPPL NAme
```

How the Incoming Connection Service Works

The X.25 connection service gateway allows X.25 PAD-attached terminals to connect to IP Internet-attached Telnet or Rlogin hosts, OSI-based hosts, and to hosts attached to host ports on a communications server that supports Telnet, Rlogin, or the Virtual Terminal Protocol (VTP). WAN-to-LAN connections are also referred to as *incoming connections* and are controlled by the gateway's incoming connection service. Figure 50-6 is an example of WAN-to-LAN connections.

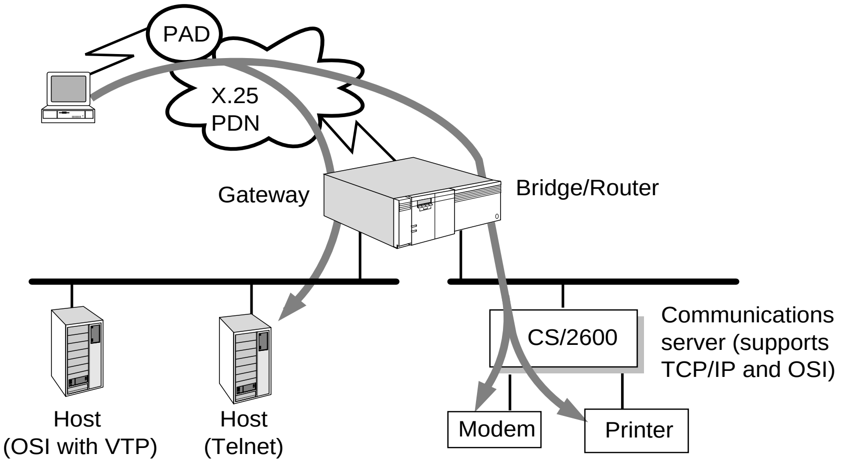


Figure 50-6 WAN-to-LAN Connections (Incoming)

The X.25 connection service gateway offers two type of incoming connections:

- Automatic (one-step)

  End users can enter a connection command from the X.25 PAD-attached terminal, and the gateway automatically establishes the connection to the Telnet, Rlogin, or OSI server.

- Extended (two-step)

  End users can enter a connection command from the X.25 PAD-attached terminal and establish a connection to the gateway user interface, the same interface that is seen when connecting to the local console of the 3Com router or connecting through Telnet or through OSI VTP. After connecting to the gateway user interface, users can make connections to Telnet, Rlogin or OSI servers. Users with Network Manager privilege can also configure, manage, and monitor the system.

With incoming automatic calls, you can connect only to a single host. With incoming extended calls, you can connect to multiple hosts and establish up to eight sessions per port.

51 CONFIGURING LOCAL ACCESS CONTROL

This chapter describes how to configure access control to regulate user access to the NETBuilder bridge/router. You can access the bridge/router either through a console port or Telnet connection, or through the gateway.

During incoming extended connection requests from X.25 packet assembler/disassembler (PAD)-attached terminals to Internet Protocol (IP) Internet-attached Telnet and Rlogin hosts, as well as Open Systems Interconnection (OSI)-based hosts, limiting access is crucial. Other types of access control can be implemented to prevent or restrict remote access to the gateway and to force users to log on to Rlogin servers. This chapter describes how to configure local access control, log on and log out, change user passwords, and control Rlogin connections.

Configuring Local Access Control

Local access control requires a user to specify a user name and a password before entering commands. You can use a user account name given to you by your network administrator or you can access the NETBuilder bridge/router by using the user account name, “root.”

Procedure

To configure access control, follow these steps:

1 Enable local access control, via a console port or Telnet connection, by entering:

```
SETDefault -AC RESolutionOrder = Local
```

2 For X.25 PAD users to enable local access control through the gateway, enter:

```
SETDefault -AC CONTrol = Enable
```

The default setting for access control is Enable, so this parameter may already be set.

3 Assign a user account name by entering the AddUser command.

a At the prompt, enter the user’s account name.

The account name is limited to 15 characters.

b Enter the user’s full name at the next prompt.

The full user name is limited to 23 characters.

4 Assign the level of access by entering the user’s maximum access privilege (Max. Privilege). Enter NetMgr or NM for net manager privileges or “User” or “U” for user privileges.

5 Enter a password for the user.

The password is limited to 15 characters and is case-sensitive. For security reasons, passwords are not echoed on the screen.

When prompted, reenter the password.

You can also use the menu-driven UserManage command to add user names and passwords. The DELeteUser command removes user accounts from the database. For more information on these commands, refer to Chapter 1 in *Reference for NETBuilder Software Version 9.3.*

You can set other parameters that apply to local access control, such as EXPirationTimer. For more information, refer to Chapter 2 in *Reference for NETBuilder Software Version 9.3.*

Related Information

Local access control can prevent users from logging on to the router, but it cannot prevent users from accessing the router remotely using the REMote command. You can prevent users from accessing your router remotely, or you can restrict specific users' remote access to your router.

Logging On and Logging Out (in the CX package)

When an incoming X.25 extended connection is made to the gateway, the user must log on before entering commands if local access control is enabled.

To log on and log out, follow these steps:

1 Log on at the NetLogin prompt by entering the user name assigned to you by the network manager.

The gateway prompts for a password.

User account names and passwords are case-sensitive.

2 Enter the password assigned by the network manager.

After you enter the correct password, the following command prompt is displayed:

```
NETBuilder>
```

3 Log out by entering the LOGout command.

This command disconnects all sessions on a port and requires the user to log on again before entering commands.

Changing User Passwords

Users can change their passwords.

Passwords are case-sensitive, and must be entered exactly as they were assigned.

To change the password, follow these steps:

1 Enter the PassWord command.

The gateway prompts you for the old password.

2 Enter your old password.

The gateway prompts you for your new password.

3 Enter your new password.

For security reasons, passwords are not displayed on the screen and cannot be viewed by the network manager.

52 MANAGING SESSIONS FOR INCOMING EXTENDED CALLS

This chapter describes how to make connections from the gateway's user interface to Internet Protocol (IP) Internet-attached Telnet and Rlogin servers, and to Open Systems Interconnection (OSI) hosts. When the connection has been made, you can manage the session, establish and manage multiple sessions, and disconnect sessions. It is assumed that you have already made an incoming extended connection from an X.25 packet assembler/disassembler (PAD)-attached terminal to the gateway user interface. For information on making an incoming extended connection, refer to Chapter 50.

This chapter also describes procedures for making connections to different types of network resources, and provides general information on how to manage sessions, including moving between sessions and disconnecting sessions.

For conceptual information, refer to "Managing Sessions" on page 52-10.

Making Connections to IP Internet-attached and OSI Hosts

After establishing a connection from the X.25 PAD-attached terminal to the gateway user interface, you can use different commands to establish connections to IP Internet-attached hosts and to OSI-based hosts. The specific commands are listed in Table 52-1; their availability depends on the protocols being run.

Table 52-1 Establishing Connections to IP Internet-attached and OSI Hosts

Setup	Step	Command
TCP/IP *or* OSI connections	Connect to a host.	`Connect <name> | <address>`
TCP/IP connections	Connect to a host using the Telnet protocol.	`TELnet <name> | <address>`
	Connect to a host using the Rlogin protocol.	`RLOGin <name> | <address>`
OSI connections	Connect to an OSI host.	`VTp <name> | <psapaddress>`
Troubleshooting	Check the status of a TCP/IP destination.	`PING <address>`
	Check the status of an OSI destination.	`OPING <nsapaddress>`

The sections and procedures that follow explain the meaning of each command in detail and give examples.

Making Connections with the Connect Command

You can use the Connect command to make connections to most resource types on the network, except for Rlogin connections.

To make a connection using the Connect command, follow these steps:

1. Connect to the desired resource.

   - To connect to an IP address, use:

     ```
     Connect <IP address>
     ```

   - To connect to an OSI PSAP address, use:

     ```
     Connect <OSI PSAP address>
     ```

     For more information regarding presentation service access point (PSAP) addresses and 3Com's OSI address conventions, refer to Appendix E.

   For example, to connect to a resource named "marketing," enter:

   ```
   Connect marketing
   ```

   To complete the connection, the name "marketing" must be an IP name in a Transmission Control Protocol/Internet Protocol (TCP/IP) environment, or a name in an OSI environment.

   The syntax for the Connect command can vary greatly, depending on the resource being accessed. For more information on the different syntax possibilities, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

   You can also specify the gateway to make a connection, then immediately reenter command mode (allows you to enter other connection service commands and is indicated by the NETBuilder prompt). To enter command mode, type the letters "ECM" after the address or name of the resource as shown in the following example:

   ```
   Connect marketing ECM
   ```

   If the resource you are accessing accepts the connection, the gateway sends the following message to the terminal:

   ```
   session n -- connected to marketing
   ```

   where *n* refers to the number of the session.

   If the connection does not succeed, you will receive an error message. A connection attempt can fail for a number of reasons. For more information, refer to "Troubleshooting Connection Error Messages" on page 52-7. For information on the command mode and other modes of operation, refer to "Establishing a Single Session" on page 52-10.

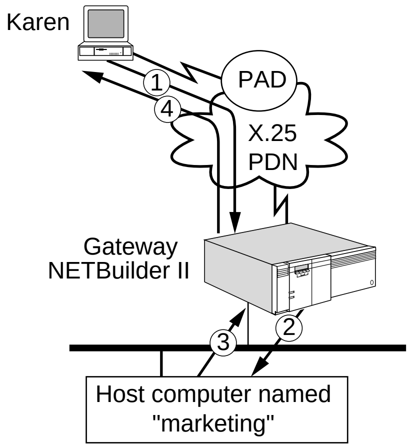
2. When you have reached the resource, perform the actions appropriate for the resource application.

3. To disconnect from the session, log out from the host.

   You can also enter the enter command mode (ECM) character, and then the DisConnect command, from the NETBuilder prompt.

4. To disconnect from the gateway and return to the PAD terminal prompt, use the LOGout or LIsten command.

Figure 52-1 shows how a connection is established between a terminal user named "Karen" at the PAD-attached terminal and the host computer named "marketing."



① Karen enters: **connect marketing**

② The gateway requests a connection to marketing.

③ Marketing accepts the connection.

④ The gateway sends Karen the message:
session <n> -- connected to marketing

Figure 52-1 Establishing a Connection with the Connect Command

Making Telnet Connections to TCP/IP Resources

The TELnet command can be used to make Telnet connections to TCP/IP resources on the network.

To make a connection with the TELnet command, follow these steps:

1 Connect to the desired resource.

- To connect to an Internet address, use:

```
TELnet <Internet address>
```

- To connect to a specific resource, use:

```
TELnet <resource name>
```

The syntax for the TELnet command can vary depending on the resource being accessed. For more information on the different syntax possibilities, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

You can also specify the gateway to make a connection, and then immediately reenter command mode (allows you to enter other connection service commands and is indicated by the NETBuilder prompt).

To enter command mode, type the letters “ECM” after the address or name of the resource as shown in the following example:

```
TELnet finance ECM
```

The TELnet command makes a TCP connection to the specified host (or another server) using the Telnet protocol. If the resource you are accessing accepts the connection, the gateway sends the following message to the terminal:

```
session n -- connected to finance
```

where *n* refers to the number of the session.

If the connection does not succeed, you will receive an error message. A connection attempt can fail for a number of reasons. For more information, refer to “Troubleshooting Connection Error Messages” on page 52-7.

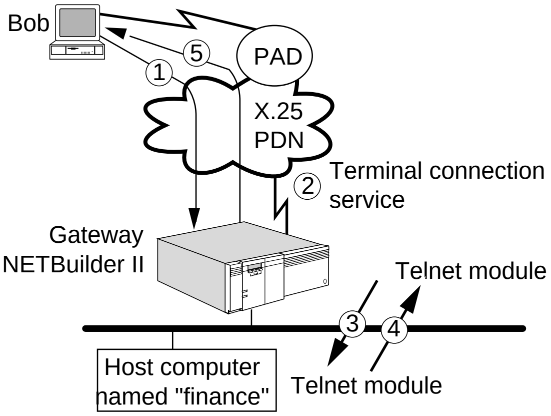
2 When you have reached the resource, perform the actions appropriate for the resource application.

3 To disconnect from the session, log out from the host.

You can also enter the ECM character, and then the DisConnect command, from the NETBuilder prompt.

4 To disconnect from the gateway and return to the PAD terminal prompt, use the LOGout or LIsten command.

Figure 52-2 shows how a connection is established using the TELnet command between a user named "Bob" at the PAD-attached terminal and a host computer named "finance."



① Bob enters the command: **telnet finance**.

② The terminal connection service on the gateway looks up the name "finance" and matches an address to the name.

③ The Telnet module running on the gateway notifies the Telnet module running on "finance" of the connection request.

④ The Telnet module on "finance" tells the Telnet module on the gateway that the connection is open.

⑤ The gateway sends Bob a message: session <n> -- connected to finance.

Figure 52-2 Establishing a Connection with the TELnet Command

Making Rlogin Connections to Resources

To make Rlogin connections, the remote host being accessed must be running Rlogin, a UNIX environment protocol. You can access remote UNIX hosts in addition to any target host that is running the Rlogin protocol.

The RLOGin command is similar to the TELnet command, but is used in a slightly different way. RLOGin differs from the TELnet command as follows:

- When making a connection with RLOGin, the client terminal always communicates the terminal type, baud rate, and user name to the host. In some cases, the client terminal may also communicate the number of rows and columns on the terminal.
- RLOGin allows the host to enable and disable flow control on the session. For example, if the client terminal is running an application where the [Ctrl]+S character has a specific meaning, the Rlogin protocol makes sure this character works in the application instead of performing the normal terminal function of [Ctrl]+S, which stops the data flow. These features are not available in all Telnet implementations.

To make a connection with the RLOGin command, follow these steps:

1 Connect to the desired address or resource.

- To connect to an Rlogin resource with an Internet address, use:

```
RLOGin <Internet address>
```

- To connect to an Rlogin resource with a specific name, use:

```
RLOGin <resource name>
```

You can also specify the gateway to make a connection, and then immediately reenter command mode (allows you to enter other connection service commands and is indicated by the NETBuilder prompt).

To enter command mode, type the letters “ECM” after the address or name of the resource as follows:

```
RLOGin <resource name> ECM
```

If the resource you are accessing accepts the connection, the gateway sends the following message to the terminal:

```
session n -- connected to marketing
```

where *n* refers to the number of the session.

If the Rlogin connection does not succeed, you will receive an error message. A connection attempt can fail for a number of reasons. For more information, refer to “Troubleshooting Connection Error Messages” on page 52-7.

When Rlogin connections are made, the client user name (the user name on the client side), and the server user name (user name to be used for login on the server side) are communicated to the server during the connection negotiation. The servername is usually the same, unless you use the -l option (the letter “l”). For example, to enter an Rlogin command specifying a server user name, use:

```
RLOGin <Internet address> -l <server username>
```

Rlogin connections specifying client and server user names can affect access control. For more information on configuring access control, refer to Chapter 51.

2 When you have reached the resource, perform the actions appropriate for the resource application.

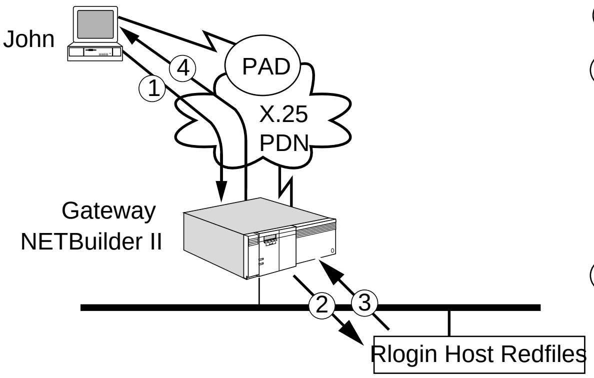
Depending on how the Rlogin host is configured, you may need to enter a password to access the Rlogin host.

3 To disconnect from the session, log out from the host.

You can also enter the ECM character, and then the DisConnect command, from the NETBuilder prompt.

4 To disconnect from the gateway and return to the PAD terminal prompt, use the LOGout or LIsten command.

Figure 52-3 shows how an Rlogin connection is established between a terminal user named “John” at a PAD-attached terminal an Rlogin host named “redfiles.”



1 John enters: **rlog redfiles**

2 The gateway requests a connection to the Rlogin host, sending the Rlogin configuration for John's port, including:
Terminal type
Client username
Server username
Baud rate

3 The Rlogin host sends back confirmation of the request to the gateway. The host may request information on rows and columns from the gateway.

4 The connection is made to John's terminal port.If no username is sent in the connection request, the user must enter a password to access the remote host.

Figure 52-3 Establishing a Connection with the RLOGin Command

Making Connections to OSI Resources

You can use the VTp command to connect to OSI resources on the network. When the VTp command is specified, an OSI connection is made to a specified name or PSAP address. If a list of addresses or names is entered, the gateway tries one address or name after another in the given order until a connection is made.

To make an OSI connection, follow these steps:

1 At the NETBuilder prompt, enter VTp, followed by the name or address to which you want to connect.

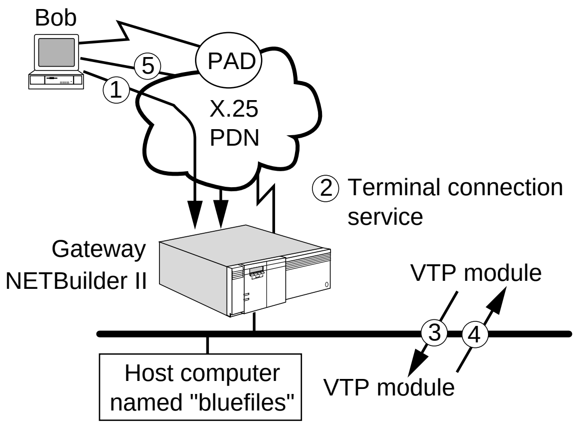
For example, to connect to an OSI resource named “bluefiles” shown in Figure 52-4, enter:

```
VTp bluefiles
```

If you do not know the name of a destination, or a name has not been defined, you can specify the destination NSAP address followed by the upper layer addresses. On a 3Com terminal server, a selector is used to specify a port number. Figure 52-5 shows how a PAD-attached user connects to the modem attached to port 6 on Server B (NSAP address /47/0005/01ABCDEF000000100030080002056821900) by entering:

```
VTp /47/0005/01ABCDEF000000100030080002056821900!6
```

In this example, !6 is the selector. For information on selectors, refer to Appendix E.


Bob
5
1
PAD
X.25
PDN
2 Terminal connection service
Gateway
NETBuilder II
VTP module
3
4
Host computer named "bluefiles"
VTP module

1 Bob enters the command: vt bluefiles

2 The terminal connection service on the gateway looks up the name "bluefiles" and matches an address to the name.

3 The VTP module running on the gateway notifies the VTP module running on "bluefiles" of the connection request.

4 VTP module on "bluefiles" tells the VTP module on gateway that the connection is open.

5 The gateway sends Bob this message: session <*n*> -- connected to bluefiles

Figure 52-4 Connecting to a Name

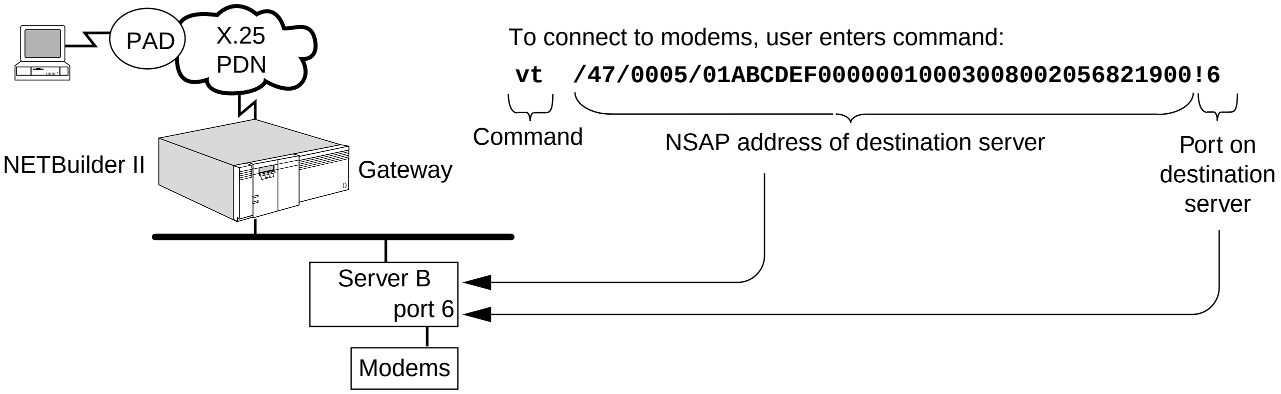

PAD
X.25
PDN
NETBuilder II
Gateway
To connect to modems, user enters command:
vt /47/0005/01ABCDEF000000100030080002056821900!6
Command
NSAP address of destination server
Port on destination server
Server B
port 6
Modems

Figure 52-5 Connecting to an OSI Address

You can also specify the gateway to make an OSI connection, then immediately reenter command mode (allows you to enter other connection service commands and is indicated by the NETBuilder prompt). Type the letters “ECM” following the address or name of the resource as shown in the following example:

```
VTp bluefiles ECM
```

If the connection does not succeed, you will receive an error message. There are different reasons why a connection attempt can fail. For more information, refer to “Troubleshooting Connection Error Messages.“ For information on the command mode and other modes of operation, refer to “Establishing a Single Session” on page 52-10.

2 When you have reached the resource, perform the actions appropriate for the resource application.

3 To disconnect from the session, log out from the host.

4 To disconnect from the gateway and return to the PAD terminal prompt, use the LOGout or LIsten command.

You can also enter the ECM character, and then the DisConnect command from the NETBuilder prompt.

Troubleshooting Connection Error Messages

Connection attempts can fail for different reasons, ranging from errors in the command syntax to discrepancies in how resources are configured. Also, a specified resource may not be configured, or it may not be reachable because of a problem on the network. This section lists the meaning of some common error messages.

Connecting using IP ... aborted, no response from remote host

Meaning: The destination host did not respond to the connection request. The host could be down, or no path exists on the network to this host. The gateway sends this message when it does not receive a response from the host within a given time.

Action: None.

Connecting using IP ... Terminated by the remote TCP host, Reset received

Meaning: The destination host responded to the connection request with a reset packet, in effect, refusing the connection.

Action: None.

IPName: No adequate response received

Meaning: The gateway did not resolve the Internet name entered in the connection request. The name does not exist on the network, or the name was not entered correctly.

Action: Verify that the correct name (including upper- and lowercase letters) was entered.

No more sessions for this port

Meaning: The maximum amount of active sessions for the port has been reached.

Action: To change the number of sessions allowed on each port, the network manager uses the SETDefault !<configfile> -TERM MaxSessions syntax. A single port cannot run more than eight incoming sessions at a time.

X.500 has been selected in -OSI NameSourceOrder but DSAAddress is not configured

Meaning: The gateway did not find the OSI name or address specified. This message can also appear if a non-OSI connection attempt fails; depending on how the -DIR RESolutionOrder parameter is configured, the gateway tries an OSI connection if the non-OSI connection attempt fails. (For example, if an Internet address is entered, an OSI host will not recognize it.)

Action: None.

Checking Network Resources

If you have difficulty connecting to a network resource, you can check to see if the network resource is alive. This procedure differs if you are trying to connect to a TCIP/IP resource or an OSI resource; refer to the appropriate sections below.

Checking TCP/IP Network Resources

If a Telnet or Rlogin connection attempt fails, or if you are not sure if a network resource is available for TCP/IP connections, you can check to see if the resource is "alive," or able to accept a connection by using the PING command.

You can ping a resource with an Internet address using:

```
PING <Internet address>
```

After you enter the command, the gateway sends a request to the target resource to see if it is alive (and available for a connection). If so, the gateway provides an acknowledgment. For example, with an Internet address of 129.41.8.36 the following message is displayed:

```
ping 129.41.8.36
pinging 129.41.8.36 . . . 129.41.8.36 is alive
```

If the target resource is alive, you can then make a connection. If the target resource is not alive, or the gateway cannot find the target resource, you will receive one of several possible error messages depending on the problem. For example, if the resource is on the network, but is not alive, you will receive a message similar to the following message:

```
ping 129.41.8.36
pinging 129.41.8.36 . . . 129.41.8.36 is not responding
```

Other problems can cause a lack of response. For example, the target resource may be on the network, but cannot respond because of a configuration or a hardware problem. Or, the name of the resource entered may be entered incorrectly, or may not exist on the network.

You also may not get a response if there is no route configured to the IP address. If no route exists, a message similar to the following is displayed:

```
pinging 129.41.8.36 . . . 129.41.8.36 is unreachable - no local
  route
```

This message indicates that the gateway cannot reach the address, either because the address does not exist on the network or a route has not been configured to reach that address. You can receive this message if you enter the address incorrectly. For example, if you enter the address 129.14.8.36 when the correct address is 129.41.8.36, the gateway will not find the subnet because the subnet numbers are transposed.

Checking OSI Network Resources

If a VTp connection attempt fails, or if you are not sure if a network resource is available for OSI connections, you can check to see if the resource is “alive,” or able to accept a connection. Use the OPING command to see if an OSI network resource is alive.

You can ping a resource with an network service access point (NSAP) address by using the following OPING syntax which includes the NSAP address:

```
OPING <NSAP address>
```

After you enter the command, the gateway sends a request to the target resource to see if it is alive (and available for a connection). If so, the gateway provides an acknowledgment.

For example, with an NSAP address of /47/0004/0035110008000201F00801 the following message is displayed:

```
oping   /47/0004/0035110008000201F00801
pinging . . . destination is alive
```

If the target resource is alive, you can then make a connection. If the target resource is not alive, or the gateway cannot find the target resource, you will receive one of several possible error messages depending on the problem. For example, if the resource is on the network, but is not alive, a message similar to the following is displayed:

```
oping /47/0004/0035110008000201F00801
pinging . . . dest unreachable according to local routing table
```

Other problems can cause a lack of response. For example, the target resource may be on the network, but cannot respond because of a configuration or a hardware problem. Or, the name of the resource entered may be entered incorrectly, or may not exist on the network.

You also may not get a response if there is no route configured to the NSAP address. If no route exists, a message similar to the following is displayed:

```
pinging. . .received Error Report PDU code 128
```

This message indicates that the gateway cannot reach the address, either because the address does not exist on the network or a route has not been configured to reach that address. You can receive this message if you enter the address incorrectly.

If you still get no response, you can use the OTraceRoute command to trace a path to an OSI destination. For example, to trace the path to the NSAP address above, you would enter:

```
OTraceRoute /47/0004/0035110008000201F00801
```

This command will then display the path to the destination, if it can be found. For more information on the OTraceRoute command, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

Managing Sessions

A *session* is a logical connection between two devices through one or more gateways. A session usually is initiated from a terminal at one end of the connection. Sessions also can be initiated by a network manager on either a local or a remote station as described in Chapter 53.

Table 52-2 summarizes the session management commands that can be used after incoming extended connection session establishment.

Table 52-2 Session Management Commands

Setup	Step	Command

Establish a single session	Enter a connection command.	`Connect <name> | <address>`
		`TELnet <name> | <address>`
		`RLOGin <name> | <address>`
		`VTp <psapaddress> | <name>`

Establish multiple sessions	Enter “ECM” after connection command.	`Connect <name> | <address> ECM`
		`TELnet <name> | <address> ECM`
		`RLOGin <name> | <address> ECM`
		`VTp <psapaddress> | <name> ECM`

Move between sessions	Resume the current session.	`RESume`
	Resume a session other than the current one.	`RESume <sessionnumber>`
	Resume the session number following the current session.	`FORwards`
	Resume the session number preceding the current session.	`BACkwards`
	Change the current session.	`SWitch <sessionnumber>`
Display sessions	Show all sessions on an active port.	`SHow -TERM SESsions`
	Show all sessions on the gateway.	`SHow -TERM AllSessions`
Change modes of operation	Change from command to listen mode.	`LIsten`
	Change from data transfer mode to command mode.	Enter ECM character (default is [Ctrl]+[Shift]+6
Disconnect session connections	Disconnect the current session.	`DisConnect`
	Disconnect a session other than the current one.	`DisConnect <sessionnumber>`

The sections that follow describe each of these commands in detail and give examples.

Establishing a Single Session

To establish a single session, enter a connection command (for example Connect, TELnet, RLOGin, or VTp) in command mode. Command mode is indicated by the command prompt. The default command prompt is NETBuilder > at User privilege level and NETBuilder # at Network Manager privilege level. During connection establishment, the gateway selects the next available port; the port number can be from 0 to 127 on a NETBuilder II system. After the gateway establishes the connection, the selected port at which you enter commands is in data transfer mode, and is actively communicating with a destination.

Figure 52-6 shows the difference among command mode, data transfer mode, and the inactive state of listening mode.

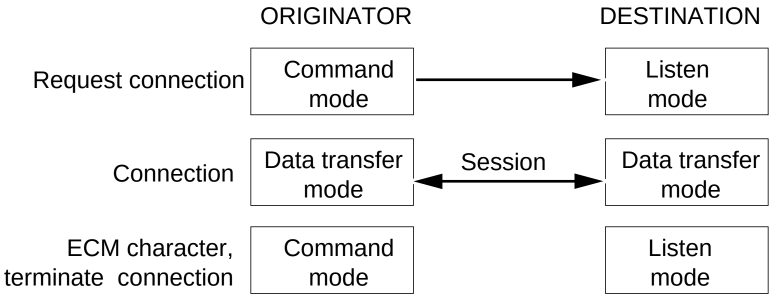


Figure 52-6 Different Port Modes

Establishing Multiple Sessions

You can hold more than one session at a time during incoming extended connections, but you are limited to one session incoming automatic connections. Use the SHow -TERM MaxSessions command to determine the maximum number of sessions that you can hold simultaneously.

If you are already connected to a resource and want to initiate another session, you must enter the ECM command option to switch from data transfer to command mode. Alternatively, you can specify the ECM option when entering the Connect command to restore the port to command mode. You then initiate another session by entering the appropriate connection command.

For example, the following commands are entered to establish three sessions from the connection service gateway:

```
Connect greenfiles ECM
Connect redfiles ECM
Connect bluefiles
```

The first command establishes a session with a resource named “greenfiles” and leaves the port in command mode. The second command establishes a session with a resource named “redfiles” and leaves the port in command mode. The third command establishes a session with “bluefiles” and places the port in data transfer mode. The session with “bluefiles” is called the current session. Only one session at a time is active; all other sessions are suspended and flow-controlled. The sessions are numbered sequentially as you create them. Figure 52-7 illustrates the concept of multiple sessions on one port.

When you specify ECM in a gateway command type the letters “ECM”; when specifying the ECM character from an actual session press the key combination [Ctrl]+[Shift]+6. For more information on the ECM character, refer to “Using the ECM Character to Enter Command Mode” on page 52-13.

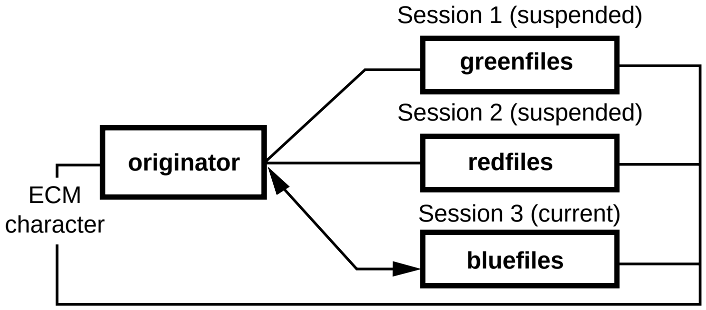


Figure 52-7 Multiple Sessions

Displaying Session Information

To display a numbered list of sessions on your PAD-attached terminal, follow these steps:

1 If the terminal is currently in a session, enter the ECM character to restore the port to command mode.

The default is [Ctrl]+[Shift]+6.

2 To display active sessions on the port you are currently using for connections, enter:

```
SHow -TERM SESsions
```

The following display appears:

```
Sessions on Portid !1
Port/session# state/protocol    Td cnt   Rd cnt
! 1/3 CNCTD/OSI TO bluefiles   0        0
! 1/2 CNCTD/TCP TO redfiles    0        0
! 1/1 CNCTD/TCP TO greenfiles  0        0
```

The first session listed in the display is always the current session. For example, in the preceding display, the session with “bluefiles” is the current session.

Changing the Current Session

To change the current session, follow these steps:

1 Enter the ECM character.

The default is [Ctrl]+[Shift]+6.

2 Switch to session 2 by entering: **SWitch 2**

Session 2 with “redfiles” is now the current session. To confirm that the current session is now session 2, enter:

```
SHow -TERM SESsions
```

This command displays active sessions on the port you are currently using for connections.

Moving between Sessions

Use the RESume, FORwards, and BACkwards commands to move between sessions.

Using the RESume Command

To resume the current session from the gateway command mode, enter the RESume command. For example, to resume the session with “redfiles” in the previous example, follow these steps:

1 At the NETBuilder prompt, enter:

```
RESume
```

2 To resume a session other than the current one, enter the RESume command followed by the session number.

If you are unsure of the number of the session you want to resume, enter:

```
SHow -TERM SESsions
```

Using the FORwards and BACkwards Commands

Use the FORwards command to resume the session number following the number of your current session and the BACkwards command to resume the session number preceding the number of the current session. For example, if you have multiple sessions on port 1 and you enter the SHow -TERM SESsions command, the following display appears:

```
Sessions on Portid !1
Port/session# state/protocol       Td cnt      Rd cnt
! 1/2 CNCTD/TCP TO redfiles        0           0
! 1/3 CNCTD/TCP TO greenfiles      0           0
! 1/1 CNCTD/OSI TO bluefiles       0           0
```

In this example, session 2 is the current session. If you enter the FORwards command, session 3 is resumed. If you enter the BACkwards command, session 1 is resumed.

Using the ECM Character to Enter Command Mode

After you have established a session, the gateway is in data transfer mode, and the gateway commands are not accessible. To establish other sessions, disconnect sessions, or enter other gateway commands, you must exit the current session using the ECM character.

The default ECM character is the key combination [Ctrl]+[Shift]+6, which produces a double caret (^^). On most standard keyboards, the caret (^) is the same as the key for the number 6; if the caret is on a different key, press the Control key and the appropriate caret key.

You can change the default ECM character when the application requires that the double caret character (^^) be transmitted as data. You can change the default ECM character for all your new sessions on the gateway using the SETDefault -TERM ECMChar command. If you change the default ECM character, it only affects new sessions; for sessions already existing, the previous ECM character is still required. For example, to change the ECM character for the gateway to the key combination [Ctrl]+T (^T), enter:

```
SETDefault -TERM ECMChar = ‘^T’
```

You can also change the ECM character for only the current session with the SET -TERM ECMChar command. The change affects only the current session and does not affect sessions on other ports. For example, to change the ECM character for a session on an active port to the key combination [Ctrl]+T (^T), enter:

```
SET -TERM ECMChar = ‘^T’
```

When you exit a session with the ECM character, you are not disconnecting the session. For more information about disconnecting sessions, refer to the next section.

Some applications require that the ECM character be disabled because the ECM escape characters are interpreted as normal data. In such cases, the BReakAction parameter can be configured so that entering the Break key causes the gateway to enter command mode (for example, setting BReakAction to “EscDTM”). For more information, refer to Chapter 61 in *Reference for NETBuilder Family Software*

The ECM escape character is used only to enter command mode from *within* an active session. By typing the letters "ECM" at the end of a connection command, you can instruct the gateway to make a connection, then automatically reenter command mode. For example, to connect to a resource named "greenfiles," and then enter command mode, enter:

```
Connect greenfiles ECM
```

Disconnecting a Single Session

To disconnect a session, enter:

```
DisConnect
```

The port is in command mode.

Disconnecting Multiple Sessions

When holding more than one session, to disconnect the current session enter:

```
DisConnect
```

To disconnect a session other than the current one, use the DisConnect command followed by the session number. For example, to disconnect the session with "redfiles" (shown in Figure 52-7), enter:

```
DisConnect 2
```

The DisConnect command leaves your port in command mode. To disconnect all PAD sessions and the X.25 connection, and place the port in listening mode, enter:

```
LIsten
```

Changing Session Parameters

To change session parameters for the current session, use:

```
SET -TERM <parameter>
```

To change your session parameters for future sessions, use:

```
SETDefault -TERM <parameter>
```

To view session-related parameters that can be changed, enter:

```
SHow -TERM DefaultParams
```

You must have an active port to display the default parameter values that initialize the port and session.

For a list of parameters affecting sessions, refer to Chapter 61 in *Reference for NETBuilder Family Software*.

53 NETWORK MANAGEMENT

The bridge/router participates in different types of network management activities. Most management activities require configuration because they are disabled by default. The system manages networks in the following ways:

- Using file service
- Building network maps (netmaps)
- Sending AuditLog messages to a Network Management Station

This chapter describes these management activities, but it does not describe the protocols involved. For information on the protocols, refer to the appropriate RFCs. For information on the parameters referenced in this chapter, refer to *Reference for NETBuilder Family Software*. The network management information in this chapter applies to the system regardless of its functionality unless otherwise specified.

Simple Network Management Protocol

Simple Network Management Protocol (SNMP) allows you to modify and display some of the bridge/router system parameters from a host; you do not need to attach a terminal to the system console port to change its configuration. The system implementation of SNMP follows the specifications in RFCs 1155, 1157, and 1213. The system parameters and the 3Com-extended parameters that are described in several RFCs including RFC 1213 can both be accessed from the host.

On a 3Com bridge/router, SNMP support is enabled by default. If you want to disable SNMP support, enter:

```
SETDefault -SNMP CONTrol = NoManage
```

Proceed to the next section for instructions on configuring other parameters. All the parameters referenced in this section are SNMP parameters.

Configuring the SNMP Service

This section describes how to configure the SNMP Service.

By default, the community name ANYCOM exists with read access to management information base (MIB) variables and allows unrestricted access to the bridge/router. To ensure that access is available only to the proper system administrator, 3Com recommends that you delete the ANYCOM community name, and add the appropriate community string and the manager's IP address.

Procedure

To configure the bridge/router for SNMP management, follow these steps:

1 Delete the default community string “ANYCOM.”

For example:

```
DELete -SNMP COMmunity “ANYCOM”
```

2 Configure at least one new community string with read/write access.

For example:

```
ADD -SNMP COMmunity “private” Triv RW AL1
```

3 Add other community strings with read-only access as required.

For example:

```
ADD -SNMP COMmunity “public” Triv RO AL1
```

You can have up to ten managers for each community. Including ANYCOM, you can have up to six communities.

When you enter ANYCOM to the list of community names, it must be entered in all uppercase letters.

4 Configure at least one SNMP manager to the read/write community string.

For example:

```
ADD -SNMP MANager “private” <IP addr>
```

5 SNMP is enabled by default. If SNMP is disabled on your system and you want to enable it, enter:

```
SETDefault -SNMP CONTrol = Manage
```

Related Information

The information in the following sections provide you with additional information about SNMP.

Request Validation

The following options are available with a request for validation:

- For security purposes, the SNMP agent on the system validates SNMP requests before responding. This prevents unauthorized users from viewing or changing the bridge/router configuration.
- You can specify that only the hosts with known community names can send requests. All the community names known to the system are specified by the COMmunity parameter. A request cannot be authenticated if its community name is not included in the COMmunity parameter. To allow requests from any community, add ANYCOM to the list of community names.
- The information in the ANYCOM entry then processes requests with unmatched community names. When adding a community to the list, you can also specify the level of access to the MIB, read or read/write, and the type of trap sent to managers associated with the community name.
- In addition to specifying a set of community names, you can create a list of managers for each community name. If there is no manager list associated with a community name, the system responds to any request with that community name; otherwise, before an incoming request is processed, it must have a matching Internet address for the community name that is specified by the MANager parameter.

Using Traps. Traps are sent by the SNMP agent to alert the network management station of unusual events. The following six events are defined by the IETF in RFC1213:

- Cold Start
- Warm Start
- Link Up
- Link Down
- Authentication Failure
- EGP NeighborLoss

With the exception of the Warm Start event, traps are sent each time an event occurs. In addition to these traps, the SNMP agent also sends traps when a Frame Relay virtual circuit changes state, when a bridge port changes spanning tree states, when the station becomes the root of a bridge spanning tree, when an RMON event is triggered, and when an unauthorized Dynamic Host Configuration Protocol (DHCP) or Bootstrap Protocol (BOOTP) server is detected.

Set Operation. You can use the SNMP SetRequest protocol data unit (PDU) to set the objects on a bridge/router. The objects may be single or a table entry field (that is, you may specify multiple objects, single or a table entry field, for one PDU). If the request does not follow these restrictions, the SNMP agent responds with a general error PDU, and the set operation is disallowed.

Remote Network Monitoring Alarms

To assist network managers in identifying abnormal activity that could adversely affect network performance, the system contains a Remote Monitoring (RMON) agent based on RFC 1271. The 3Com implementation supports two RMON object groups as follows:

- The Alarm group

  You can set up alarms to monitor the MIB objects of interest through the Alarm group; this group includes an Alarm Table, which you must configure before using alarms. 3Com allows five alarms to be configured in the Alarm Table.

- The Event group

  You can set up events to either record the monitoring information or notify the network management station. The Event group includes an Event Table and a Log Table. You must configure the Event Table before using it. The Log Table needs no prior configuration; it is a read-only data table for the network management station. 3Com allows five events to be configured in the Event Table.

  Because the system contains no user interface to access the RMON agent, you must access it through an SNMP network management station by using SNMP SET/GET/GET-NEXT requests, or by using an SNMP application that can generate SNMP requests. In order to monitor MIB objects of interest according to a set or preset alarms and events, you need to configure the RMON alarms and events with the desired control parameters. The control parameters to be configured include alarm threshold values, which are used to determine if an event needs to be generated. An event is generated if the newly read MIB value has crossed the threshold. The event can take any of the following actions:

  - The system sends an SNMP trap to the network management station.

    The management station is notified immediately. The management station determines how to react to the SNMP trap.

- The system logs the event into a Log Table in the agent system.

  The management station can retrieve the information stored in the Log Table for further analysis. For example, the information collected can be used in selecting proper threshold values.
- The system sends an SNMP trap and logs into the Log Table.

Network Maps

A network map (netmap) contains the Ethernet and Internet addresses of each 3Com device on the attached network and the software version on the device. The bridge/router can participate in building the netmap by broadcasting its addresses to the network at regular intervals (defined by NetMapTime in the SYS Service).

By default, the value of the NetMapTime parameter is set to 0, which means that the system does not broadcast its addresses to the network. To configure the system to broadcast its addresses to the network a specific number of seconds, use:

```
SETDefault -SYS NetMapTime = <number>(0 to 120 seconds)
```

The network map can be used as a network management tool, because you can see at a glance which 3Com devices are on the attached network. To display the network map, enter:

```
SHow -SYS NetMAP
```

For the possible variations for this command and what the various commands will display, refer to Chapter 58 in *Reference for NETBuilder Family Software*. For example, to show all the devices that support the TCP/IP Protocol on the network attached to port 1 of the system, enter:

```
SHow !1 -SYS NetMAP tcp
```

The following display appears:

```
NETWORK 1 MAP
0-%08000200F84A  129.213.16.206  1-%08000201136A  129.213.16.208
2-%08000200D26C  129.213.17.152  3-%080002013E86  129.213.18.47
4-%080002013E98  129.213.19.67   5-%080002013E9C  129.213.18.52
6-%0800020128B8  129.213.17.147  7-%08000201ECD3  129.213.16.215
```

To show all the devices that support the Xerox Network Systems (XNS) Protocol on the networks attached to each port of the system, enter:

```
SHow -SYS NetMAP xns
```

The following display appears:

```
                    NETWORK &00000002 MAP
3-%080002A01339
                    NETWORK &00000000 MAP
0-%080002A0133A
                    NETWORK &00000003 MAP
0-%080002A03C1B
                    NETWORK &00000000 MAP
0-%080002A01339
```

If you want to display the software version running on each 3Com device, use the Long value in the command. For example, enter:

```
SHow !1 -SYS NetMAP Long
```

Because this command does not specify the XNS or TCP value, devices that support TCP on the network attached to port 1 are displayed. The following display appears:

```
NETWORK 1 MAP
0-%08000200F84A       129.213.16.206        SW/1-PCS 10000
1-%08000201136A       129.213.16.208        SW/1-PCS 10000
2-%08000200D26C       129.213.17.152        SW/1-TCP 3.0
3-%080002013E86       129.213.18.47         SW/NB-IB-2.0
4-%080002013E98       129.213.19.67         SW/NB-BR-3.0
5-%080002013E9C       129.213.18.52         SW/NB-IB-2.0
6-%0800020128B8       129.213.17.147        SW/200-TCP 3.0
7-%08000201ECD3       129.213.16.215        SW/1-PCS 1001h
```

To display the software version running on each device that supports the XNS Protocol, enter:

```
SHow -SYS NetMAP xns Long
```

Because this command does not specify a port, devices that support the XNS Protocol on the networks attached to each port of the system are displayed. The following display appears:

```
                    NETWORK &00000001 MAP
0-%080002A03C1B     SW/NB-BR-3.0
                    NETWORK &00000000 MAP
0-%080002A03C1C     SW/NB-BR-3.0
                    NETWORK &00000003 MAP
0-%080002A03C1B     SW/NB-BR-3.0
3-%080002A01339     SW/NB-BR-3.0
                    NETWORK &00000000 MAP
0-%080002A03C1B     SW/NB-BR-3.0
```

Logging Configuration Changes

The AuditLog Service sends event log messages to a network management to provide a history of configuration changes and other events useful in monitoring NETBuilder bridge/routers. To use the AuditLog Service you must have IP and User Datagram Protocol (UDP), and a configuration of the UNIX Syslog daemon.

The log messages provided by the AuditLog Service offer information concerning:

- User logins and listens (logouts)
- Failed login or set privilege attempts
- Successfully executed configuration commands
- Invalid SNMP community strings
- SNMP configuration changes

- File operations
- System and dial history messages
- Reboot information
- Audit trail messages

A sample AuditLog display follows:

The UNIX Syslog daemon inserts its own time stamp, which preceeds the columns of the AuditLog display.

```
#51 Wed Feb  7 17:35:25 1996 129.213.40.51  root CONSOLE “Login Successful”
#52 Wed Feb  7 17:39:58 1996 129.213.40.51  root CONSOLE “setd !2 -pa name=”Ring_1”name=”Ring_1””
#53 Wed Feb  7 17:40:43 1996 129.213.40.51  root   “Path 2 DOWN”
#54 Wed Feb  7 17:40:43 1996 129.213.40.51  root CONSOLE “setd !2 -pa cont=ena”
#55 Wed Feb  7 17:40:55 1996 129.213.40.51  root CONSOLE “Listen executed”
#56 Wed Feb  7 17:40:58 1996 129.213.40.51  root   “Path 2 UP”
#57 Wed Feb  7 17:41:19 1996 129.213.40.51  root   “Port 2 - Forwarding State”
#58 Wed Feb  7 17:41:27 1996 129.213.40.51  login TELNET 129.213.40.10  “Login Failed”
#59 Wed Feb  7 17:41:27 1996 129.213.40.51  root TELNET 129.213.40.10  “Listen executed”
#60 Wed Feb  7 17:43:29 1996 129.213.40.51  root SNMP 129.213.40.59  Invalid Community string: “pass”
#61 Wed Feb  7 17:43:30 1996 129.213.40.51  root SNMP 129.213.40.59  Invalid Community string: “pass”
#62 Wed Feb  7 17:43:31 1996 129.213.40.51  root SNMP 129.213.40.59  Invalid Community string: “pass”
#63 Wed Feb  7 17:44:22 1996 129.213.40.51  root SNMP 129.213.40.59  “1.3.6.1.4.1.43.2.29.1.2.0  1"
#64 Wed Feb  7 17:47:42 1996 129.213.40.51  root REMOTE 129.213.40.52  “fl -ip allr”
#65 Wed Feb  7 17:50:40 1996 129.213.40.51  root REMOTE 129.213.40.52  “setd !8 -pa cont=ena”
#66 Wed Feb  7 17:51:39 1996 129.213.40.51  root TELNET 129.213.40.10  “Login Successful”
#67 Wed Feb  7 17:52:53 1996 129.213.40.51  root TELNET 129.213.40.10  “ping 129.213.40.59"
#68 Wed Feb  7 17:54:04 1996 129.213.40.51  root TELNET 129.213.40.10  “fl -br allr”
#69 Wed Feb  7 17:54:55 1996 129.213.40.51  root Changed LNM password (encrypted:)~9i6n!^DHDsR^N^D$^]:
#70 Wed Feb  7 17:55:46 1996 129.213.40.51  root TELNET 129.213.40.10  “saw”
#71 Wed Feb  7 17:56:15 1996 129.213.40.51  root TELNET 129.213.40.10  “Listen executed”
#72 Wed Feb  7 17:56:23 1996 129.213.40.51  root CONSOLE “Login Successful”
#73 Wed Feb  7 17:56:31 1996 129.213.40.51  root CONSOLE “Set Privilege Failed”
#74 Wed Feb  7 18:11:31 1996 129.213.40.51  root CONSOLE “UI session terminated”
```

The message includes the following fields:

Sequence number	A number from 0 to 255.
Time Stamp	Specifies the time (date, time, and year) a message is logged.
Recording NETBuilder	Specifies the IP address of the lowest port number of the recording NETBuilder II system.
User	Root user is the default.
Access	May be console, remote, TELNET or SNMP (Remote, TELNET and SNMP are all followed by an IP address), Event-Based Macro Execution or Scheduler. The Access field is not generated for MEssage category log messages.
Command or Message	Specifies the actual message in quotes or successfully executed commands such as SETDefault, SET, Add, or Delete. For SNMP, the Object ID is displayed and value. Any SEcurity-related or privilege violations display a warning message.

Audit Trail Messages

Beginning in software version 9.0, audit trail messages that were formerly sent to Remote Boot Configuration Services (RBCS) servers are now logged through the AuditLog Service when specified by the Audit Trail control category. For a list of audit trail messages generated by the NETBuilder II bridge/router, refer to Appendix J.

Configuring the Network Management Station for AuditLog

The AuditLog Service uses the Syslog logging service provided by the syslogd daemon available on most UNIX systems. To use AuditLog, the network must support the IP/UDP protocol between the NETBuilder II bridge/router and the network management station. Because delivery of UDP messages is not guaranteed, some Syslog messages may be lost due to network conditions between the NETBuilder II bridge/router and the network management station.

Before using the AuditLog Service, you must configure a network management station to receive NETBuilder log messages. On a UNIX network management station that already has the Syslog daemon running, follow these steps:

1. Log on to your network management station as root.
2. Add an entry for local7 (or the facility selected through the -AuditLog LocalFacility parameter) at the end of your /etc/syslog.conf file. For example:

   ```
   # 3Com NETBuilder AuditLog messages
   LOCAL7.LOG_INFO var/log/auditlog
   ```

   Use ONLY TAB characters between the facility/priority code and the filename. The location of the log file may vary depending on the system you are using. Check your system references or other entries in the /etc/syslog.conf file. You may want to choose a location different from the default location.
3. Create an empty log file to receive the Syslog messages. This must be the same file you specified in syslog.conf. For example, use the filename:

   ```
   cat /dev/null > /var/log/auditlog
   ```
4. Restart the Syslog daemon by entering:

   ```
   kill -1 ‘cat /etc/syslog.pid`
   ```

SNMP Event Notification Traps

The AuditLog Service for the NETBuilder II bridge/router provides audit trail and notification for tracking and management of NETBuilder configuration changes. Notification is based on SNMP enterprise-specific trap messages. You can generate traps independently, even if the audit log capability has been disabled.

The following types of events can trigger a change or notification trap:

- Configuration Change Trap (Numeric Value 101)

  This trap tracks changes originating from the execution of the SET, SETDefault, Add, and DELete commands.
- User Authentication Trap (Numeric Value 102)

  This trap captures failed logins.
- File Operation Command (Numeric Value 103)

  This trap captures file operations from COpy, RemoveFile, RemoveDir, ReName, MakeDir, GET, and PUT commands.

Audit log notification requires no specific configuration or parameters, but is associated with the configuration of the SNMP Service trap generation. To configure SNMP for audit trail notification, follow these steps:

1. Enable SNMP trap generation by entering:

   ```
   SETDefault -SNMP CONTRrol = Trap
   ```
2. Add an SNMP community for your network management station using:

   ```
   ADD -SNMP COMmunity <“name”> TRiv RW ALl
   ```

The value All specifies that all SNMP traps are enabled. The SNMP trap types are general, authentication, and enterprise-specific. The change traps are enterprise-specific.

3 Specify a destination for traps using:

```
ADD -SNMP MANager <community> <IPAddress>[<mask>]
```

Remote Access of Your System

You can access your system remotely to perform network management operations. These operations can be completed through a remote station, such as a Sun workstation, which does not need to be physically connected to the console port of your system. Remote access is accomplished by using the proprietary REMote command from another 3Com bridge/router or by using the Telnet Protocol from a Telnet client.

Depending on your reasons for accessing the system remotely, you may want to use TELnet instead of REMote. These commands differ in the following ways:

- The REMote command provides access to a subset of the bridge/router commands, is UDP-based, and can truncate long displays.
- The TELnet command provides access to all bridge/router commands and is TCP-based.

After the connection has been established through remote mode or the Telnet Protocol, you can change your privilege level to Network Manager, provide the correct password, and perform network management operations or configuration procedures.

You can also prevent unauthorized users from making remote connections to your system by configuring the -SYS NetAccess parameter, and you can restrict remote access to specific users by configuring the -SYS RemoteManager parameter.

Using the REMote Command or the TELnet Command

The REMote command allows you to execute commands on your bridge/router from a remote terminal. After you enter the REMote command followed by an IP address or name, you are in remote mode. Remote mode is indicated on your remote terminal by the appearance of the remote prompt (Remote:). In remote mode, all commands entered affect your bridge/router until you exit remote mode. To exit remote mode, press the Break key and enter TELnet to return to Telnet mode.

The REMote command is not subject to a configured password. To restrict access to the system, disable REMote and use the TELnet command instead.

To display the software version on a bridge/router with the address 129.3.4.2, see Figure 53-1, and follow these steps.

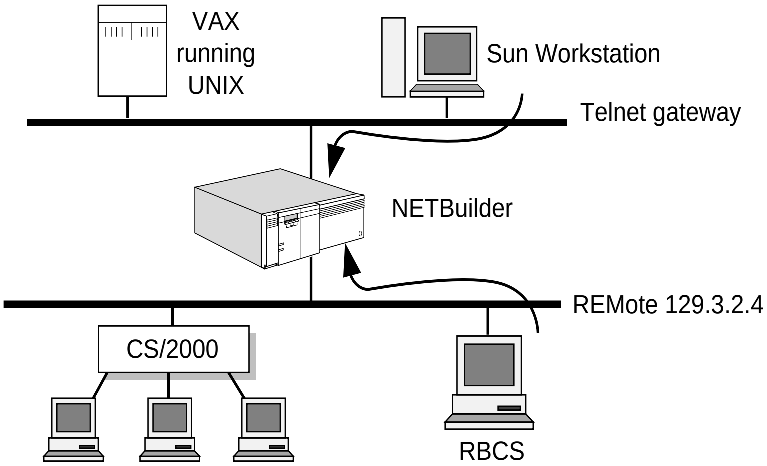


Figure 53-1 Accessing the System Remotely

1. On the remote RBCS terminal, enter:

   ```
   REMote 129.3.4.2
   ```

   The prompt from the remote system is displayed.

2. At the remote prompt enter:

   ```
   SHow -SYS VERSion
   ```

   The version information of the software running on the system is displayed.

3. Press the Break key to return to the command prompt of your remote terminal.

You also can enter the REMote command, followed by the address of the bridge/router, and then followed by a command to be executed.

For example, the following command displays the IP Routing Table of the system that has the address of 129.3.2.4:

```
REMote 129.3.2.4 SHow -IP AllRoutes
```

Some bridge/router commands cannot be used in remote mode. For a list of these commands, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

The Telnet Protocol also can be used on a remote terminal to access your bridge/router. In this situation, your system functions as the Telnet server (the destination), and the remote terminal functions as the Telnet client (the initiator).

To access a bridge/router called “gateway”, on the remote terminal (RBCS or Sun workstation), see Figure 53-1 and enter:

```
TELnet gateway
```

The user level command prompt of the bridge/router named “gateway” appears on the remote terminal. After you change the privilege level to Network Manager and enter the password, you can perform network management procedures. After your management activities are complete, enter the LIsten command to disconnect the session and place the port in listen mode.

Preventing Remote Access

By default, your system can be accessed remotely using the REMote command or the Telnet Protocol. You can regulate access from remote devices by using the SETDefault -SYS NetAccess command.

- To disable access with the REMote command, enter:

  SETDefault -SYS NetAccess = NoRemote

- To disable access with the Telnet Protocol, enter:

  SETDefault -SYS NetAccess = NoTelnet

CAUTION: *The software allows the bridge/router to be disabled without giving any warning messages. After assigning NoRemote or NoTelnet, you can no longer access the system parameters to perform software configuration. You must boot the system software diskette that contains an enabled NetAccess parameter before you can regain access.*

If IP security options are implemented on the system ports, a remote station without matching IP security options is not allowed to access the system. For information on restricting access to IP routers and end system configurations, refer to Chapter 8.

Restricting Remote Access

To specify that only certain devices can access your system, use:

```
ADD -SYS RemoteManager <IPaddress>
```

This command does not control XNS-based remote access.

To allow only the device with the address 129.98.96.99 to access your system, obtain Network Manager privilege and follow these steps:

1 Remove remote access to your system from all devices by entering:

DELete -SYS RemoteManager *.*.*.*

2 Specify that only the address 129.98.96.99 can access your system by entering:

ADD -SYS RemoteManager 129.98.96.99

You can configure a maximum of three RemoteManager addresses. For additional information on these commands, refer to Chapter 58 in *Reference for NETBuilder Family Software*.

Restricting Telnet Access

You can specify that only certain devices can access your system using the Telnet Protocol using:

```
ADD -SYS TelnetManager <IP address>
```

For example, to allow only the devices with the address 129.213.48.18 and 139.87.180.* to access your system using the Telnet Protocol, obtain Network Manager privilege and complete the following steps:

1 Remove Telnet access to your system from all devices by entering:

DELete -SYS TelnetManager *.*.*.*

2 Specify that only the devices with the address 129.213.48.18 and 139.87.180.* can access your system by entering:

```
ADD -SYS TelnetManager 129.213.48.18
ADD -SYS TelnetManager 139.87.180.*
```

You can configure a maximum of six TelnetManager addresses. For more information, refer to Chapter 58 in *Reference for NETBuilder Family Software*.

Resynchronization Feature for Encryption Devices

You can connect KG-81/KG-94 encryption devices to WAN ports (for example, ports 3 and 4, which have an RS-449 connector). The KG-81 and KG-94 encryptors are available for use by U.S. government installations only.

After you establish the connection between the bridge/router and one of these devices, enable the resynchronization feature on the bridge/router using:

```
SETDefault !<path> -PATH CONTrol = Crypto
```

For example, to enable the resynchronization feature on path 3, enter:

```
SETDefault !3 -PATH CONTrol = Crypto
```

When this feature is enabled, if a path goes down, a pulse that attempts to resynchronize the devices is generated on one of the signal lines (DTR) on the RS-449 connector. Each pulse is 10,000 microseconds long and is sent approximately every 10 seconds. Pulses will be sent until resynchronization occurs.

Disable the resynchronization feature only when you disconnect the bridge/router from the device. In the following example, the resynchronization feature is being disabled on path 3:

```
SETDefault !3 -PATH CONTrol = NoCrypto
```

To display the current settings of the CONTrol parameter in the PATH Service, enter:

```
SHow -PATH CONTrol
```

This feature is supported only on ports running PLG and PPP.

LAN Net Manager Support

You can configure the bridge/router to provide information to LAN Net Manager stations on your token ring network. LAN Net Manager is an IBM network management application used to monitor and perform some configuration of token ring networks. As implemented by IBM, the LAN Net Manager application communicates with one of five management servers that reside on each token ring network. These servers provide information to LAN Net Manager regarding current conditions of the ring, and provide some control over the ring. For example, using the application you can remove stations from the ring or change their operating parameters.

Of the five IBM management servers, the bridge/router provides support for the following four servers:

- LAN Reporting Mechanism
- Ring Error Monitor
- Configuration Report Server
- Ring Parameter Server

For more information, refer to your LAN Net Manager documentation.

Configuring LAN Net Manager Support

When you configure LAN Net Manager support on a bridge/router, the bridge/router and the attached token rings become eligible for monitoring by the IBM LAN Net Manager application.

By default, LAN Net Manager support is disabled. To configure a bridge/router for LAN Net Manager support, follow these steps:

1 Set the baud rate for the LAN Net Manager paths using:

```
SETDefault !<path> -PATH BAud = <kbps>
```

2 Enable global bridging by entering:

```
SETDefault -BRidge CONTrol = Bridge
```

3 Enable Logical Link Control, type 2 (LLC2) on the LAN Net Manager ports using:

```
SETDefault !<port> -LLC2 CONTrol = Enable
```

4 Configure source route bridging.

a Assign a unique bridge number to the bridge/router using:

```
  SETDefault -SR BridgeNumber = <number> (0–15) | 0x<number>
 (0–F)
```

b Assign a unique ring number to each LAN Net Manager port:

```
SETDefault !<port> -SR RingNumber = <number> (1–4095) |
  0x<number> (1–FFF)]
```

c Configure the LAN Net Manager ports for source route bridging using:

```
SETDefault !<port> -SR SrcRouBridge = SrcRouBridge
```

d Turn on source route discovery for LLC2 on the LAN Net Manager ports using:

```
SETDefault !<port> -SR RouteDiscovery = LLC2
```

5 Reenable the LAN Net Manager paths and ports using:

```
SETDefault !<path> -PATH CONTrol = Enabled
SETDefault !<port> -PORT CONTrol = Enabled
```

6 If LNM is used in Virtual Ring mode, set up a virtual ring and bridges.

a Assign a virtual ring number to the bridge/router using:

```
SETDefault -LNM VirRingNum = <number> (1–4095)
```

Once a virtual ring number is assigned to a bridge/router, it cannot be linked to as a physical bridge.

b Assign a virtual bridge number to each LAN Net Manager port using:

```
SETDefault !<port> -LNM VirBrNum = <number> (0–15)
```

7 If the password for Bridge parameters in LAN NET Manager has been changed, enter the same password in the 3Com bridge/router, using:

```
SETDefault -LNM PassWord = “<string>”
```

8 Set the number of alternate LAN Net Manager stations supported by the bridge/router using:

```
SETDefault -LNM NumAltMgrs = <number> (0–5)
```

The default value is 4.

9 Enable LAN Net Manager control by entering:

```
SETDefault -LNM CONTrol = Enabled
```

You can set other parameters in the LNM Service to customize timers and thresholds for your LAN Net Manager configuration. For more information about parameters in the LNM Service, refer to Chapter 35 in *Reference for NETBuilder Family Software*.

Configuring Virtual Bridges and a Virtual Ring for NETBuilder II

When supporting LAN Net Manager, the NETBuilder II system must adapt to certain limitations imposed by the LAN Net Manager application. LAN Net Manager assumes that all bridges have only two ports, and as a result imposes this limit on the number of ports on a bridge. Since the NETBuilder II system can support multiple bridged ports, these must appear to LAN Net Manager as multiple two-port virtual bridges. Each virtual bridge connects a token ring port to an internal virtual ring.

Figure 53-2 is an example of a NETBuilder II system with four virtual bridges connected to a single virtual ring that is internal to the system.

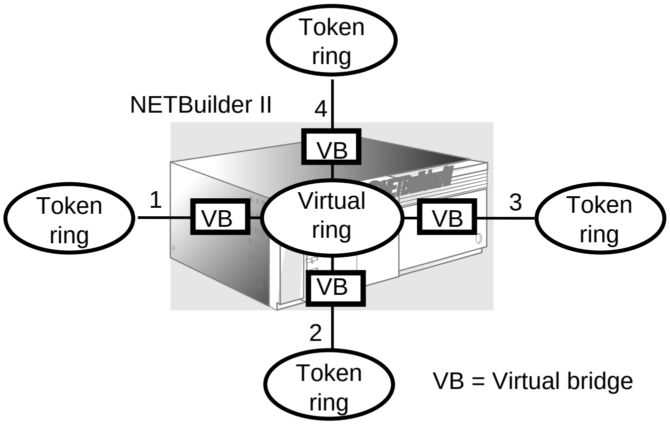


Figure 53-2 Virtual Bridges and Virtual Ring on a NETBuilder II System

When configuring LAN Net Manager to monitor a the token rings of a NETBuilder II bridge/router, several virtual bridges must be defined, one for each NETBuilder II token ring. One port of each virtual bridge corresponds to a real token ring port while the other port is attached to the virtual ring. In the figure, for example, there are four virtual bridges, each connected to a token ring.

When configuring this virtual bridge on the LAN Net Manager, enter the media access control (MAC) address of the token ring port and any dummy MAC address you may have configured for the virtual port. The NETBuilder II system automatically assumes it is a port on the virtual ring. On the NETBuilder II system, bridge numbers must be assigned to each of the virtual bridges, and a unique ring number must be assigned to the virtual ring. These numbers are used only to work around the two port limitations of the LAN Net Manager, and will not affect other source route bridging operations.

Disabling LAN Net Manager Support

To disable LAN Net Manager support, enter:

```
SETDefault -LNM CONTrol = Disable
```

When LAN Net Manager support is disabled, the bridge/router does not respond to requests from LAN Net Manager stations, nor does it send notifications to LAN Net Manager stations. If LAN Net Manager support is disabled when reporting links to LAN Net Manager stations are established, links will be gracefully terminated (as defined by IBM) by the bridge/router before disabling.

To reenable LAN Net Manager support, enter:

```
SETDefault -LNM CONTrol = Enable
```

If you enable LAN Net Manager support on a token ring where LAN Net Manager is not resident, you must configure the bridge/router to use end system source routing. To do this, turn on source route discovery for LLC2 using:

```
SETDefault !<port> -SR RouteDiscovery = LLC2
```

AMP-Based Network Device Discovery

Adapter Management Protocol (AMP) discovery is a 3Com protocol used by 3Com network management platforms to discover network devices attached to LAN segments. AMP operates at the MAC/LLC layer and uses group addressing. AMP discovery to 3Com bridge/routers includes a built-in discovery responder in the software.

Discovery is accomplished by a station transmitting a discovery request frame addressed to the AMP group address. If the transmitting station is attached to an Ethernet or FDDI segment, a multicast address is used. If the station is attached to a token ring segment, a functional address is used. Devices receiving the request frame respond by transmitting a discovery response frame directly addressed (unicast) to the requesting station.

The responder for the bridge/router listens for discovery request frames on LAN media interfaces (Ethernet, FDDI, token ring, and bridged serial). A bridge/router operating as both a bridge and a router responds to and forwards requests over bridged interfaces. When a request frame is forwarded between an Ethernet or FDDI segment and a token ring segment, the destination address requires a mapping between multicast and functional addresses. Only request frames require this mapping; all other AMP discovery frames are directly addressed.

The multicast address and default functional address used for AMP discovery are shown in Table 53-1. The multicast address is reserved and is not configurable. The functional address is configurable.

Table 53-1 AMP Multicast and Functional Addresses

	Noncanonical	Canonical
AMP Multicast Address	8006 3188 8858	0160 8C11 111A
AMP Functional Address	C000 0100 0000	0300 8000 0000

Configuring the Discovery Responder

The Discovery Responder is a built-in service, and no configuration commands apply directly to it. When the bridge/router starts up, it attempts to map the AMP multicast address to a functional address (either the AMP default or a user-defined one). If this attempt is successful, the Discovery Responder for the bridge/router will be able to receive requests that originate on a token ring segment.

User configuration is required only when the AMP default functional address needs to be mapped to some other multicast address. In this case, you can use the BRidge Service to establish a mapping between the AMP multicast address and a different AMP functional address.

CAUTION: *The functional address used for AMP discovery is a network-wide address. All devices supporting AMP discovery (including PCs with 3Com token ring adapters and bridge/routers) must use the same functional address.*

Configuring AMP Using the BRidge Service

To see which functional addresses are mapped to which multicast addresses, use the SHow command. In the following example, entry 1 shows the default AMP functional-multicast mapping.

Enter:

```
SHow -BRidge FunctionalAddr
```

The following display appears:

```
No.           Functional Address         Multicast Address
----          ------------------         -----------------
       1      %030080000000              %01608C11111A
       2      %0300FFFFFFFF              %FFFFFFFFFFFF
       3      %030000008000              %0180C2000000
       4      %FFFFFFFFFFFF              %FFFFFFFFFFFF
-- Entries displayed = 4
```

To remove the mapping of the AMP default functional address to the AMP multicast address, use the DELete command. For example:

```
DELete -BRidge FunctionalAddr = %030080000000
```

To map a new (non-default) AMP functional address to the AMP multicast address, use the ADD command. For example:

```
ADD -BRidge FunctionalAddr = %030040000000 MultiCastAddr =
  %01608C11111A
```

For information about available functional addresses, refer to “Adding Functional-Address-to-Multicast-Address Mappings to the Default Table” on page 3-17.

A SWAPPING NETBUILDER II HARDWARE MODULES

This appendix describes how to swap modules in your NETBuilder II bridge/router while your bridge/router software continues operating on other modules and ports.

You can swap one type of module with another, or you can swap two modules of the same type.

Swapping Hardware Modules

To swap hardware modules, follow these steps:

1 Disable the path mapped to the hardware module using:

```
SETDefault !<path> -PATH CONTrol = Disabled
```

***CAUTION**: When you disable the path, you risk losing network connections associated with that path.*

2 Hot-swap the board.

Remove the board and replace it with a new board.

For more information on performing the swap operation, refer to the appropriate hardware module installation guide.

3 Re-enable the path using:

```
SETDefault !<path> -PATH CONTrol = Enabled
```

B

Dial-up Progress and Error Messages

This appendix provides dial-up progress and error messages for modems and integrated services digital network (ISDN) terminal adapters (TAs). It also provides information about the NETBuilder II I/O module, which supports data terminal ready (DTR) and V.25bis dialing. In addition, it lists the transmit and receive states the data terminal equipment (DTE) connector on SuperStack II NETBuilder bridge/router needs to be in to operate.

For more information about cables, modems, TAs, and telco services, refer to the WAN Cabling and Connectivity Guide. You can find this guide on the 3Com Corporation World Wide Web site by entering:

`http://www.3com.com/`

HSS Line Driver Cards

The NETBuilder II bridge/router supports all HSS cards except Rev. A on the HSS V.35/RS-232 module.

To verify that your HSS I/O module is not Rev. A (assembly number 06-107-000), enter:

```
SHow -SYS IOboardInfo
```

DTE Connector Transmit and Receive States

Table B-1 lists each DTE connector on a NETBuilder II bridge/router and the state the connector needs to be in while data is received or transmitted.

Table B-1 DTE Connector Transmit and Receive States

State	Signal	RS-232 Pin	RS-449 Pin	V.35	Signal Direction
High	DSR	6	11, 29	E	To NETBuilder II
High	DCD	8	13, 31	F	To NETBuilder II
High	DTR	20	12, 30	H	From NETBuilder II

Dial-Up Progress and Error Messages

The modem, model 42x and 52x SuperStack II NETBuilder bridge/routers, ISDN TA, or NETBuilder software may return a message indicating the reason for a call failure, the progress of a call, or the presence of an incoming call.

Software Messages for Modems

The NETBuilder II bridge/router sends the following messages to indicate call progress or failure on lines configured with modems:

```
INCOMING CALL ON PATH <path>, PORT <port>
CALL ON PATH <path>, PORT <port> REJECTED
INCOMING CALL ON PATH <path>, PORT <port> CONNECTED
CALL ON PATH <path>, PORT <port> CONNECTED
CALL ON PATH <path>, PORT <port> REJECTED, NO CARRIER
DISCONNECT ON PATH <path>, PORT <port>
PATH NOT COMING UP, INITIATING HANGUP ON PATH <path>
PRIMARY IS UP, INITIATING HANGUP ON PATH <path>
SECONDARY IS IDLE, INITIATING HANGUP ON PATH <path>
PATH IS IDLE, INITIATING HANGUP ON PATH <path>
AUTODIAL INITIATING CALL ON PATH <path>, PORT <port>
USER INITIATING CALL ON PATH <path>, PORT <port>
BOD FEATURE INITIATING CALL ON PATH <path>, PORT <port>
DR FEATURE INITIATING CALL ON PATH <path>, PORT <port>
RETRY INITIATING CALL ON PATH <path>, PORT <port>
DIALNO IS REQUIRED FOR V.25BIS CALLS ON PATH <path>
NO CALL ATTEMPTED, NO-ORIGINATE SET ON PATH <path>
CALL ON PATH <path>, PORT <port> REJECTED, CODE = <xx>
DOD INITIATING CALL ON PATH <path>, PORT <port>
DOD RETRY INITIATING CALL ON PATH <path>, PORT <port>
```

Some of these messages include a two-letter response code. This code is also displayed by the SHow -PORT DialHistory command. Table B-2 lists these modem response codes.

V.25 Modems

In response to a failed attempt, a V.25bis modem may return one of the error codes listed in Table B-2. These codes are associated with the following messages:

```
DIALNO IS REQUIRED FOR V.25BIS CALLS ON PATH <path>
CALL ON PATH <path>, PORT <port> REJECTED, CODE = <xx>
```

Table B-2 Modem Response Codes

Response Code	Meaning
AB	Abort call
CB	Local DCE busy
ET	Engaged Tone
FC	Forbidden call
NS	Number not stored
NT	Answer tone not detected
RT	Ring tone

Software Messages for SuperStack II NETBuilder Bridge/Router

Table B-2 provides error codes and messages the model 42x and 52x SuperStack II NETBuilder bridge/routers may return in response to failed attempts to communicate.

Table B-3 ISDN Dial Failure Cause Codes

Cause Number*	Cause	Definition†
1	Unallocated (unassigned) number	Indicates that the destination requested by the calling user cannot be reached because, although the number is in a valid format, it is not currently assigned (allocated).
2	No route to specified transit network	Indicates that the equipment sending this cause has received a request to route the call through a particular transit network, which it does not recognize. The equipment sending this cause does not recognize the transit network either because the transit network does not exist or because that particular transit network, while it does exist, does not service the equipment that is sending this cause. This cause is supported on a network-dependent basis.
3	No route to destination	Indicates that the called user cannot be reached because the network through which the call has been routed does not serve the destination desired. This cause is supported on a network-dependent basis.
6	Channel unacceptable	Indicates the channel most recently identified is not acceptable to the sending entity for use in this call.
7	Call awarded and being delivered in an established channel	Indicates that the user has been awarded the incoming call, and that the incoming call is being connected to a channel already established to that user for similar calls.
16	Normal call clearing	Indicates that the call is being cleared because one of the users involved in the call has requested that the call be cleared. Under normal situations, the source of this cause is not the network.
17	User busy	Used when the called user has indicated the inability to accept another call. The user equipment is compatible with the call.
18	No user responding	Used when a user does not respond to a call establishment message with either an alerting or connect indication within the prescribed period of time allocated.
19	No answer from user (user alerted)	Used when a user has provided an alerting indication but has not provided a connect indication within a prescribed period of time.
21	Call rejected	Indicates that the equipment sending this cause does not wish to accept this call, although it could have accepted the call because the equipment sending this cause is neither busy nor incompatible.
22	Number changed	Returned to a calling user when the called party number indicated by the calling user is no longer assigned. If a network does not support this capability, cause #1 “unallocated (unassigned) number” shall be used.
26	Non-selected user clearing	Indicates that the user has not been awarded the incoming call.
27	Destination out of order	Indicates that the destination indicated by the user cannot be reached because the interface to the destination is not functioning correctly. The term “not functioning correctly” indicates that a signalling message was unable to be delivered to the remote user; for example, a physical layer or data link layer failure at the remote user, user equipment off-line, and so forth.
28	Invalid number format	Indicates that the called user cannot be reached because the called party number is not in a valid format or is not complete.
29	Facility rejected	Returned when a facility requested by the user cannot be provided by the network.
31	Normal, unspecified	Used to report a normal event only when no other cause in the normal class applies.
34	No circuit/channel available	Indicates that there is no circuit/channel presently available to handle the call.
38	Network out of order	Indicates that the network is not functioning correctly and that the condition is likely to last a relatively long period of time; for example, immediately retrying the call is not likely to be successful.
41	Temporary failure	Indicates that the network is not functioning correctly and that the condition is not likely to last a long period of time; for example, the user can try another call attempt almost immediately.

(continued)

Table B-3 ISDN Dial Failure Cause Codes (continued)

Cause Number*	Cause	Definition†
42	Switching equipment congestion	Indicates that the switching equipment generating this cause is experiencing a period of high traffic.
43	Access information discarded	Indicates that the network could not deliver access information to the remote user as requested; for example, a user-to-user information, low layer compatibility, high layer compatibility, or subaddress.
44	Requested circuit/channel not available	Returned when the circuit or channel indicated by the requesting entity cannot be provided by the other side of the interface.
47	Resources unavailable, unspecified	Used to report a resource unavailable event only when no other cause in the resource unavailable class applies.
49	Quality of service not available	Used to report that the requested quality of service, as defined in CCITT Recommendation X.213, cannot be provided (for example, throughput or transit delay cannot be supported).
50	Requested facility not subscribed	Indicates that the requested supplementary service could not be provided by the network because the user has not completed the necessary administrative arrangements with its supporting network.
57	Bearer capability not authorized	Indicates that the user has requested a bearer capability that is implemented by the equipment, which generated this cause, but the user is not authorized to use.
58	Bearer capability not presently available	Indicates that the user has requested a bearer capability that is implemented by the equipment, which generated this cause, but which is not available at this time.
63	Service or option not available, unspecified	Used to report a service or option not available event only when no other cause in the service or option not available class applies.
65	Bearer capability not implemented	Indicates that the equipment sending this cause does not support the bearer capability requested.
66	Channel type not implemented	Indicates that the equipment sending this cause does not support the channel type requested.
69	Requested facility not implemented	Indicates that the equipment sending this cause does not support the requested supplementary service.
70	Only restricted digital information bearer capability is available	Indicates that an equipment has requested an unrestricted bearer service but that the equipment sending this cause only supports the restricted version of the requested bearer capability.
79	Service or option not implemented, unspecified	Used to report a service or option not implemented event only when no other cause in the service or option not implemented class applies.
81	Invalid call reference value	Indicates that the equipment sending this cause has received a message with a call reference that is not currently in use on the user-network interface.
82	Identified channel does not exist	Indicates that the equipment sending this cause has received a request to use a channel not activated on the interface for a call. For example, if a user has subscribed to those channels on a primary rate interface numbered from 1 to 12 and the user equipment or the network attempts to use channels 13 through 23, this cause is generated.
83	A suspended call exists, but this call identity does not	Indicates that a call resume has been attempted with a call identity, which differs from that in use for any presently suspended call(s).
84	Call identity in use	Indicates that the network has received a call suspend request. The call suspend request contained a call identity (including the null call identity) that is already in use for a suspended call within the domain of interfaces over which the call might be resumed.
85	No call suspended	Indicates that the network has received a call resume request. The call resume request contained a call identity information element that presently does not indicate any suspended call within the domain of interfaces over which calls may be resumed.

(continued)

Table B-3 ISDN Dial Failure Cause Codes (continued)

Cause Number*	Cause	Definition†
86	Call having the requested call identity has been cleared	Indicates that the network has received a call resume request. The call resume request contained a call identity information element that once indicated a suspended call; however, that suspended call was cleared while suspended (either by network timeout or by the remote user).
88	Incompatible destination	Indicates that the equipment sending this cause has received a request to establish a call that has low layer compatibility, high layer compatibility, or other compatibility attributes (for example, data rate), which cannot be accommodated.
91	Invalid transit network selection	Indicates that a transit network identification was received that is of an incorrect format.
95	Invalid message, unspecified	Used to report an invalid message event only when no other cause in the invalid message class applies.
96	Mandatory information element is missing	Indicates that the equipment sending this cause has received a message that is missing an information element, which must be present in the message before that message can be processed.
97	Message type non-existent or not implemented	Indicates that the equipment sending this cause has received a message with a message type it does not recognize either because this is a message not defined or defined but not implemented by the equipment sending this cause.
98	Message not compatible with call state or message type non-existent or not implemented	Indicates that the equipment sending this cause has received a message such that the procedures do not indicate that this is a permissible message to receive while in the call state, or a STATUS message was received indicating an incompatible call state.
99	Information element non-existent or not implemented	Indicates that the equipment sending this cause has received a message, which includes information elements not recognized because the information element identifier is not defined or it is defined but not implemented by the equipment sending the cause. However, the information element is not required to be present in the message in order for the equipment sending the cause to process the message.
100	Invalid information element contents	Indicates that the equipment sending this cause has received an information element that it has implemented; however, one or more of the fields in the information element are coded in such a way that has not been implemented by the equipment sending this cause. One reason this cause may occur is that the bridge/router requested 56K rate adaption in auto-rate mode, but the switch does not support 56K rate adaption.
101	Message not compatible with call state	Indicates that a message has been received, which is incompatible with the call state.
102	Recovery on timer expiry	Indicates that a procedure has been initiated by the expiry of a timer in association with ETS 300 102-1 error handling procedures.
111	Protocol error, unspecified	Used to report a protocol error event only when no other cause in the protocol error class applies.
127	Interworking, unspecified	Indicates that there has been interworking with a network that does not provide causes for the actions it takes; the exact cause for a message that is being sent cannot be determined.

* These cause numbers and definitions are found in ETS300 102-1, CCITT Q931 specification.

† In response to the error code, the dial-up software may select a new path, may try another phone number, or may keep the same path and number when retrying the call.

C LOOPBACK TESTING

Very few modems or terminal adapters support loopback initiated by controlling the LL and RL pins on the data terminal equipment (DTE)/data communications equipment (DCE) interface. These loopback functions can be accomplished by setting the DLTest TestMode to Loopback and manually configuring the modems for the desired loopback type. This appendix describes how to set up a dial-up loopback test.

Dial-up Loopback Testing Using Modems

The following information describes how to perform a loopback test on a dial-up device and supplements the information currently found under the DLTest command in the Chapter 1 in *Reference for NETBuilder Family Software*. Figure C-1 shows a sample representation of the procedure.

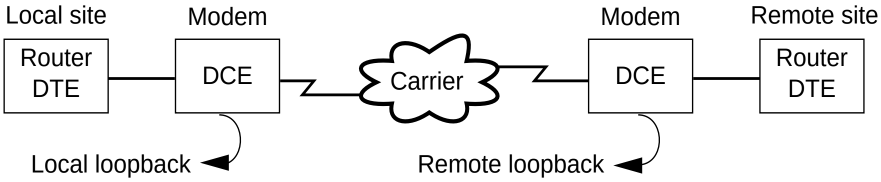


Figure C-1 Typical Dial or Leased Connection

To successfully troubleshoot your network, you need to determine the procedure your DCE uses to perform DCE loopback. There are several different ways DCEs use loopback. Check your DCE vendor manual, and identify the ways you can activate the DCE to perform the following functions:

- Local loopback

  A loopback is made at the local DCE with the transmit data being returned on the receive data to the DTE (router).
- Call or connection

  A connection is made to a remote DCE.
- Remote loopback

  A loopback is made by a far-end DCE where the received data is being returned on the transmit path, back to the near-end DCE.

A DCE can be configured in one or more of the following ways:

- Front panel switches
- Console control through an asynchronous terminal connected to a separate control port
- Signal line control of V.24 circuits
- AT commands sent over a combination data and control port

Most DCEs do not support all of these modes of operation. To make sure which type of control your DCE offers, refer to your DCE vendor's documentation.

For the following information, refer to the "Commands" chapter in *Reference for NETBuilder Family Software*.

- DLTest command
- High-speed serial (HSS) software settings
- Internal clock setting
- Leased-line setting

For specific settings, refer to the DCE manufacturer's documentation.

Before conducting a loopback test, check the basic network connectivity by following these steps:

1 Check the connections between the DCE and the telephone network.

2 Verify both DCE configuration settings.

Both DCEs should support common communication standards.

If you still experience problems, troubleshoot the connection to isolate the problem.

When performing the following tests, use commands or methods that are specific to each DCE.

Performing a Local Loopback Test

To conduct a local loopback test, follow these steps (see Figure C-2):

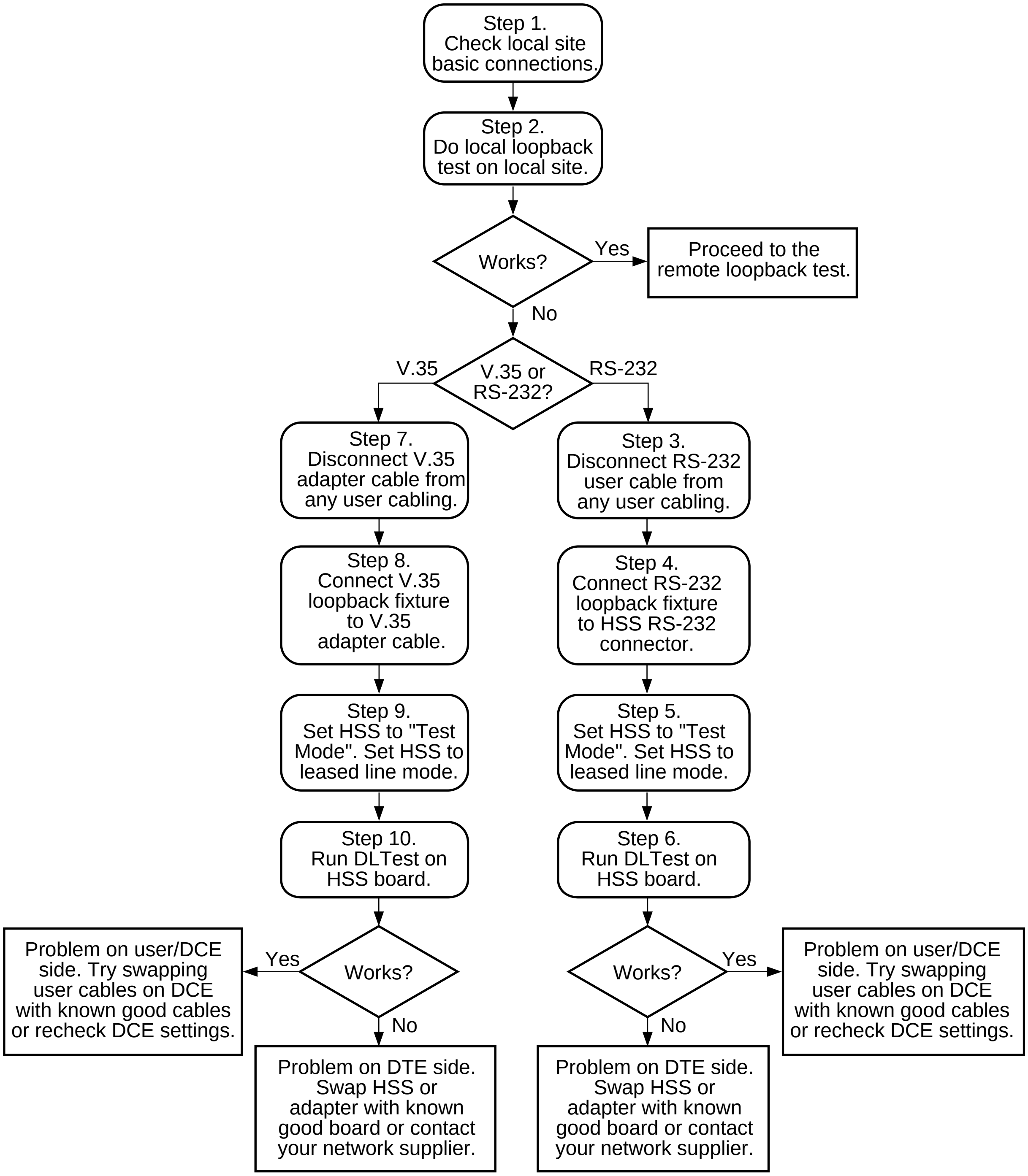

Step 1.
Check local site basic connections.
Step 2.
Do local loopback test on local site.
Works?
Yes
Proceed to the remote loopback test.
No
V.35 or RS-232?
V.35
RS-232
Step 7.
Disconnect V.35 adapter cable from any user cabling.
Step 8.
Connect V.35 loopback fixture to V.35 adapter cable.
Step 9.
Set HSS to "Test Mode". Set HSS to leased line mode.
Step 10.
Run DLTest on HSS board.
Works?
Yes
Problem on user/DCE side. Try swapping user cables on DCE with known good cables or recheck DCE settings.
No
Problem on DTE side. Swap HSS or adapter with known good board or contact your network supplier.
Step 3.
Disconnect RS-232 user cable from any user cabling.
Step 4.
Connect RS-232 loopback fixture to HSS RS-232 connector.
Step 5.
Set HSS to "Test Mode". Set HSS to leased line mode.
Step 6.
Run DLTest on HSS board.
Works?
Yes
Problem on user/DCE side. Try swapping user cables on DCE with known good cables or recheck DCE settings.
No
Problem on DTE side. Swap HSS or adapter with known good board or contact your network supplier.

Figure C-2 HSS V.35 and RS-232 Local Loopback Flowchart

1 Check local site basic configurations.

2 At the local DCE, initiate the loopback mode.

a Set the HSS board to Leased mode.

b Run the loopback mode on the HSS board using the DLTest command.

If this test passes, proceed to the remote loopback procedure in the next section.

If the loopback test fails and you are using a V.35 interface, proceed to step 7.

If the loopback test fails and you are using an RS-232 interface, proceed to step 3.

3 Disconnect the RS-232 user cable from the HSS board.

4 Connect the RS-232 loopback fixture to the HSS board (refer to Table C-1 and Table C-2).

5 Set the HSS board to internal clock source and leased-line mode.

6 Run the loopback mode on the HSS board using the DLTest command.

If the test passes, the DCE or the user cables are faulty. Try using a known good DCE or user cable. After you isolate the connection problem, remember to change back the HSS board settings.

If the test fails, try using a known good HSS board or contact 3Com.

7 Disconnect the V.35 user cable from the HSS board.

8 Connect the V.35 loopback fixture to the HSS board (refer to Table C-1 and Table C-2).

9 Set the HSS board to Test Mode and leased-line mode.

10 Run the loopback mode on the HSS board using the DLTest command.

If the test passes, the DCE or the user cables are faulty. Try using a known good DCE or user cable. After you isolate the connection problem, remember to change back the HSS board settings.

If the test fails, try using a known good HSS board or contact 3Com.

Performing a Remote Loopback Test

To perform a remote loopback test, follow these steps (see Figure C-3). For DCE configuration information, refer to your vendor documentation.

This test is between modem and modem.

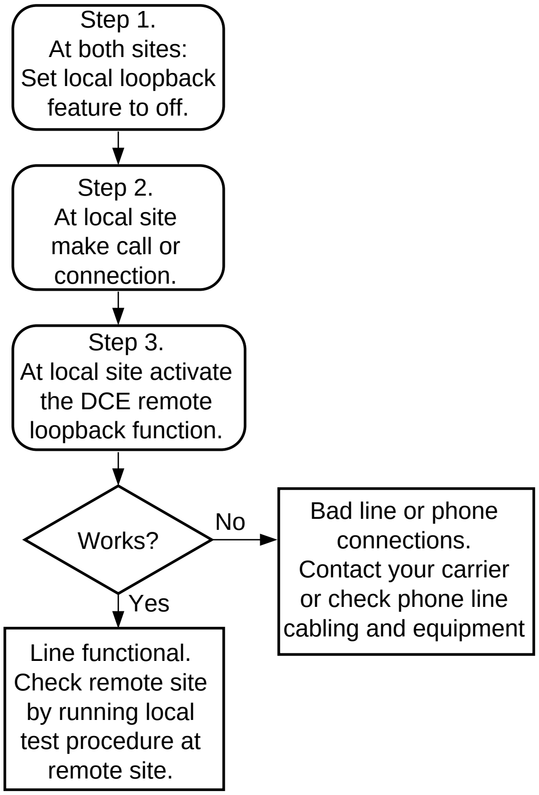


Figure C-3 HSS V.35 and RS-232 Remote Loopback Flowchart

1. At both sites, turn the local loopback feature off.
2. At the local site, make the DCE call the remote DCE.
3. At the local DCE, start a remote loopback test using a self-generated pattern.

   If the test passes, the line and remote DCE are functional. Run the local test procedure at the remote site. After you isolate the connection problem, remember to change back the HSS board settings.

   If the test fails, the line or remote DCE, or phone connections at either the local or remote site are faulty; recheck the phone, cabling, DCE phone line settings, and/or contact your carrier.

Making the Loopback Fixture

To make the RS-232 loopback fixture, follow these steps (refer to Table C-1):

1. Obtain a male RS-232 connector.
2. Wire the pins according to Table C-1.

Table C-1 RS-232 Loopback Pin Assignments

Name	Pin	Name	Pin
TD	2	RD	3
RTS	4	CTS	5
DSR	6	DCD DTR	8 20
RXC	17	TT	24

To make the V.35 loopback fixture, follow these steps (refer to Table C-2):

1 Obtain a male V.35 connector.

2 Wire the pins according to Table C-2.

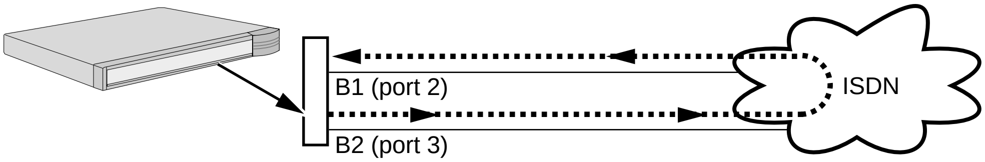
Table C-2 V.35 Loopback Pin Assignments

Name	Pin	Name	Pin
SCTEA	U	SCRA	V
SCTEB	W	SCRB	X
RTS	C	CTS	D
DTR	H	DSR DCD	E F
SCTA	Y		
SCTB	AA		

ISDN Loopback Testing

This section describes how to perform a loopback test using two B channels on one Integrated Services Digital Network (ISDN) line.

Figure C-4 shows the data flow when performing a loopback test using the two B channels of an ISDN basic rate interface (BRI) line. Both of these channels occupy the same physical connector and no modem is required in this configuration, however, the unit must be attached to an ISDN line.



Both channels (B1 and B2) occupy the same physical connector.

Figure C-4 ISDN Loopback Testing

Procedure

To run the loopback diagnostics test, you must have console running at 9,600 baud connected to your SuperStack II NETBuilder bridge/router.

To set up the loopback test, follow these steps:

1 Set the path line type to Dialup using:

```
SETDefault !<path> -PATH LineType = Dialup
```

To perform the test shown in the example, enter the following commands:

```
SETDefault !2.1 -PATH LineType = Dialup
SETDefault !2.2 -PATH LineType = Dialup
```

2 Set the rate adaption parameter to automatically detect the speed of the interface by using:

```
SETDefault !<path> -PATH RateAdaption = Auto
```

In the example in step 1, the test originates from path 2.2 and targets path 2.1. To specify this for path 2.2, enter:

```
SETDefault !2.2 -PATH RateAdaption = Auto
```

3 Set the switch type using:

```
SETDefault !<path> -PATH SwitchType = ETSI | NTT | ATT5ESS | NT1
| DMS100 |KDD
```

To set the switch type to ETSI, enter:

```
SETDefault !2 -PATH SwitchType = ETSI
```

4 Establish the local dial numbers for the bearer channels using the following syntax:

```
SETDefault !<port> -PATH LocalDialNo = “<string>”
```

To establish the local dial numbers for the two bearer channels, enter:

```
SETDefault !2.1 -PATH LocalDialNo = “4962124”
SETDefault !2.2 -PATH LocalDialNo = “4962125”
```

5 Configure the ports for loopback testing using:

```
SETDefault !<port> -PORT OWNer = Loopback
```

Enter Loopback as the owner on both the sending and receiving ports; for example:

```
SETDefault !2 -PORT OWNer = Loopback
SETDefault !3 -PORT OWNer = Loopback
```

6 Establish a connection between the two bearer channels by dialing out on one channel and dialing into the other using:

```
DIal !<path> “<string>”
```

To dial port 2 from port 3, enter:

```
DIal !2.2 “4962124”
```

Path 2.2 places a call to the specified number, which is the number for path 2.1. It is not important which port originates or answers the call as long as the port does not try to call itself.

7 When the connection is successfully established, select the loopback testing mode by entering:

```
DLTest TestMode Loopback
```

You can specify the number of seconds the test should run. You can enter this value any time before entering the DLTest START command. If a value is not specified, an infinite time duration is assumed. To run the test for a specific number of seconds, use:

```
DLTest TestDuration <seconds>
```

Use caution when running the test for a specified test duration. The test ends abruptly as soon as the time duration expires, and a discrepancy between the number of packets transmitted and the number received may result.

8 Start the DLtest using:

```
DLTest Start <sendingport>, <receivingport>
```

To start the DLTest and designate port 2 to send the DLTest data and port 3 to receive and loop back the data, enter:

```
DLtest START 2,3
```

The loopback test is successful when the number of received packets equals or approximately equals the number of transmitted packets. If the test is not successful, verify that your system is cabled correctly and your model is installed correctly. You can check the number of packets transmitted and the number of errors by entering:

```
DLTest Stat
```

9 Stop the DLTest by entering:

```
DLTest Abort
```

10 Disconnect the call by entering:

```
HangUp !2.2
```

11 Change the port owner from Loopback to the original owner using:

```
SETDefault !<port> -PORT OWNer = PPP
```

INTERNET ADDRESSING

This appendix provides information about the following topics:

- Internet addresses and classes
- Dotted decimal notation
- Addressing rules
- Subnet addressing and subnet masks
- Variable length subnet masks

Internet Addresses

Any universal communications system requires a globally accepted method of identifying individual computers; one globally accepted method is to have the Network Manager assign unique Internet addresses to devices, or *hosts*, on the Internet. These hosts can be personal computers, communications servers, ports on a communications server, internetwork bridges, network control servers, or UNIX hosts. The Internet uses these assigned addresses when sending or receiving packets.

You can obtain valid and unique Internet addresses through the InterNIC Registration Services. For additional information, refer to the New Installation for NETBuilder II Software.

The Internet Protocol (IP) uses Internet addresses. Internet addressing uses a 32-bit address field numbered 0 to 31. This address field is composed of two parts: one part identifies the network on which the host resides, and the second part identifies the host itself. Hosts attached to the same network must share a common prefix designating their network number. Conceptually, each address is a pair (net#, host#) where *net#* identifies the network, and *host#* identifies a host on that network.

Internet addressing is divided into four classes: A, B, C, and D. Each address class begins with a unique bit pattern that is used by the Internet software residing on network hosts to identify the address class. Once the software has identified the address class from the leading bits of the Internet address, it can determine which bits are used to represent the network number and which bits are used to identify the host portion of the address. The next four sections describe the four Internet address classes.

Class A Address Format

The first type of address, Class A, has a 7-bit network field and a 24-bit local address. The highest-order bit is set to 0. This allows 127 Class A networks to be defined (network number 0 is not allowed).

The following diagram describes the format of a Class A address with the bit numbers on the first line:

0	1 7	8 31
0	Network	Local Host Address

Class B Address Format

The second type of address, Class B, has a 14-bit network field and a 16-bit local address. The two highest-order bits are set to 1 and 0. This allows 16,383 Class B networks to be defined.

The following diagram describes the format of a Class B address with the bit numbers on the first line:

0 1	2 15	16 31
1 0	Network	Local Host Address

Class C Address Format

The third type of address, Class C, has a 21-bit network field and an 8-bit local address. The three highest-order bits are set to 1, 1, and 0. This allows 2,097,151 Class C networks to be defined.

The following diagram describes the format of a Class C address, with the bit numbers on the first line:

0 1 2	3 23	24 31
1 1 0	Network	Local Host Address

Class D Address Format

The fourth type of address, Class D, has a 23-bit multicast address field. The four highest-order bits are set to 1, 1, 1, and 0. This allows 8,000,000 multicast addresses to be defined.

The following diagram describes the format of a Class D address, with the bit numbers on the first line:

0 1 2 3	4 5 *	6 31
1 1 1 0	0 0	Multicast Address

* Not used.

No addresses are allowed with the four highest-order bits set to 1-1-1-1. These addresses are reserved.

Dotted Decimal Notation

An Internet address is specified as four decimal numbers, each separated by a dot. This format is called *dotted decimal notation*. The 32-bit Internet address is divided into four 8-bit fields, called octets; the value of each field is specified as a decimal number with the fields separated by periods.

For example, the Internet address of USC-ISIB.ARPA in binary octets and dotted decimal notation is as follows:

Binary octets: 00001010 00000011 00000000 00110100

Dotted decimal notations: 010.003.000.052 or 10.3.0.52

Valid network numbers for each address class are provided. The “nnn” represents the network portion of the address, which is assigned by the InterNIC. The “hhh” represents the host portion of the address, which is assigned by the network manager.

Class A networks: (nnn.hhh.hhh.hhh):001.hhh.hhh.hhh through 126.hhh.hhh.hhh

Class B networks: (nnn.nnn.hhh.hhh):128.001.hhh.hhh through 191.254.hhh.hhh

Class C networks: (nnn.nnn.nnn.hhh):192.000.001.hhh through 223.255.254.hhh

Class D networks: 224.000.000.000 through 239.255.255.255

The bits defining the local address portion of an Internet address cannot be all zero bits or all 1 bits. These are special addresses and are described in “Addressing Rules” on page D-3.

Addressing Rules

These general guidelines should be observed when assigning Internet addresses:

- The bits used to define the host portion of an Internet address should not be all one bits.

  According to the standard, any Internet address with the host portion consisting of all ones is interpreted as meaning “all,” as in “all hosts.” For example, the address 128.1.255.255 is interpreted as meaning all hosts on network 128.1 and is reserved for directed broadcast addressing.

- The bits used to define the network portion of an Internet address should not be all zero bits.

  According to the standard, any Internet address with the network portion consisting of all zeros is interpreted as meaning “this,” as in “this network.” For example, the address 0.0.0.63 is interpreted as meaning host 63 on this network.

- The class A network number 127 is assigned the “loopback” function.

  The loopback function allows a datagram to be sent by a higher-level protocol to network 127 to loopback inside the host. For example, in a Berkeley Software Distribution (BSD) UNIX environment, if program A needs to communicate with program B, which is running on the same machine, they can do this with IP network number 127. However, no datagram should ever appear on any network with a source or destination network address of 127.

- No addresses are allowed with the four highest-order bits set to 1111.

  These addresses are reserved for Class E networks.

Sample Network Using the Class B Address Format

Figure D-1 shows a sample network using the Class B address format. As shown in this illustration, segments connected by internetwork bridges share the same network fields while having different host fields. For example, the bridge segments Network A into two LANs, both of which have the network address of 128.001.

Segments interconnected by routers must have different network fields to be physically separate networks. For example, the router physically separates Network A from Network B as indicated by the different network addresses (Network A has a network address of 128.001 and Network B has a network address of 128.002).

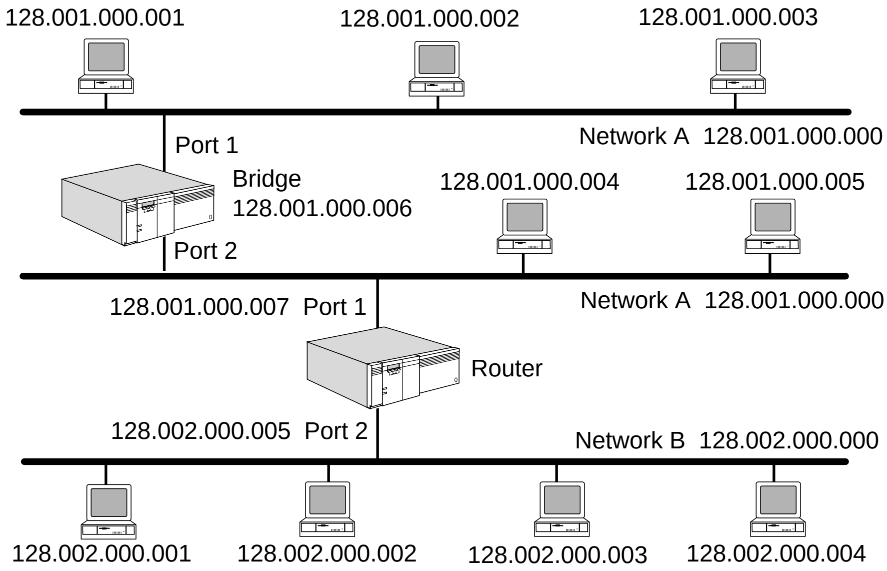


Figure D-1 Sample Network Using the Class B Address Format

Subnet Addresses and Subnet Masks

The original interpretation of Internet addresses (described in the previous sections) was based on a two-level hierarchy. In this model, each host sees its network as a single entity.

A number of organizations have added a third level to the interpretation of Internet addresses. In this structure, a given Internet network is divided into a collection of subnets. The three-level model is useful in networks in moderately large organizations, where it is often necessary to use more than one LAN cable to cover a local area. Each LAN can then be treated as a subnet belonging to a given main Internet network number. These independent networks are then connected by routers. However, each organization that wants to connect to the Internet can usually obtain only a single Internet number.

Subnet Addressing

If multiple Transmission Control Protocol/Internet Protocol (TCP/IP) networks are interconnected across routers, you must assign a different network field to each network. (However, if the network is part of the Internet, you cannot use different network fields because the network field must be assigned by the InterNIC.) Subnet addressing allows an organization to use a single Internet network number for multiple physical networks. Subnets can be used with any class of Internet addressing except Class D.

In Figure D-2, a site with two physical networks uses subnet addressing to span them with a single class B network address. The router accepts all traffic for network 128.005.000.000 and chooses a physical network based on the third octet of the address.

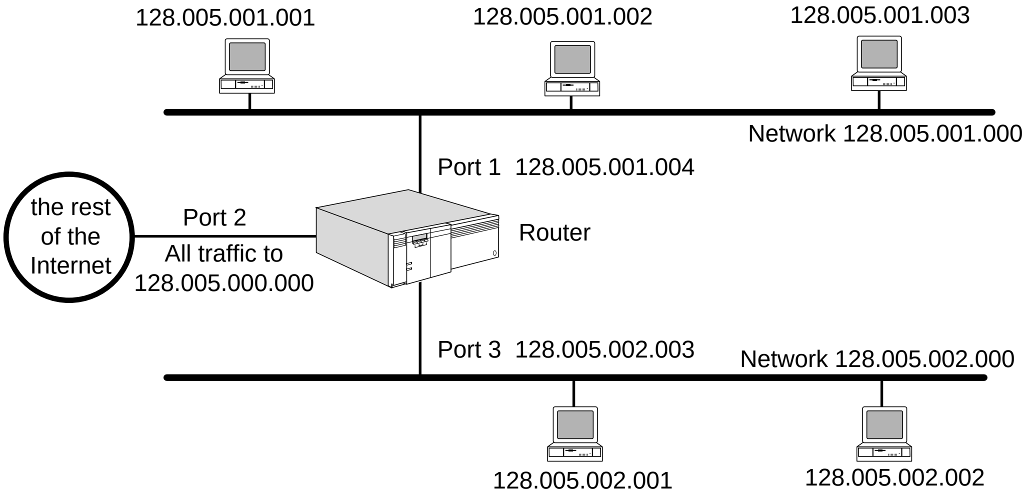


Figure D-2 Subnet Addressing

The format of a regular Internet address and an Internet address with subnet mask are as follows:

Regular Internet Address Format

Network Number	Host Number

Subnet Address Format

Network Number	Subnet Number	Host Number

The network field is already defined in the previous section. The width of the subnet field is constant for a given network number.

For example, on a Class B network with a 6-bit-wide subnet field, an Internet address can be broken down as follows:

0 1	2 15	16 21	22 31
1 0	Network	Subnet	Local Host Address

The subnet portion of an Internet address cannot be defined as all 1 bits; the host portion of an Internet address with the preceding definition can not be defined as all 1 bits.

In the preceding example, the subnet field can have any value between 0 and 62, and the host field can have any value between 1 and 1022 (all numbers are decimal). A typical class B Internet address that fits the requirements of the preceding example is the Internet address 128.5.61.100 with a subnet mask of 255.255.252.0.

Subnet Masks

The subnet mask allows the host portion of an Internet address to be divided into two parts. One part is used to identify a physical subnet, and the second part is used to identify a host on that subnet.

Bits in the subnet mask are set to 1 if the network treats the corresponding bit in the Internet address as part of the network address. Bits in the subnet mask are set to 0 if it treats the bit as part of the host identifier.

Subnet Address Format

Network Number	Subnet Number	Host Number

Subnet Mask

11111111 11111111	11111111	00000000

The subnet mask is also defined in the dotted decimal notation. For example, with a Class B address of 128.121.61.100, the subnet mask is as follows:

Subnet Address Format for 128.121.61.100

10000000 01111001	00111101	01100100

Subnet Mask

11111111 11111111	11111111	00000000

The subnet mask for the preceding example would then be 255.255.255.0.

Adhering to the preceding constant width requirement, when using RIP the value of the subnet mask should be the same on all subnets defined for a given network number. 3Com bridge/routers support variable length subnet masks when using OSPF. For more information, refer to "Variable Length Subnet Masks" on page D-10.

Subnet masks are assigned using the NETaddr parameter in the IP Service. For example, to assign the IP address of 128.005.001.001 with a subnet mask of 255.255.255.000 to port 1 of a router, enter:

```
SETDefault !1 -IP NETaddr = 128.005.001.001 255.255.255.000
```

For more information on the NETaddr parameter, refer to Chapter 29 in *Reference for NETBuilder Family Software*.

Subnets: Example 1

The InterNIC assigns you a Class B Internet address of 128.001.000.000. You need to establish 254 subnets with each subnet capable of supporting up to 254 hosts. This is the simplest form of subnetting. The first and second octets of the IP address identify the network, the third octet identifies the subnet, and the fourth octet identifies a host on the subnet.

To solve this problem, follow these steps:

1 Convert the address assigned by the InterNIC to binary format.

For example:

128.001.000.000 = <u>10000000 00000001</u> 00000000 00000000

The underlined binary digits represent the network portion of the Internet address assigned by the InterNIC.

2 Determine the number of binary digits you need to represent 254 subnets.

Because eight binary digits are required to define 254 subnets ($2^8 = 256$), the subnets can be numbered 1 through 254. The values of all zeros and all ones in the subnet field should not be assigned to the actual (physical) subnets (subnets 0 and 255 are not valid).

Decimal	Binary
1	00000001
2	00000010
...	...
254	11111110

3 Select the eight most significant bits of the host portion of the Internet address to define the subnets.

These bits are displayed in bold text:

128.001.000.000 = 10000000.00000001.**00000000**.00000000

4 Define a subnet mask so that all bits of the network and future subnet fields are set to 1, and all bits of the future host field are set to 0.

Network #: 0000000.00000001.**00000000**.00000000 =128.001.000.000

Subnet Mask:11111111.11111111.11111111.00000000 = 255.255.255.000

This subnet mask (255.255.255.000) must be configured on each host and defined for each router. Use the same subnet mask for devices on the same subnetted subnet that share the same Internet address.

5 Determine the subnet address for each host.

The 254 subnets have the following addresses:

Subnet #1:10000000.00000001**.00000001**.00000000 = 128.001.001.000

Subnet #2:10000000.00000001**.00000010**.00000000 = 128.001.002.000

Subnet #3:10000000.00000001**.00000011**.00000000 = 128.001.003.000

. . .

Subnet #254:10000000.00000001.**11111110**.00000000 =128.001.254.000

The range of addresses that you can assign for subnet #1 are as follows:

Subnet #1:10000000.00000001.00000001.00000000 = 128.001.001.000

Low Address:10000000.00000001.00000001.00000001 = 128.001.001.001

High Address:10000000.00000001.00000001.11111110 = 128.001.001.254

The range of addresses that you can assign for subnet #35 are as follows:

Subnet #35:10000000.00000001.00100011.00000000 = 128.001.035.000

Low Address:10000000.00000001.00100011.00000001 = 128.001.035.001

High Address:10000000.00000001.00100011.11111110 = 128.001.035.254

The range of addresses that you can assign for subnet #129 are as follows:

Subnet #129:10000000.00000001.10000001.00000000 = 128.001.129.000

Low Address:10000000.00000001.10000001.00000001 = 128.001.129.001

High Address:10000000.00000001.10000001.11111110 = 128.001.129.254

6 Assign the Internet address to the bridge/router.

For example, if subnet #1 is connected to bridge/router port #1, to assign the Internet address you can enter:

```
SETDefault !1 -IP NETaddr = 128.001.001.001 255.255.255.000
```

Subnets: Example 2

The InterNIC assigns you a Class B Internet address of 128.001.000.000. You need to establish two subnets with each subnet capable of supporting up to 16,381 hosts.

To solve this problem, follow these steps:

1 Convert the address assigned by the InterNIC to binary format.

For example:

128.001.000.000 = 10000000.00000001.00000000.00000000

The underlined binary digits represent the network portion of the Internet address assigned by the InterNIC.

2 Determine the number of binary digits you need to represent two subnets.

Because two binary digits are required to define 2 subnets (2^2 = 4), the subnets can be numbered 1 through 2. The values of all zeros and all ones in the subnet field should not be assigned to the actual (physical) subnets (subnets 0 and 3 are not valid).

Decimal	Binary
1	00000001
2	00000010

3 Select the two most significant bits of the host portion of the Internet address to define the subnets.

These bits are displayed in bold text:

128.001.000.000 = 10000000.00000001.**00**000000.00000000

4 Define a subnet mask so that all bits of the network and future subnet fields are set to 1, and all bits of the future host field are set to 0.

Network #:10000000.00000001.**00**000000.00000000 = 128.001.000.000

Subnet Mask:11111111.11111111.**11**000000.00000000 = 255.255.192.000

This subnet mask (255.255.192.000) must be configured on each host and defined for each router. You should use the same subnet mask for devices on the same subnetted subnet that share the same Internet address.

5 Determine the subnet address for each host.

The two subnets have the following addresses:

Subnet #1:10000000.00000001.**01**000000.00000000 = 128.001.064.000

Subnet #2:10000000.00000001.**10**000000.00000000 = 128.001.128.000

The range of addresses that you can assign for subnet #1 are as follows:

Subnet #1:10000000.00000001.01000000.00000000 = 128.001.064.000

Low Address:10000000.00000001.01000000.00000001 = 128.001.064.001

High Address:10000000.00000001.01111111.11111110 = 128.001.127.254

The range of addresses that you can assign for subnet #2 are as follows:

Subnet #1:10000000.00000001.10000000.00000000 = 128.001.128.000

Low Address:10000000.00000001.10000000.00000001 = 128.001.128.001

High Address:10000000.00000001.10111111.11111110 = 128.001.191.254

6 Assign the Internet address to the bridge/router.

For example, if subnet #1 is connected to bridge/router port #2, to assign the Internet address you can enter:

```
SETDefault !2 -IP NETaddr = 128.001.064.001 255.255.192.000
```

Subnets: Example 3

The InterNIC assigns you a Class B Internet address of 128.001.000.000. You need to establish six subnets with each subnet capable of supporting up to 8,190 hosts.

To solve this problem, follow these steps:

1 Convert the address assigned by the InterNIC to binary format.

For example:

128.001.000.000 = 10000000.00000001.00000000.00000000

The underlined binary digits represent the network portion of the Internet address assigned by the InterNIC.

2 Determine the number of binary digits you need to represent six subnets.

Because three binary digits are required to define 6 subnets (2^3 = 8), the subnets can be numbered 1 through 6. The values of all zeros and all ones in the subnet field should not be assigned to the actual (physical) subnets (subnets 0 and 7 are not valid).

Decimal	Binary
1	00000001
2	00000010
3	00000011
4	00000100
5	00000101
6	00000110

3 Select the three most significant bits of the host portion of the Internet address to define the subnets.

These bits are displayed in bold text:

128.001.000.000 = 10000000.00000001.**000**00000.00000000

4 Define a subnet mask so that all bits of the network and future subnet fields are set to 1, and all bits of the future host field are set to 0.

Network #:10000000.00000001.**000**00000.00000000 = 128.001.000.000

Subnet Mask:11111111.11111111.**111**00000.00000000 = 255.255.224.000

This subnet mask (255.255.224.000) must be configured on each host and defined for each router. You should use the same subnet mask for devices on the same subnetted subnet that share the same Internet address.

5 Determine the subnet address for each host.

The six subnets have the following addresses:

Subnet #1:10000000.00000001.**001**00000.00000000 = 128.001.032.000

Subnet #2:10000000.00000001.**010**00000.00000000 = 128.001.064.000

Subnet #3:10000000.00000001.**011**00000.00000000 = 128.001.096.000

Subnet #4:10000000.00000001.**100**00000.00000000 = 128.001.128.000

Subnet #5:10000000.00000001.**101**00000.00000000 = 128.001.160.000

Subnet #6:10000000.00000001.**110**00000.00000000 = 128.001.192.000

The range of addresses that you can assign for subnet #3 are as follows:

Subnet #3:10000000.00000001.01100000.00000000 = 128.001.096.000

Low Address:10000000.00000001.01100000.00000001 = 128.001.096.001

High Address:10000000.00000001.01111111.11111110 = 128.001.127.254

The range of addresses that you can assign for subnet #5 are as follows:

Subnet #5:10000000.00000001.10100000.00000000 = 128.001.160.000

Low Address:10000000.00000001.10100000.00000001 = 128.001.160.001

High Address:10000000.00000001.10111111.11111110 = 128.001.191.254

6 Assign the Internet address to the bridge/router.

For example, if subnet #3 is connected to bridge/router port #1, to assign the Internet address you can enter:

```
SETDefault !1 -IP NETaddr = 128.001.096.001 255.255.224.000
```

Variable Length Subnet Masks

The 3Com bridge/router supports variable length subnet masks; more than one subnet mask (of different lengths) can be configured for a given network address. In this case, the bridge/router supports up to five different subnet masks for a given network address.

For example, on a Class B network, the following four subnet masks can be assigned for Network 128.1.0.0:

- 255.255.240.0

  FF.FF.F0.00 supports 128.1.16.0 – 128.1.224.0
- 255.255.255.0

  FF.FF.FF.00 supports 128.1.1.0 – 128.1.15.0
- 255.255.255.252

  FF.FF.FF.FC supports 128.1.0.4 – 128.1.0.12
- 255.255.255.240

  FF.FF.FF.F0 supports 128.1.0.16 – 128.1.0.240

The subnet masks are stored in the routing tables so that the longest subnet mask takes precedence over the shortest subnet mask.

E

NSAP AND PSAP ADDRESSING

This appendix provides information about network service access point (NSAP) and presentation service access point (PSAP) addressing, which applies when you are using the bridge/router for Open Systems Interconnection (OSI) routing. It provides the information necessary to establish NSAP and PSAP address values.

NSAP Address Structure

For computer equipment to communicate in a multivendor environment, each device must have a unique address. To accomplish this, a specific portion of every OSI address represents a globally unique location in the open systems environment. This part of the OSI address is the NSAP.

The parts of an NSAP address identify the individual subdomains within the global addressing domain. The NSAP address consists of two parts: the initial domain part (IDP) and the domain specific part (DSP). The IDP contains the authority and format identifier (AFI) and initial domain identifier (IDI).

Table E-1 shows the relationship between an NSAP address, IDP, and the DSP addressing scheme for the Government Open Systems Interconnection Profile (GOSIP) version II. The NSAP address contains a maximum of 20 octets or 40 decimal digits.

Table E-1 NSAP Address Structure

IDP		**DSP**						
47 1 octet	0005 2 octets	DFI 1 octet	Admin.Author. 3 octets	Resrvd. 2 octets	Routing Domain 2 octets	Area 2 octets	System 6 octets	N-SEL 1 octet

The AFI field specifies the following information:

- Addressing authority responsible for assigning values to the IDI
- IDI format (X.121, E.163, etc.)
- Whether the DSP is in binary or decimal format

Table E-2 later in this appendix lists the AFI values according to the IDI formats.

Following the AFI field is the IDI field, which identifies the network addressing authority responsible for determining the format of the DSP. For example, the GOSIP version II specification defines the DSP as containing the following information:

- Data format identifier
- Administration authority
- Reserved
- Routing domain

- Area ID
- System ID
- N-selector

NSAP Address Assignment

Different organizations may have different DSP structures and values. The U.S. government has specified its use of NSAP addresses in GOSIP; your organization can specify its own use of NSAP addresses. However, it is the network administrator's responsibility to determine the proper means for obtaining globally unique addresses.

When you assign an NSAP address to your server, follow these rules:

- The IDI is always in decimal.
- The DSP can be in decimal or hexadecimal. If the DSP is in hexadecimal, it must contain an even number of digits.
- A station ID can be either a logical ID or the unique physical address associated with some communications medium. (This type of physical address is called a Subnetwork Point of Attachment, or SNPA.)
- The preceding slash (/) in the NSAP address is mandatory. Slashes may be used as optional separators before the IDI and DSP (0005 and 01ABCDEF, respectively, in the example below). However, if one of these optional slashes is used, the other must be present also.

Example

In this example, GOSIP version II specifies the AFI to be 47, indicating that the IDI value comes from ISO 6523-ICD (International Code Designator). The U.S. GOSIP program has been assigned International Code Designator 0005 by the British Standards Institute (BSI). The Government Services Administration (GSA) can administer the values of the DSP for various U.S. federal offices.

For instance, the GSA can assign an administration authority ID to the Department of Agriculture. The Department of Agriculture can then assign routing domains to its branches (for example, different subnet IDs to branch offices in different states). Each branch office can administer the area and system IDs of its equipment.

If the AFI is an odd number, the DSP is in binary; an even AFI indicates that the DSP is in decimal.

Figure E-1 shows an example of the NSAP address using the mandatory preceding slash only.

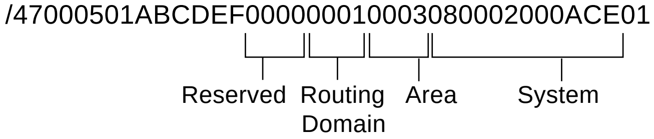


Figure E-1 NSAP Address with Mandatory Preceding Slash

The following is an example of an NSAP address using the mandatory preceding slash as well as the optional slashes:

/47/0005/01ABCDEF00000001000308000200DACE01

Default NSAP Values

3Com bridge/routers are shipped with default values for the AFI, IDI, and prefix DSP fields. Values for the ID and selector fields are generated at boot time.

The following is an example of an NSAP address assigned by 3Com:

/49/0053080002A0089D00

Table E-2 shows the values of individual fields of this NSAP address.

Table E-2 Example of an NSAP Address Assigned by 3Com

IDP		DSP			
AFI	IDI	Organization ID	Subnet ID	MAC address	N-SEL
49	Null	00	53	080002A0089D	00

For explanations of each of these fields, refer to "NSAP and PSAP Address Field Definitions" on page E-5.

Values Derived from NSAP Addresses

From an NSAP address, the following values are derived:

- Area address
- Network Entity Title (NET)

The area address is the NSAP address without the ID and selector fields. It consists of the AFI, the IDI, and the prefix of the DSP. The area address is critical to intermediate system-to-intermediate system (IS-IS) routing operations.

NET is an NSAP address with the selector value of 0. NET is used for IS-IS and Connectionless Network Protocol (CLNP) operations.

NSAP Registration Authorities

In the U.S., there are two registration authorities: the American National Standards Institute (ANSI) and the Government Services Administration (GSA). The GSA is the registration authority for all NSAP addresses that follow the U.S GOSIP version II NSAP address format. The NSAP address format is as follows:

AFI =	47 (1 octet)
IDI =	0005 (2 octets)
DSP =	Version
	DFI
	AAI
	Routing domain
	Area ID
	System ID (6 octets)
	Selector (1 octet)

You can obtain registration information by writing to the following authority:

Government Services Administration
Office of Telecommunication Services
Registration Services, Room 1221-L KBA
18th and F Street N.W.
Washington D.C. 20405

PSAP Addresses

The PSAP address contains the NSAP address and a full set of (N)-Selector: T-selector, S-selector, and P-selector.

PSAP addresses are used on the bridge/router for OSI connection services only. For normal OSI routing, use the NSAP addressing scheme.

There are three layers above the Network Layer that require addressing information: Transport, Session, and Presentation. In OSI terminology, these addresses are called (N)-selectors, where N is an OSI layer. The corresponding selectors are termed T-, S-, and P-selectors. In an open system, the combination of these selectors uniquely identifies an application entity.

The NIST Implementation Agreements specify that maximum lengths of 32, 16, and 4 octets for the T-, S-, and P-selectors be supported. Selectors for open systems from different vendors may differ in length or value.

On the 3Com OSI connection service, the P-selector portion of the PSAP address is used by the Virtual Terminal Protocol (VTP) to map to an X.25 address. The following shows the syntax for the complete PSAP address:

```
<NSAP address> | <T-SEL> | <S-SEL> | <P-SEL>
```

(N)-selectors provide the local addressing elements for accessing OSI-layer protocol processes on the server or host equipment at the destination address. In the 3Com syntax of the OSI address, (N)-selector values follow the NSAP address. Each (N)-selector field is preceded by the (|) character.

On 3Com servers, the (N)-selector values are the names of the protocol module processes operating in the respective OSI model layers. The special character "!" may be used to represent the T-selector and S-selector fields in a 3Com PSAP address. For example, the address can be in the following syntax:

```
<NSAP address>!<P-SEL>
```

The exclamation mark (!) in the syntax is interpreted as the T-selector and S-selector values. For the actual values of these selectors in a 3Com address, refer to "NSAP and PSAP Address Field Definitions" on page E-5.

You cannot assume that the (N)-selector values of the destination server or host are the same as those on the local system. Other environments may choose to omit the use of (N)-selectors or use simple numeric values. If the destination server or host does not have or need a full set of (N)-selectors, the absence of an (N)-selector must be indicated with an empty field.

The following example specifies a PSAP address with absent transport (T-SEL) and session (S-SEL) layer selector values:

```
<NSAP address> | | | <P-SEL>
```

NSAP and PSAP Address Field Definitions

This section describes each field in the NSAP and/or PSAP address relevant to bridge/router operation:

AFI The Authority and Format Identifier contains two decimal digits. In 3Com syntax, this field is always preceded with a slash (/), which identifies the NSAP portion of an NSAP or PSAP address. The AFI specifies the official body responsible for allocating IDI field values, the format of the IDI field, and whether the syntax of the DSP should be specified with binary or decimal digits.

IDI The Initial Domain Identifier contains up to 15 decimal digits depending on IDI format established in the AFI field. In 3Com syntax, this field may be preceded with a slash(/). It identifies the network addressing authority responsible for determining the format of the DSP field. The IDI field always follows the AFI field.

DSP The format and length of the Domain Specific Part is determined by the combined AFI and IDI fields. In 3Com syntax, this field may be preceded with a slash (/). In one case, the DSP field may contain the organization ID, network number or subnet ID, and MAC address fields to provide additional levels of addressing for networks such as those described in the GOSIP specification. In another case, it may contain an Internet or Ethernet address for local OSI networks.

N-SEL The length of the N-selector field is always a single octet. On 3Com servers or bridge/routers, the value of this field is a one. 3Com servers and bridge/routers use this field to identify the client of the Network layer of the OSI model, which is always the OSI Transport protocol. A special value of 0 is used to identify the network entity itself, which forms the Network Entity Title (NET).

T-SEL The T-selector field contains up to 32 octets according to the NIST agreements. The T-selector values are vendor-dependent. 3Com servers use "S" and "E" as the first and second octets in this field, respectively, to identify the ISO Session Protocol. 3Com servers use this field to identify the client of the Transport layer of the OSI model.

S-SEL The S-selector field contains up to 16 octets according to the NIST agreements. 3Com servers use "P," "R," and "E" as the first, second, and third octets in this field, respectively, to identify the ISO Presentation Protocol. 3Com servers use this field to identity the client of the Session Layer of the OSI model.

P-SEL The P-selector field in the Presentation Address is 2 octets in length for the 3Com OSI Connection Service and the value of the first octet must be either 0 or 4. When you want to make a connection using Telnet profiles, 0 is used, and 4 is used for the X.3 profiles. As a result, the mapping is only for the second octet of the P-selector.

Table E-3 lists the AFI values and their associated IDI formats, as described in Addendum 2 of the International Standard 8348. The AFI value identifies the abstract syntax (decimal or binary format) for the DSP portion of the NSAP address. The IDI formats identify the addressing authority responsible for assigning values of the DSP.

Table E-3 AFI Values

	AFI Values	
IDI Format	**Decimal (Max. DSP Length*)**	**Binary (Max. DSP Length*)**
X.121	36 (24)	37 (9) 12†
ISO DCC	38 (35)	39 (14) 17†
F.69	40 (30)	41 (12) 15†
E.163	42 (26)	43 (10) 13†
E.164	44 (23)	45 (9) 12†
ISO ICD	46 (34)	47 (13) 17†
Local	48 (38)	49 (15) 19†

* Decimal digits for decimal; binary octets for binary.

† Maximum length of binary DSP reflect change in Standard 8348 pDAM3.

Each of the IDI format values is described here:

X.121 The IDI format adheres to CCITT Recommendation X.121. The maximum IDP length is 16 digits.

ISO DCC The IDI format adheres to values allocated by an ISO DCC IDI Format Registration Authority. The IDP length is 5 digits.

F.69 The IDI format adheres to CCITT Recommendation F.69. The maximum IDP length is 10 digits.

E.163 The IDI format adheres to CCITT Recommendation E.163. The maximum IDP length is 14 digits.

E.164 The IDI format adheres to CCITT Recommendation E.164. The maximum IDP length is 17 digits.

ISO ICD The IDI format adheres to International Code Designator (ICD) values allocated by ISO 6523. The IDP length is 6 digits.

Local The IDI is a null value. The local network administrator is responsible for allocating the values of the DSP. The IDP length is 2 digits.

F

SUPPORTED MIBS

This appendix lists all management information base (MIB) modules supported by the NETBuilder family of products and the software packages that run on NETBuilder systems. To determine which MIB modules are supported by a particular software package, obtain the list of object IDs assigned to 3Com products by anonymous ftp from ftp.3com.com. The file 3com-products.mib in the directory pub/3com-mibs/all-mibs/ contains this list.

Supported Operations

The Get and GetNext operations are supported for all simple objects and tables. The Set operation is supported with limitations on all objects that provide write access. Set operations take effect immediately, and changes are saved to the disk so new configurations are not lost after reboot.

The tables within the 3Com-defined MIBs support a subset of the functionality provided by the rowStatus textual convention. Row creation is allowed using only the *createAndGo* method. With the createAndGo method, the *Status* object of the table is set to creatAndGo(4) within the same protocol data unit (PDU) that carries the other columnar values; the result is that the new row is immediately marked as active(1). Once active, the row cannot be modified (there are a few exceptions). Changes to rows can be made only by first deleting the row, and then recreating it with the proper values.

Port Numbering Convention in SNMP

Throughout this appendix, references to port-numbering assume the format used in the NETBuilder II user interface. The current implementation of Simple Network Management Protocol (SNMP) uses a port-numbering scheme that differs from the NETBuilder II user interface. Table F-1 shows the relationship between these two schemes.

Table F-1 Port Numbering in SNMP

UI Port Label	8-Slot NB II SNMP Port Label	UI Port Label	8-Slot NB II SNMP Port Label
1/1A	1	5/5A	5
1B	9	5B	13
1C	17	5C	21
1D	25	5D	29
1E	33	5E	37
1F	41	5F	45
2/2A	2	6/6A	6
2B	10	6B	14
2C	18	6C	22

(continued)

Table F-1 Port Numbering in SNMP (continued)

UI Port Label	8-Slot NB II SNMP Port Label	UI Port Label	8-Slot NB II SNMP Port Label
2D	26	6D	30
2E	34	6E	38
2F	42	6F	46
3/3A	3	7/7A	7
3B	11	7B	15
3C	19	7C	23
3D	27	7D	31
3E	35	7E	35
3F	43	7F	47
4/4A	4	8/8A	8
4B	12	8B	16
4C	20	8C	24
4D	28	8D	32
4E	36	8E	40
4F	44	8F	48

MIBs Supported by the Bridge/Router

The bridge/router supports thirteen SNMP MIB modules defined by the Internetworking Engineering Task Force (IETF), ten MIB modules defined by 3Com, one defined by IBM, and one defined by Novell.

The IETF MIB modules are:

- RFC 1243 (AppleTalk MIB)
- RFC 1286 (Bridge MIB)
- RFC 1284 (Ethernet-like MIB)
- RFC 1285 (FDDI MIB)
- RFC 1315 (Frame Relay DTE MIB)

  Except for frCircuitCommittedBurst and frCircuitExcessBurst
- RFC 1354 (IP Forwarding MIB)
- RFC 1213 (MIB II)
- RFC 1253 (OSPF MIB)
- RFC 1271 (RMON Alarm and Event MIB)
- RFC 1593 (APPN MIB)
- RFC 1749 (Source Route MIB)
- RFC 1231 (Token Ring MIB)
- RFC 1304 (SMDS Interface Protocol (SIP) MIB)

The Token RIng MIB was moved from underneath the experimental branch to the transmission branch per RFC 1239. The 3Com implementation of the Token Ring MIB supports the dot5Table and the dot5StatsTable, but it does not support the optional dot5Timer Table.

The 3Com private MIB modules control the following bridge/router services:

- AuditLog
- Bridge extension
- DLSw
- DVMRP
- IP
- IP RIP
- IP security options
- IPX
- IPX policies
- LLC
- Multicast IP
- Multiple logical networks
- Mnemonic filtering
- Port and path
- SDLC
- System

*The IBM-defined MIB controls the DLSw Service. The Novell-defined MIB controls the NLSP Service**.***

3Com Private MIBs

The 16 3Com private MIBs are located under the following headings on the MIB disk:

AuditLog MIB	A3Com-AUDL-MIB
Bridge Extension MIB	A3Com-Bridge-MIB
LLC MIB	A3Com-LLC-MIB
DLSw MIB	A3Com-DLSw-MIB
MLN MIB	A3Com-MLN-MIB
Filtering MIB	A3Com-Filter-MIB
IP Extension MIB	A3Com-IPextns-MIB
IP Security Options MIB	A3Com-IPSO-MIB
IPX MIB	A3Com-IPX-MIB
IPX Policies MIB	A3Com-IPXpolicy-MIB
Port and Path MIB	A3Com-PortPath-MIB
RIP IP MIB	A3Com-RIP-IPextns-MIB
SDLC MIB	A3Com-SDLC-MIB
System MIB	A3Com-System-MIB
Multicast IP MIB	A3Com-Mip-MIB
DVMRP MIB	A3Com-Dvmrp-MIB

To get a listing of the levels of MIB support offered by each bridge/router product, ftp to ftp.3Com.com, enter the log-on command as anonymous, enter the cd command to change the directory to pub/docs/3Com-mibs, and enter the get command to obtain the README file.

G

MACRO FEATURES

This appendix provides information about macro conventions and macros with conditional statements.

Macro Conventions

Macro contents must begin with a left parenthesis. If the definition requires more than one line, press the Return key after the opening left parenthesis. The macro: prompt then appears as a locator for you. All characters entered between the opening and closing parentheses are part of the macro. Nested parentheses in balanced pairs are allowed. When you end the macro with the closing right parenthesis, the normal server prompt returns.

A single macro cannot contain more than 256 characters. Macro names must follow the DOS file naming conventions: the macro name cannot contain more than 14 characters and the macro name extension, if any, cannot contain more than three characters.

If an error is detected in the macro, the macro stops executing, and an error message appears. The error message includes the macro name and a short explanation. After the message appears, you are returned to Command Mode.

A macro can include the DO command to call another macro. Embedded calls to other macros is called *nesting*. Because of the large amount of memory required to keep track of the calling history and variables, the limit for nested macros is 10.

If you use command substitution in macros, the user interface may hang because of the echoing of flow-control characters between the quotation marks in the string sent from the bridge/router.

Macros With Conditional Statements

Macros with conditional statements contain variables (such as arguments and return codes) and control structures (such as “if-else-end” and “switch-case-end”). Control structures instruct the macro to test conditions or make comparisons from which execution decisions can be made. Variables contain the values from which these comparisons are made. The execution decisions affect the final macro output.

Macro Variables

Macro variables store values in memory, which can be evaluated by a macro during execution. Values can be either numeric or strings, and are represented by a variable name such as $1 or $rc. Examples of variable types are arguments, input or output requests, return codes, or global variables. A variable can be readable, writable, or both readable and writable by the macro.

Table G-1 lists all possible variables within macros. The table breaks down the variables by type (argument, input, output, return code, asynchronous event, global, system/user), value (numeric or string), and actions (readable or writable). Each variable name begins with a dollar sign ($).

Table G-1 Variables within Macros

		Value		
Type	**Name**	**Numeric (N)/ String (S)**	**Readable**	**Writable**
Argument	#	N	*	*
	1	S	*	*
	2	S	*	*
	3	S	*	*
	4	S	*	*
	5	S	*	*
Input	<	S	*	
Output	>	S		*
Return Code	rc	N	*	
Asynchronous Event	error	S	*	*
	brk	S	*	*
Global	global	N	*	*
	lpw	S	*	
	sess	N	*	
	prompt	S	*	
	portid	N	*	
	priv	N	*	
	user	N	*	
	nm	N	*	
	lnm	N	*	
	gnm	N	*	
	eth_add	S	*	

Variable Types

There are nine types of variables:

- Argument (local)
- Input
- Output
- Return code
- Asynchronous
- Global
- System/user information
- Numeric and String
- Readable and Writable

These variables are described in the following sections.

Argument (local). Argument (local) variables ($#, $1, $2, $3, $4, and $5) provide the option of passing up to five arguments to a macro. Within the macro, these arguments can be referenced by $1 through $5. The $# variable contains the actual number of arguments passed. Local argument variables apply only to a particular macro. For example:

```
                  $1        $2       $3        $4         $5 (empty)
do <macro-name>   call      me       at        370-6610
```

The $1 variable will contain call, $2 will contain me, $3 will contain at, $4 will contain 370–6610, and $5 is empty. The $# variable will be 4.

Similar to a C language procedure, argument variable values exist only within the macro, and these values disappear when the macro terminates. The same macro can be executed with different argument variables assigned, giving it a completely independent value. For nested macros, the called macro has its own set of argument variables, independent of the calling macro.

Input. Input variables ($<) cause the macro to stop executing and wait for your input. $< is then substituted by your input, and the macro continues executing based on your input. Input variables are illegal in macros that are submitted to the SCHeduling Service.

Output. Output variables ($>) cause any string of characters assigned to this variable to be displayed on your terminal. Output variables can generate all 127 characters on the terminal screen, as does the Echo command. For example, both of the following lines generate a bell ([Ctrl]+G) to the terminal when they are executed:

```
echo “^G”
$> = ^G
```

However, $> does not operate exactly the same as the Echo command. The Echo command automatically appends a CR-LF after the string being echoed, and the $> variable does not. Therefore, $> is more convenient to use for controlling screen layout.

Return Code. Return code variables ($rc) contain the return status of the last executed user interface command. $rc is always 0 (no errors) if the last command executed successfully. When a called macro returns to the calling macro, the $rc variable is not affected by the return operation.

Asynchronous Event. Asynchronous event variables ($error and $brk) handle unexpected conditions that cause a macro to abort. A macro will abort under one of two conditions: when an internal error is detected or when the user presses the Break key.

The $error variable is used to recover from an error. With $error defined, if an error occurs, the macro will stop executing, and a new macro, as specified by the $error variable, will automatically begin executing to clean up or recover from the error. Without $error defined, an error detected in a macro stops the execution of the macro and returns to command mode.

The $break variable defines a macro that will begin executing when the user presses the Break key. For example, you can define a macro called “recover” that will put the user into Listen mode, then assign the macro “recover” to $break so that recover is executed when the Break key is pressed.

Without the $break variable defined, you can exit a macro while it is executing by pressing the Break key (unless the NoMacroBreak option is set in the InterAction parameter). The Break key exits the macro and returns to Command mode.

The $error and $brk variables do not cancel the effects of errors or breaks. They restart a new macro service in order to handle the error or break signal. If you do not define $error or $brk, errors and break signals will force the macro to stop executing, and will return to command mode.

In addition to a macro name, $error and $brk can contain up to five arguments. For example:

```
$error = <macro name for handling error> arg1 arg2 ...
```

For descriptions of these variables, refer to “Argument (local)” on page G-3.

Global. Global variables ($global) provide another way to pass information between macros when calling a macro. A global variable is a variable that is globally shared among all macros executed from the same user port.

You can use $global to test the return status of a macro.

System/user Information. System/user information variables include the following:

$eth_add	Server Ethernet address - media access control (MAC) address of the first interface
$lpw	Local password
$portid	User port number executing the macro
$prompt	Prompt strings, depending on user privilege
$sess	Number of user sessions outstanding (on this port)
$priv	Privilege of the user executing the macro
$user	User privilege value is 0 (user)
$lnm	User privilege value is 1
$gnm	User privilege value is 1
$nm	Network manager, 1

$priv contains the current privilege level of the user. Its value will be equal to $user or $nm depending on the privilege level of the user. $priv tests the privilege level of the user within a macro. $lpw contains the passwords for Network Manager privilege levels. These variables are used to compare passwords within a macro.

Numeric Variables and String. Numeric variables store decimal values between 32767 and -32768. String variables can store any numeric value (within the described limit) or any character sequence. String variables can perform all the functions of numeric variables. They can be compared with other strings or numeric values, incremented or decremented, or assigned to another numeric variable. Numeric values incremented beyond 32767 become negative.

Readable and Writable. All variables, except the output variable $>, are readable by the macro, which means that the values they store can be interpreted and compared by the macro in any expression. The $> variable generates output to the user's screen.

Some variables are writable by the macro, which means they can be reassigned new values within the macro. As shown in Table G-1, only the following variables are writable:

$#	$5
$1	$>
$2	$error
$3	$brk
$4	$global

Assigning values to non-writable variables causes a syntax error and aborts the macro. For example, if you want to change the global password, $lpw = <password> will not work because the variable $lpw is not writable.

Comparing and Reassigning Variables

Six comparison operators are available for testing macro variables against each other or constant values. Comparison operators are used most often in the if-else-end control structure to compare values. Table G-2 lists the available comparison operators.

Table G-2 Comparison Operators

Operator	Comparison Performed
==	Values are equal.
!=	Values are not equal.
>=	Value on left is greater than or equal to value on right.
<=	Value on left is less than or equal to value on right.
<	Value on left is less than value on right.
>	Value on left is greater than value on right.

Numeric and string variables can be compared with each other. These rules apply:

- When both variables are numeric, which can be string variables containing numeric values, the comparison is based on value. For example:

  ```
  123 == 00123
  ```

- If any one of the variables is a string value (containing a character other than 0–9, excluding space and tab), a string comparison is performed. The difference between uppercase and lowercase is ignored. Only the first character is compared. If the first characters are equal, the next character is compared until a decision is made. For example:

  ABC is equal to AbC

  ABC is not equal to 123

- Variables can be reassigned with statements such as the following:

  ```
  <variable> = <value>
  ```

- Only numeric values or string variables containing numeric values can be assigned to numeric variables. Otherwise, a syntax error is detected and the macro execution is aborted.
- Numeric variables can be incremented or decremented with plus and minus statements. The plus statement is used most often within a loop structure to increment a counter, which can then be tested against a value.

  ```
  variable ++
  variable --
  ```

Variable Substitutions

Immediately before a line is executed, the line is scanned and all variables are replaced with their values. Substitution can be done only once. Variables can appear anywhere in the line and still be substituted. For example:

```
Echo “My arguments are $# $1 $2 $3 $4 and $5”
REMote 192.9.200.$1
```

Two dollar signs ($) allow you to escape variable substitutions. For example, if you enter:

```
Echo “argument $$1 is $1”
```

the following display appears:

```
argument $1 is <substituted value>
```

Control Structures

Control structures are the tools that can alter the sequences of execution. The syntax is similar to a C program. Control structures must begin and end within the boundary of the macro. For example, in the if-else-end structure, all three parts of the conditional statement (if, else, and end) must be contained within the macro. If any part of the structure is missing, a syntax error is detected and the macro aborts.

Control structures are free to nest within one another. For example, within one loop structure you can have several if-end structures. There is no limit to the number of nested control structures allowed.

If-Else-End

The if-else-end structure is used to make two-way decisions. The syntax is as follows:

```
if <expression>
commands ...
else
commands ...
end
```

The else part is optional. The <expression> is evaluated; if it is TRUE, the macro executes the immediately following commands. If it is FALSE and there is an else statement, then the commands following the else statement are executed. If it is FALSE and there is no else statement, then the commands following the end statement are executed. There can be any number of commands between if-else-end, including none.

The syntax for <expression> is:

```
<variable> <op> <value>
```

<op> can be one of the six comparison operations ==, !=, >=, >, <=, and <. A single variable must be on the left side of the comparison operator. <value> can be any string of characters and digits, with variables intermixed, or it can be empty.

Both <variable> and <value> can contain numeric values or strings. Both can contain more than one word, but only the first word is compared.

Switch-Case-End

The switch structure is a multiway decision maker. It is usually used with $< to make a comparison based on the user's input. The syntax is as follows:

```
switch <value>
  case <value>
  commands ...
  case <value>
  commands ...
  case *
  commands ...
end
```

The switch structure tests whether the <value> immediately following the switch matches one of the <values> after case. If the values match, the macro executes the immediately following commands.

There can be any number of commands after each case, including none.

In situations where there is no match, but there is a case* (wildcard character) before the end, the command following case* is executed. The * can appear after any case between the switch and end. The case* is not required in the switch-case-end structure.

If the value immediately following the switch does not match any case within the switch, the macro will continue to execute the commands after the end.

Loop-End

Any of the following commands can appear within the loop structure:

```
loop
  commands ...
end
```

The loop structure comprises a set of instructions that can be executed repeatedly while certain conditions prevail. You define the conditions within the loop structure using the comparison operators. The loop can be terminated by using the "break," "return," and "exit" keywords inside the loop-end structure.

Keywords

If a keyword is not the first word in a line, it is not recognized. The following keywords can be used:

audit	exit
break	if
case	loop
continue	return
else	switch
end	

Audit

This keyword generates an audit trail record of the macro information (MI) type. You can provide a string of data following the keyword.

Break

This keyword terminates the current loop structure, and the macro execution continues after the end keyword. The current loop structure is the structure that contains the break keyword. The break function does not apply to the switch-end structure. Break is only meaningful within a loop-end structure.

Continue

This keyword directs macro execution to the beginning of the enclosing loop-end structure. It is only meaningful within a loop-end structure.

Exit

This keyword stops execution of all macros. It frees all associated buffers and returns the user to Command mode.

Return

This keyword stops execution of the current macro. It resumes the previous calling macro, if any.

Macro Caching and Shared Macros

As macros become larger, more complex, and more heavily nested, even a relatively simple macro can require a large number of nested macros. The dependence on macro file service from the floppy diskette on the local server becomes more and more critical. A sudden failure of the local disk drive can create a serious service interruption. The server has the following features to handle such failures:

- The server keeps track of all the macros currently being executed in a macro cache.

  When a macro is not being executed, it is kept in the cache memory as long as there is space available. The next time you request a macro file, the cache is searched first. If the macro is found, it is automatically executed. This reduces the dependence on the server and speeds up macro execution.

- The server links several users to a single copy of a macro instead of distributing many copies of it, thereby sharing the macro.

  Sharing macros relieves the memory overhead associated with keeping numerous similar macros. There is no limit to the number of users that can be linked to a macro.

As long as a macro is linked to a user, that macro stays cached. If there are no users linked to the macro, the server keeps the macro in the cache based on the amount of space available in the cache.

For example, if the macro cache is between 80% and 100% full, a macro that is not linked to any users will be stored for up to 10 minutes.

Table G-3 lists the macro cache aging algorithm.

Table G-3 Macro Cache Aging

Cache Usage Level	Aging
Below 50%	No aging
Below 80% but higher than 50%	Cached up to 8 hours
Below 100% but higher than 80%	Cached up to 10 minutes

If the cache overflows, the server rebuilds its cache memory, which frees all macros that are not linked.

Macro caching can cause a discrepancy between the DO <macro name> and SHow MACros <macro name> commands. The DO command searches for the file first in the cache and then in the local diskette or macro server. The SHow command reads the macro file directly from the local diskette or macro file server and never checks the cache. If the file stored in the cache is not the same as the one on the diskette or file server, you get different results.

If the network manager modifies the macro files, the cache aging algorithm may not pick up new macros until after the aging period. The FLush MACros command is available to force the server to flush its cache.

Larger Macros

The size limit for each macro that is stored is 256 bytes. In many applications, this is barely enough for a moderately sophisticated macro. To solve this problem, the network manager can create macros that are the necessary size using the plus sign (+).

To store a macro larger than 256 bytes, the macro must be split into smaller macros such as m1, m2. The name of the large macro will be m1+m2 when it is cached in the memory. The macro cache stores up to eight bytes of the macro name. When the cache is searched for a macro, only the first 14 bytes of the macro name are considered significant. If a macro name exceeds the limit, it is truncated. For example, the following two macro names are considered the same in the cache:

macro1+macro2+macro3
macro1+macro2+XX

Spaces are not allowed around the plus sign (+). To execute the macro, use the DO command m1+m2, which informs the server which two macros must be read and concatenated into one large macro. The command DO m1<space>+m2<space> means execute +m2 as its argument.

You can break up control structures across two or more macros that will be concatenated with the plus sign (+), because concatenated macros are considered one macro. For example, macro1 could contain the if part of the if-else-end control structure, and macro2 could contain the else and end parts of the structure.

There is no set limit on the number of plus signs (+) that you can use; however, the + operation requires a great deal of memory from the large server buffers and should not be overused.

Macro Nesting

A macro can call other macros (including itself), similar to a subroutine call in any computer language. These calls to other macros create macro nesting.

The limit to the number of nested macros is 10. Exceeding this limit causes an error and aborts the macro service. The variable $error can be set up to automatically capture the error and start a new macro service.

Example

The following example shows macros using features such as variables and conditional statements. This example executes the TraceRoute command on the address specified in the first argument to the macro and loops the number of times specified in the second argument to the macro.

```
NETBuilder [1]# define tracen = (
if $# !=2
echo “USAGE: tracen <destination -IP-addr> <times-to-loop>”
exit
else
if $2 ==0
echo “TRACEN: iterations must be > 0”
exit
end
loop
echo “ “
$>=#$2-
traceroute $1
$2--
if $2 <= 0
exit
end
end
end)
```

H

STATISTICS DISPLAYS

This appendix provides displays of accumulated system statistics for a particular service. To display statistics, use the SHow -SYS STATistics -<service> <option> syntax.

The statistics displayed are based on the time period in which they have been gathered. For example, during the busiest minute, specified by the SETDefault -SYS SampleOption command, and on the time interval in which you want to see the statistics, specified by the SETDefault -SYS SampleTime command.

A statistical display showing a blank in any field indicates that the service was not configured for the specified port.

The FLush -SYS STATistics -service command may take several seconds before statistics sampling is restarted. For more information on the FLush command, refer to Chapter 1 in *Reference for NETBuilder Family Software*.

For more information on syntax, refer to “STATistics” on page 58-14 in *Reference for NETBuilder Family Software*. For more information on displaying FR and X25 statistics, refer to Chapter 25 and Chapter 65 in *Reference for NETBuilder Family Software*.

This appendix provides the statistics displays in alphabetical order.

The displays in this appendix are examples only. Actual displays will vary according to your system configuration.

AppleTalk Service

The following is an example of a display generated by the SHow -SYS STATistics -AppleTalk command:

```
ACCUMULATED VALUES
== AppleTalk statistics ====== 1======= 3======  5======  7=======

DDP Statistics :
   General Datagram Counts :
   Locally Originated          16       65       -        -
   Short DDP Out               0        0        -        -
   Long DDP Out                16       65       -        -
   Total In                    25       104      -        -
   In - Not Local Dest         0        0        -        -
   In - Locally Destined       25       104      -        -

Dropped Datagram Counts :
   No Recipient                25       55       -        -
   No Route                    0        0        -        -
   Data Too Short              0        0        -        -
   Data Too Long               0        0        -        -
   Broadcast Error             0        0         -        -
   Short Header Error          0        0         -        -
   Hop Count Error             0        0         -        -
   Checksum Error              0        0         -        -

RTMP Statistics :
   Network Filter Matches      0        0         -        -
   Data/Responses In           0        0         -        -
   Data/Responses Out          5        5         -        -
   Requests In                 0        0         -        -
   Requests Out                10       10        -        -
   Route changes (= dist)      0        0         -        -
   Route changes (shorter)     0        0         -        -
   Network Dist. Exceeded      0        0         -        -
   Network Route Deletes       0        0         -        -
   Invalid Packets             0        0         -        -
   Bad Tuple Packets           0        0         -        -
   Net Number Overlaps         0        0         -        -

ZIP Statistics :
   Queries In                  0        0         -        -
   Queries Out                 0        0         -        -
   Replies In                  0        0         -        -
   Replies Out                 0        0         -        -
   Extended Replies In         0        0         -        -
   Extended Replies Out        0        0         -        -
   GetZoneList Req. In         0        2         -        -
   GetZoneList Rep. Out        0        2         -        -
   GetLocalZones Req. In       0        0         -        -
   GetLocalZones Rep. Out      0        0         -        -
   GetMyZone Req. In           0        0         -        -
   GetMyZone Rep. Out          0        0         -        -
   GetNetInfo Req. In          0        2         -        -
   GetNetInfo Rep. Out         0        2         -        -
   GetNetInfo Req. Out         0        0         -        -
   GetNetInfo Rep. In          0        0         -        -
   Invalid Packets             0        0         -        -
   Zone Name Conflicts         0        0         -        -
   Zone Count Conflicts        0        0         -        -

AEP Statistics :
   Echo Requests In            0        0         -        -
   Echo Replies Out            0        0         -        -
   Echo Requests Out           0        0         -        -
   Echo Replies In             0        0         -        -

NBP Statistics :
   Entity Filter Matches       -        -         -        -
```

The elements of this display are described as follows:

DDP Statistics

General Datagram Counts

Locally Originated — Number of packets transmitted out a port that originated within the router (for example, Routing Table Maintenance Protocol (RTMP) route information packets).

Short DDP Out — Number of short Datagram Delivery Protocol (DDP) packets transmitted out a port (always 0, because AppleTalk Phase 2 does not use short DDP headers; present because management information base (MIB) uses the same data structures).

Long DDP Out
Number of packets transmitted out a port with Long DDP Headers (all non-AppleTalk Address Resolution Protocol (AARP) packets).

Total In
Number of packets received on a port by DDP from external devices.

In - Not Local Dest
Number of packets received from external devices out port not addressed specifically to this router and that are not broadcast/multicast packets of interest to RTMP, Zone Information Protocol (ZIP) or Name Binding Protocol (NBP) protocols.

In - Locally Destined
Number of packets received from external devices out port addressed specifically to this box or that are broadcast/multicast packets of interest to the AppleTalk protocols implemented RTMP, ZIP, NBP, AppleTalk Echo Protocol (AEP)

Dropped Datagram Counts

No Recipient
Number of packets received on port destined for AppleTalk node on router for protocols not present or ready to accept packets.

No Route
Number of packets received on port not destined for this router for which no route was found in the AppleTalk routing table.

Data Too Short
The total number of input DDP datagrams dropped because the received data length was less than the data length specified in the DDP header or the received data length was less than the length of the expected DDP header.

Data Too Long
The total number of input DDP datagrams dropped because the received data length was greater than the data length specified in the DDP header or because they exceeded the maximum DDP datagram size.

Broadcast Error
The total number of input DDP datagrams dropped because this entity was not their final destination and they were addressed to the link level broadcast.

Short Header Error
The total number of input DDP datagrams dropped because this error was not their final destination and their type was short DDP (always 0 since AppleTalk Phase 2)

Hop Count Error
The total number of input DDP datagrams dropped because this entity was not their final destination and their hop count would exceed 15.

Checksum Error
The total number of input DDP datagrams dropped because of a checksum error.

RTMP Statistics

Network Filter
Number of packets not routed through port because of a Network Number

Matches
Filter in effect.

Data/Responses In
Number of RTMP Data packets or Route Data Request responses received over port.

Data/Responses Out
Number of RTMP Data packets broadcast or Route Data Request responses sent out port.

Requests In
Number of RTMP Request packets received over port.

Requests Out
Number of RTMP Request packets sent out port.

Route changes (dist)
Number of times RTMP changes the Next Internet Router in a routing entry because the hop count advertised in a routing tuple was equal to the current hop count for a particular network.

Route changes (shorter)
Number of times RTMP changes the Next Internet Router in a routing entry because the hop count advertised in a routing tuple was less than the current hop count for a particular network.

Network Dist. Exceeded
Number of times RTMP deletes a route from the table because of a distance change that makes the network inaccessible because of the 15-hop limitation.

Network Route Deletes
Number of times RTMP deletes a route because it was aged out of the table. This can help to detect routing problems.

Invalid Packets
Number of packets ignored by RTMP because of invalid data found within the RTMP data, such as invalid network numbers. SHow DIAGnostics should be checked if nonzero values are seen.

Bad Tuple Packets
Number of routing information packets received over the port containing Invalid Data Tuples. SHow DIAGnostics should be checked if nonzero values are seen.

Net Number Overlaps
Number of network overlaps detected between routing information obtained from another router over the given port and current network entries in the routing table. SHow DIAGnostics should be checked if nonzero values are seen.

ZIP Statistics

Queries In
Number of ZIP query packets received over the port.

Queries Out
Number of ZIP query packets sent to other routers over the port.

Replies In
Number of ZIP query response packets received over the port from other routers.

Replies Out
Number of extended ZIP query response packets transmitted out the port.

Extended Replies In
Number of ZIP Extended Replys received by this entity.

Extended Replies Out
Number of ZIP Extended Replys sent by this entity.

GetZoneList Req. In
Number of ZIP GetZoneList transactions received by this entity.

GetZoneList Rep. Out
Number of ZIP GetZoneList transactions sent by this entity.

GetLocalZones Req. In
Number of ZIP GetLocalZones transactions received by this entity.

GetLocalZones Rep. Out Number of ZIP GetLocalZonesReply transactions sent by this entity.

GetMyZone Req. In Number of ZIP GetMyZone transactions received by this entity.

GetMyZone Rep. Out Number of ZIP GetMyZoneReply transactions sent by this entity.

GetNetInfo Req. In Number of ZIP GetNetInfo packets received by this entity.

GetNetInfo Rep. Out Number of ZIP GetNetInfoReply packets sent by this entity.

GetNetInfo Req. Out Number of ZIP GetNetInfo packets sent by this entity.

GetNetInfo Rep. In Number of ZIP GetNetInfoReply packets received by this entity.

Invalid Packets Number of ZIP packets of all types received with invalid information detected. SHow DIAGnostics should be checked if nonzero values are seen.

Zone Name Conflicts SHow DIAGnostics should be checked if nonzero values are seen.

Zone Count Conflicts SHow DIAGnostics should be checked if nonzero values are seen.

AEP Statistics

Echo Requests In Number of AppleTalk Echo requests received.

Echo Replies Out Number of AppleTalk Echo replies sent.

Echo Requests Out Number of AppleTalk Echo requests sent.

Echo Replies In Number of AppleTalk Echo replies received.

NBP Statistics

Entity Filter Matches Number of times a Name Binding protocol lookup or reply packet is ignored because it matches an entity filter.

ARP Service

The following is an example of the display generated by the SHow -SYS STATistics -ARP command:

```
ACCUMULATED VALUES
========================= ARP statistics ========================
Data Pkts Discarded
   Aged                     0
   Queue Full               0
   Addr List Full           0
Data Pkts In Queue          0

== ARP statistics =========== ==== 1====   ==== 2====   ==== 4====
Requests Received:
   All Requests Rcvd        627273       -            -
   All Requests Rspd        54           -            -
   Proxy Requests Rcvd      625489       -            -
   Proxy Requests Rspd      0            -            -
   Agent Requests Rcvd      0            -            -
   Agent Requests Rspd      0            -            -
```

```
Requests Sent:
   All Req Sent                17           -            -
   Repeat Req Sent             0            -            -
   Refresh Req Sent            12           -            -

InARP Statistics:
   InARP Request(out)          0            -            -
   InARP Response(in)          0            -            -
   InARP Request(in)           0            -            -
   InARP Response(out)         0            -            -
   Pkts Discarded              0            -            -

RARP Statistics:
   Outgoing Requests           0            -            -
   Incoming Responses          0            -            -
   Incoming Requests           15           -            -
   Outgoing Responses          0            -            -
```

The elements of this display are described as follows:

Data Pkts Discarded

Aged Number of packets discarded on a port as a result of time-outs waiting for a response.

Queue Full Number of packets discarded on a port as a result of a full routing queue. The maximum allowable number of packets waiting for Internet address resolution is 10.

Addr List Full Number of packets discarded on a port as a result of a full address list. The maximum allowable number of addresses waiting for resolution is 10.

Data Pkts In Queue Number of packets on a port still in the queue waiting for address resolution.

Requests Received

All Requests Rcvd Number of Internet address resolution requests received on a port.

All Requests Rspd Number of responses to Internet address resolution requests on a port.

Proxy Requests Rcvd Number of proxy requests received on a port for an Internet address not on the network where the request originated.

Proxy Requests Rspd Number of responses to proxy requests responded to on a port.

Agent Requests Rcvd Number of agent requests received on a port. An agent is a designated router that can respond to Address Resolution Protocol (ARP) requests for a PC; for example, when the ARP Service is not implemented.

Agent Requests Rspd Number of responses to agent requests on a port.

Requests Sent

All Requests Sent Number of Internet address resolution requests sent on a port since the last boot or since dynamic information in routing tables was last flushed.

Repeat Req. Sent Number of Internet address resolution repeat requests sent on a port since the last boot or since dynamic information in routing tables was last flushed.

Refresh Req. Sent Number of Internet address resolution refresh requests sent on a port since the last boot or since dynamic information in routing tables was last flushed.

InARP Statistics

InARP Request (out) Number of InARP requests sent on a port.

InARP Response (in) Number of InARP responses received on a port.

InARP Request (in) Number of InARP requests received on a port.

InARP Response (out) Number of InARP responses sent on a port.

Pkts Discarded Number of packets discarded on a port.

RARP Statistics

Outgoing Requests Number of outgoing Reverse Address Resolution Protocol (RARP) requests transmitted on a port by a RARP client.

Incoming Responses Number of incoming RARP responses received on a port by a RARP client.

Incoming Requests Number of incoming RARP requests received on a port by the RARP server.

Outgoing Responses Number of outgoing RARP responses transmitted on a port by the RARP server.

ATUN Service

The following is an example of the display generated by the SHow -SYS STATistics -ATUN command:

```
ACCUMULATED VALUES
== ATUN statistics =========== 1======= 2======  3======  4=======
RCVD: Address Pkts             -        -        0        9
      Bytes                    -        -        0        18
      Broadcast Pkts           -        -        0        0
      Bytes                    -        -        0        0
      No CU Pkts               -        -        0        4
      Bytes                    -        -        0        8
Err:  Too Short                -        -        0        0
      Parity                   -        -        0        0
      Break                    -        -        0        0
      Framing                  -        -        0        0
      CD Lost                  -        -        0        0
      Internal                 -        -        0        0
Xmit: Pkts sent                -        -        0        9
      Bytes Sent               -        -        0        54
      Err: FlowControl         -        -        0        0
```

The elements of this display are described as follows:

Addressed The packet and byte count for data received from the port and addressed to a specific CU.

Broadcast The packet and byte count for data received from the port and addressed to all CUs, either via explicit broadcast or because addressing on the port is disabled.

No CU The packet and byte count for frames discarded when addressing is used, and no CU configuration can be found matching the address in the frame.

Err Provides a breakdown count of various error frames for packet counts only. Error frames are discarded.

Too Short Addressing configured on the port, but the framer received was too short to contain an address byte at the configured AddrLOCation value.

Parity This frame was terminated due to receipt of a parity error on the asynch line.

Break This frame was terminated due to receipt of a break error on the asynch line.

Framing This frame was terminated due to receipt of a framing error on the asynch line.

CD Lost This frame was terminated because the DCD control signal dropped during receipt of a character.

Internal This frame was terminated because some internal error (such as a buffer overflow) occurred.

Xmit Shows the packet and byte count of data transmitted to the port by the CU tunnels.

FlowControl Shows the count of packets discarded on transmission due to transmit overflow on the port.

BGP Service

The following is an example of the display generated by the SHow -SYS STATistics -BGP command:

```
========================================BGP Statistics=================================
BGP Received:     Messages     Updates      Keepalives  Notification   Bytes     Routes
                  0            0            0           0              0         0
                  Unreachables Duplicates
                  0            0
BGP Transmitted:  Messages     Updates      Keepalives  Notification   Bytes     Routes
                  0            0            0           0              0         0
                  Unreachables Duplicates
                  0            0
```

These are some of the statistics that will be maintained by the BGP Service.

BGP Statistics for All Peers

The following statistics are for the entire router:

Bytes in
Bytes out
Updates In
Updates Out
Notification Messages
Notifications Out
Keepalives in
Keepalives Out
Networks
AS paths
Unreachables in
Unreachables out
Network Policy discards
AS policy discards

Per-peer Statistics

InUpdates
OutUpdates
InMessages
OutMessages

BRidge Service

The following is an example of the display generated by the SHow -SYS STATistics -BRidge command:

```
ACCUMULATED VALUES
== BRIDGE statistics === ==== 1===  ==== 3===  === 5===   === 7===
Bridge Statistics
InFrames                  787        0          -          -
InDiscards                681        0          -          -
OutFrames                 0          106        -          -
OutDiscards               0          0          -          -
MtuDiscards               0          0          -          -
BCLDiscards               0          0          -          -
BCLInvoked                0          0          -          -
IPFragmented              0          0          -          -
IPFragments               0          0          -          -
```

The elements of this display are described as follows:

InFrames — Number of good incoming frames.
InDiscards — Number of incoming discarded frames.
OutFrames — Number of outgoing good frames.
OutDiscards — Number of outgoing discarded frames.
MtuDiscards — Number of packets discarded as a result of exceeding maximum packet size. Mixed media configurations only.
BCLDiscards — Number of packets discarded by the broadcast limit mechanism.
BCLInvoked — Number of timer intervals in which the broadcast limit threshold was exceeded.
IPFragmented — Total number of IP packets fragmented.
IPFragments — Total number of IP fragments generated.

While InFrames and InDiscards counters are accurate, OutFrames and OutDiscards counters report the number of packets the bridge tried to forward. If the packets are dropped for other reasons after the bridge tried to forward it, these packets will not show up in the OutDiscards counters, but in the port statistics. InDiscards are a normal condition (that is, the destination media access control (MAC) address is on the same network segment as the bridge from which it was received).

If the display indicates that there are very few InDiscards compared to OutFrames on a LAN port, the LAN segments connected by your bridge may not be evenly distributed. The bridge routinely forwards all or most of the frames received on that port. If the display indicates that there is an excessively large number of OutDiscards on a LAN port, the port may be highly saturated, the output port may not be forwarding, or the -BRidge BroadCastLimit parameter may not be appropriately set.

BSC Service

The following is an example of the display generated by the SHow -SYS STATistics -BSC command for receive statistics (the display for transmit statistics is similar except for the line heading “Rcvd General Poll”):

```
ACCUMULATED VALUES
== BSC statistics =========  === 1===  === 2===  === 3===  === 4===
Rcvd General Poll            0         0         0         0
   Specific Poll             0         0         0         0
   Selection                 0         0         0         0
   SOH Data Block            0         0         0         0
   Data Block - ETB          0         0         0         0
   Data Block - ETX          0         0         0         0
   Data Block - ITB          0         0         0         0
   Trans Data - ETB          0         0         0         0
   Trans Data - ETX          0         0         0         0
   Trans Data - ITB          0         0         0         0
   Bytes                     0         0         0         0
   ACK 0                     0         0         0         0
   ACK 1                     0         0         0         0
   ENQ                       0         0         0         0
   EOT                       0         0         0         0
   NAK                       0         0         0         0
   RVI
   TTD
   WACK
   Unknown
```

The elements of this display are described as follows:

Rcvd (Xmit) General Poll	General polls received (or transmitted).
Specific Poll	Specific polls received (or transmitted).
Selection	Selections received (or transmitted).
SOH Data Block	Data blocks received (or transmitted) with “Start of Header” in block.
Data Block - ETB	Data blocks received (or transmitted) with “End of Text Block” in block.
Data Block - ETX	Data blocks received (or transmitted) with “End of Text” in block.
Data Block - ITB	Data blocks received (or transmitted) with “Intermediate Transmission Block” in block.
Trans Data - ETB	Transparent data blocks received (or transmitted) with “End of Text Block” in block.
Trans Data - ETX	Transparent data blocks received (or transmitted) with “End of Text” in block.
Trans Data - ITB	Transparent data blocks received (or transmitted) with “Intermediate Transmission Block” in block.
Bytes	Total number of bytes received (or transmitted) that include only data frames such as “Trans Data - ETB” and excludes all control frames such as “ACK O.”
ACK 0	Positive acknowledgment for multipoint selection, point-to-point line bids and even number data blocks.
ACK 1	Positive acknowledgment for odd number data blocks.
ENQ	Enquiry.
EOT	End of Transmission.
NAK	Negative acknowledgment
RVI	Reverse interrupt.
TTD	Temporary Text Delay

WACK — Wait before Transmit Positive Acknowledgment "temporarily not ready to receive."

Unknown — Unknown blocks, block is discarded and not transmitted to DLSw. These blocks are not included in the transmit statistics as they have been discarded by the receiver.

CLNP Service

The following is an example of the display generated by the SHow -SYS STATistics -CLNP command:

```
ACCUMULATED VALUES
== CLNP statistics ========  === 1=== === 2=== === 3=== === 4===
Rcvd: good PDU               0        0        0        0
   pass to client            0        0        0        0
   bad PDU syntax            0        0        0        0
   dest unreachable          0        0        0        0
   cksum error               0        0        0        0
   lifetime expired          0        0        0        0
Xmit: good PDU               0        0        0        0
   xmit error                0        0        0        0
```

The elements of this display are described as follows:

CLNP statistics

Rcvd: good PDU

Number of protocol data units (PDUs) received.

pass to client — Number of PDUs passed to the local client of Connectionless Network Protocol (CLNP).

bad PDU syntax — Number of errors caused by PDUs with incorrect syntax.

dest unreachable — Number of errors generated because the router cannot forward the PDU.

cksum error — Number of checksum errors.

lifetime expired — Number of errors caused by expiration of the time-to-live (TTL) field in the PDU.

Xmit: good PDU

Number of PDUs transmitted successfully.

Xmit error — Number of errors caused by queue overflow.

DECnet Service

The following is an example of the display generated by the SHow -SYS STATistics -DECnet command, showing per-port statistics:

```
ACCUMULATED VALUES
=== DECnet statistics =========   1========= 2========= 3=========
Data Messages
   Received                       0          -          -
   MaxVisits Exceeded             0          -          -
   This Node                      0          -          -
   No Route                       0          -          -
   Bad version                    0          -          -
   Transmitted                    0          -          -
Routing Messages
   Level 1 Received               0          -          -
   Level 2 Received             0          -          -
   Bad Checksum                 0          -          -
   Level 1 Transmitted          0          -          -
   Level 2 Transmitted          0          -          -
Hello Messages
   Received                     0          -          -
   End Node                     0          -          -
   Discarded                    0          -          -
   Transmitted                  0          -          -
Phase V Data Messages
   Received                     0          -          -
   No Phase IV route            0          -          -
   Transmitted                  0          -          -
   No Phase V route             0          -          -
Internetwork Data Messages
   INR Transmitted              0          -          -
   ATG Transmitted              0          -          -
```

The elements of this display are described as follows:

Data Messages

Received	Number of data packets received.
MaxVisits Exceeded	Number of packets that exceed the maximum number of visits allowed by the MaxVisits parameter. These packets are assumed to be looping and are therefore discarded.
This Node	Number of packets whose destination node is the router itself and are therefore discarded.
No Route	Number of packets discarded because the router has no information (routes) available for routing them.
Bad version	Number of DECnet protocol packets encoded with a version number that is not supported by the 3Com implementation, for example, DECnet Phase III packets. Packets encoded in short or invalid format are also included.
Transmitted	Number of packets sent.

Routing Messages

Level 1 Received	Number of Level 1 routing messages received.
Level 2 Received	Number of Level 2 routing messages received.
Bad Checksum	Number of packets received with checksum errors.
Level 1 Transmitted	Number of Level 1 routing messages sent.
Level 2 Transmitted	Number of Level 2 routing messages sent.

Hello Messages

Received	Number of hello messages received.
End Node	Number of end node hello messages received.
Discarded	Number of hello messages discarded.
Transmitted	Number of hello messages sent.

Phase V Data Messages

Received	Number of Phase V data messages received and successfully transmitted as Phase I data messages.
No Phase IV Route	Number of Phase V data messages discarded because a Phase IV route to destination was not available.

Transmitted — Number of Phase IV data messages successfully transmitted as Phase V data messages.

No Phase V Route — Number of Phase IV data messages discarded because a Phase V route to destination was not available.

Internetwork Data Messages

INR Transmitted — Number of data packets sent to another directly attached network.

ATG Transmitted — Number of data packets sent using the user-defined address maps.

DLSw Service

The following is an example of the display generated by the SHow -SYS STATistics -DLSw command:

```
ACCUMULATED VALUES
======================== DLSw statistics =======================
CANUREACH:             Xmit                   9
                       Rcvd                   2
ICANREACH              Xmit                   0
                       Rcvd                   1
REACH_ACK              Xmit                   1
                       Rcvd                   0
DGRAMFRAME             Xmit                   0
                       Rcvd                   0
XIDFRAME               Xmit                   8
                       Rcvd                   7
CONTACT                Xmit                   0
                       Rcvd                   1
CONTACTED              Xmit                   1
                       Rcvd                   0
RESTART_DL             Xmit                   0
                       Rcvd                   0
DL_RESTARTED           Xmit                   0
                       Rcvd                   0
INFOFRAME              Xmit                   12
                       Rcvd                   12
ENTERBUSY              Xmit                   0
                       Rcvd                   0
EXITBUSY               Xmit                   0
                       Rcvd                   0
HALT_DL                Xmit                   0
                       Rcvd                   0
DL_HALTED              Xmit                   0
                       Rcvd                   0
NETBIOS_NQ             Xmit                   0
                       Rcvd                   0
NETBIOS_NR             Xmit                   0
                       Rcvd                   0
DATAFRAME              Xmit                   0
                       Rcvd                   0
NETBIOS-ANQ            Xmit                   0
                       Rcvd                   0
NETBIOS_ANR            Xmit                   0
                       Rcvd                   0
HALTDL_NO_ACK          Xmit                   0
```

```
                       Rcvd                 0
TEST_CIR_REQ           Xmit                 0
                       Rcvd                 0
TEST_CIR_RSP           Xmit                 0
                       Rcvd                 0
OTHERS                 Xmit                 0
                       Rcvd                 0
DISCARDED              Xmit                 0
                       Rcvd                 0
```

The elements of this display are described as follows:

CANUREACH Number of CanUReach Station messages transmitted or received.
ICANUREACH Number of ICanReach Station messages transmitted or received.
REACH_ACK Number of Reach Acknowledgment messages transmitted or received.
DGRMFRAME Number of Datagram Frame messages transmitted or received.
XIDFRAME Number of XID frames transmitted or received.
CONTACT Number of Contact Remote Station messages transmitted or received.
CONTACTED Number of Remote Station Contacted messages transmitted or received.
RESTART_DL Number of Restart Data Link messages transmitted or received.
DL_RESTARTED Number of Data Link Restarted messages transmitted or received.
INFOFRAME Number of Information (I) Frame messages transmitted or received.
ENTERBUSY Number of Enter Link Station Busy messages transmitted and received.
EXITBUSY Number of Exit Link Station Busy messages transmitted or received.
HALT_DL Number of Halt Data Link messages transmitted or received.
DL_HALTED Number of Data Link Halted messages transmitted or received.
NETBIOS_NQ Number of NetBIOS Name Query messages transmitted or received.
NETBIOS_NR Number of NetBIOS Name Recognized messages transmitted or received.
DATAFRAME Number of Dataframe messages transmitted or received.
NETBIOS_ANQ Number of NetBIOS Add Name Query messages transmitted or received.
NETBIOS_ANR Number of NetBIOS Add Name Response messages transmitted or received.
HALTDL_NO_ACK Number of Halt Data Link No Acknowledgment messages transmitted or received.
TEST_CIR_REQ Number of Test Circuit Request messages transmitted or received.
TEST_CIR_RSP Number of Test Circuit Response messages transmitted or received.
OTHERS Number of messages undefined in RFC 1434 transmitted or received.
DISCARDED Number of Discarded messages transmitted or received.

DVMRP Service

The following is an example of a display generated by the SHow -SYS STATistics -DVMRP command:

```
ACCUMULATED VALUES
== DVMRP statistics ========
Rcvd from MOSPF             0
Sent to MOSPF               0
Pruned by MOSPF             0
== DVMRP statistics ======== === 1=== === 2=== === 3=== === 4===
Pkts Received (total)       0        0        2011     0
   Reports                  0        0        12966    0
   Prunes                   0        0        667      0
   Grafts                   0        0        0        0
   Graft Acks               0        0        0        0
Pkts Transmitted            0        0        0        0
   Reports                  12959    12959             0
   Prunes                     0                   0         0
   Grafts                     0         0         0         0
   Graft Acks                 0         0         0         0
Pkts Forwarded                5960K     671       0         0
Pkts Discarded                0         0         0         0
   NoRoute                    0         0         0         0
   WrongPort                  0         0         0         0
   Unknown Type               0         0         0         0
   MiscErr                    0         0         0         0
IP over IP Statistics:        0         0         0         0
   Pkts Received              0         0         0         0
   Pkts Discarded             0         0         0         0
```

The elements of this display are described as follows:

DVMRP Statistics

Rcvd from MOSPF Total number of packets received from Multicast Open Shortest Path First Protocol (MOSPF) domains.

Sent to MOSPF Total number of packets transmitted to MOSPF domains.

Pruned by MOSPF Total number of packets pruned by MOSPF domains for no listener in the MOSPF domains.

Pkts Received

Total number of DVMRP packets received.

Reports Number of route update packets received.

Prunes Number of Prune packets received from a downstream neighbor router.

Grafts Number of Graft packets received from a downstream neighbor router.

Graft Acks Number of Graft Acknowledge packets received from the upstream parent router.

Pkts Transmitted

Total number of DVMRP packets transmitted.

Reports Number of route update packets transmitted.

Prunes Number of Prune packets transmitted to the upstream parent router.

Grafts Number of Graft packets transmitted to the upstream parent router.

Graft Acks Number of Graft Acknowledge packets transmitted to any downstream neighbor router.

Pkts Forwarded

Total number of packets forwarded.

Pkts Discarded

NoRoute Number of packets received when there is no route to the source.

WrongPort Number of packets received on a port that is not used to forward to the source.

Unknown Type Number of packets received that are of unknown type.

MiscErr Number of packets received when the system is out of resources.

IP over IP Statistics

Pkts Received Total number of IP-over-IP packets received.

Pkts Discarded Total number of IP-over-IP packets discarded.

FR Service

The following is an example of the display generated by the SHow -SYS STATistics -FR command:

```
ACCUMULATED VALUES
== FR statistics =========== === 1===  === 3===  === 5===  === 7===
Frame Relay Port Statistics:
LMI Frames Xmit               0         0         0         0
LMI Frames Recv               0         0         0         0
Invalid DLCI frames           0         0         0         0
Small Frames Recv             0         0         0         0
Port Status Change            0         0         0         0
```

The elements of this display are described as follows:

Frame Relay Port Statistics

LMI Frames Xmit	Number of messages successfully transmitted.
LMI Frames Recv	Number of messages successfully received.
Invalid DLCI frames	Number of DLCI frames invalid.
Small Frames Recv	Number of small frames successfully received.
Port Status Change	Number of port status changes.

IDP Service

The following is an example of the display generated by the SHow -SYS STATistics -IDP command:

```
ACCUMULATED VALUES
== IDP statistics ======== === 1===  ==== 3=== === 5===  === 7===
IDP Statistics:
Received                    0         0          0         0
Forwarded                   0         0          0         0
Passed to client            0         0          0         0
Xmitted                     0         0          0         0
Discarded                   0         0          0         0
```

The elements of this display are described as follows:

IDP Statistics

Received — Indicates the number of packets received on a port since boot-up time or the last flushing. This number is the total number of packets received from the network including Forwarded packets, Broadcast and Unicast packets addressed to the router and successfully delivered to initial domain identifier (IDP) clients, and some Discarded packets.

Forwarded — Indicates the number of packets routed successfully to other ports since boot-up time or the last flushing. Those packets generated by the router itself are not included in this category.

Passed to client — Indicates the number of broadcast packets or unicast packets addressed to the router and successfully delivered to proper IDP clients. On the router, only Xerox Network Systems (XNS) Routing Information Protocol (RIP) and partioned emulation programming (PEP) clients reside. So this number is the total number of RIP or PEP packets received on the port since boot-up time or the last flushing.

Xmitted — Indicates the number of packets generated and transmitted by the router since boot-up time or the last flushing. There can be only two types of packets (RIP and PEP) generated by the router.

Discarded — Indicates the number of packets discarded by IDP because of various errors such as bad framed packets, packets without any data, packets destined to other networks when IDP routing is turned off, packets addressed to the router but no clients available, etc. This number includes bad packets received from either networks or IDP clients.

IP Service

The following is an example of the display generated by the SHow -SYS STATistics -IP command:

```
ACCUMULATED VALUES
========================= IP statistics =====================
IP Datagram Rates (pkts/s):
   Received from networks                     0
   Forwarded elsewhere                        0
   Passed to local client                     0
   Originated by system                       0

IP Datagrams (totals):
   Received from networks                     16641
   Forwarded elsewhere                        0
   Discarded by filtering                     0
   Passed to local client                     16641
   Originated by system                       846

IP Fragmentation:
   Packets successfully frag'd                0
   Packets failed to fragment                 0
   Total fragments generated                  0
   Fragments Received                         18
   Datagrams Assembled                        2
   Reassembly Failures                        0

ICMP Messages:
   Received Messages                          0
   Receive Failures                           0
   Transmit Messages                          0
   Transmit Failures                          0

Errors:
   Received Bad Header                        0
   Incorrect IP Address                       32
   Route Lookup Failed                        0
   No Local Protocol                          0
   Other RCV Errs                             0
   Other XMT Errs                             0
```

The elements of this display are described as follows:

IP Datagram Rates (pkts/s)

Received from networks — Number of packets per second received from networks.

Forwarded elsewhere — Number of packets per second forwarded to other networks.

Passed to local client — Number of packets per second destined for this bridge.

Originated by system — Number of packets per second that originated from this router.

IP Datagrams (totals)

Received from networks
Total number of packets received from networks.

Forwarded elsewhere
Total number of packets forwarded to other networks.

Discarded by filtering
Number of packets discarded because they met the conditions of a standard filter.

Passed to local client
Total number of packets destined for this router (local client protocol: Transmission Control Protocol (TCP), User Datagram Protocol (UDP).

Originated by system
Total number of packets that originated from this router.

IP Fragmentation

Packets successfully frag'd
Number of packets too large to be sent that were successfully broken up into smaller units.

Packets failed to fragment
Number of packets too large to be sent that were not successfully broken up into smaller units.

Total fragments generated
Total number of units created by packet fragmentation.

Fragments Received
Number of fragments received from other networks.

Datagrams Assembled
Number of packets successfully reassembled once received.

Reassembly Failures
Number of packets not successfully reassembled before reaching the destination router.

ICMP Messages

Received Messages
Number of messages successfully received.

Receive Failures
Number of messages not successfully received because of checksum errors or because they were too short.

Transmit Messages
The total number of transmit messages.

Transmit Failures
Number of transmit messages not received by the client in the router because of an error in address or format.

Errors

Received Bad Header
Number of errors caused by bad headers.

Incorrect IP Address
Number of errors caused by invalid Internet addresses.

Route Lookup Failed
Number of errors caused by route lookup failure; for example, because the next hop or network was unreachable.

No Local Protocol
Number of errors generated because the client or the protocol was not listed in the routing table.

Other RCV Errs
Number of general receive errors.

Other XMT Errs
Number of general transmit errors.

IPX Service

The following is an example of the display generated by the SHow -SYS STATistics -IPX command:

```
ACCUMULATED VALUES
== IPX statistics ============ === 1=== ===2===  === 3=== === 4===
IPX Statistics:
Received                      0        0        0        0
Forwarded                     0        0        0        0
Passed to client              0        0        0        0
Xmitted                       0        0        0        0
Discarded                     0        0        0        0
IPX SPOOF Statistics:
Watchdog Resp (out)           255      -        25       -
```

The elements of this display are described as follows:

IPX Statistics

Received

Number of packets received on a port since boot-up time or the last flush time. This number is the total number of packets received from the network including Routed packets, Broadcast and Unicast packets addressed to the router and successfully delivered to IPX clients, and some Discarded packets.

Forwarded

Number of packets routed successfully to other ports since boot-up time or the last flush time. Those packets generated by the router itself are not included in this category.

Passed to client

Number of Broadcast packets or Unicast packets addressed to the router and successfully delivered to proper IPX clients. On the router resides only IPX RIP and IPX SAP clients. So this number is the total number of RIP or SAP packets received on the port since boot-up time or the last flush time.

Xmitted

Number of packets generated and transmitted by the router since boot-up time or the last flush time. There can be only two types of packets (RIP and SAP) generated by the router.

Discarded

Number of packets discarded by IPX because of various errors such as bad framed packets, packets without any data, packets destined to other networks when IPX routing is turned off, packets addressed to the router but no clients available, etc. This number includes bad packets received from either networks or IPX clients.

IPX SPOOF Statistics

Watchdog Resp (out)

Number of NCP KeepAliveResponse packets generated by the bridge/router and sent back out on the port as a result of NCP watchdog spoofing being active.

ISIS Service

The following is an example of the display generated by the SHow -SYS STATistics -ISIS command:

```
ACCUMULATED VALUES
========================= ISIS statistics ========================
ISIS statistics
  PDU format error          0
  Corrupted LSP             0
  L1 LinkStateData overload 0
  L2 LinkStateData overload 0
  AreaAddress dropped       0
  SeqNumber overflow        0
  SeqNumber skipped         0
  Own LSP purged            0

== ISIS statistics =========  === 1===  === 3===  === 5===  === 7===
Adjacency change             0         0         0         0
Adjacency reject             0         0         0         0
Corrupted LSP rcvd           0         0         0         0
L2 DIS changes               0         0         0         0
L1 DIS changes               0         0         0         0
PDU sent                     0         0         0         0
PDU rcvd                     0         0         0         0
```

```
ID Length mismatch            0         0         0         0

Authentication
   L1 error                   0         0         0         0
   L2 error                   0         0         0         0
   Hello error                0         0         0         0
```

The elements of this display are described as follows:

ISIS Statistics

PDU format error
Number of times an Intermediate System-to Intermediate System (ISIS) protocol data unit (PDU) with an incorrect format was received.

Corrupted LSP
Number of times an link state packet (LSP) with unacceptable format or bad information was received.

L1 LinkStateData overload
Number of times this router encountered memory resource problems when trying to store a Level 1 LSP PDU.

L2 LinkStateData overload
Number of times this router encountered memory resource overload problems when trying to store a Level 2 LSP PDU.

AreaAddress dropped
Too many area addresses in the area, causing a manual area address on the route to be dropped.

SeqNumber overflow
The sequence number field in the LSP generated by the router has reached the maximum allowed value (approximately 4 billion), which forces the router to go out of service temporarily.

SeqNumber skipped
Number of times another router claims to own an LSP generated by this router, but with a high sequence number.

Own LSP purged
Number of times another router has purged an LSP generated by this router.

ISIS statistics

Adjacency change
Number of times the adjacency state with nearby routers has gone into UP or DOWN state.

Adjacency reject
Number of times an adjacency is rejected to this router because of mismatch in the area address of the two routers.

Corrupted LSP rcvd
Number of times an LSP is received on each interface with an unacceptable format or bad information.

L2 DIS changes
Number of times the Level 2 designated intermediate system has changed.

L1 DIS changes
Number of times the Level 1 designated intermediate system has changed.

PDU sent
Number of ISIS PDUs sent, including hello, CSNP, PSNP, and LSP.

PDU rcvd
Number of ISIS PDUs received, including hello, CSNP, PSNP, and LSP. The counter includes packets received with format errors.

ID Length mismatch
Number of ISIS PDUs received with mismatched ID length fields. All PDUs are counted (hello, CSNP, PSNP, and LSP). Implementations with mismatched ID lengths cannot interoperate. This implementation supports an ID length of six octets.

L1 error
Number of L1, LSP, CSNP, or PSNP PDUs received with mismatched Level 1 password.

L2 error
Number of L2, LSP, CSNP, or PSNP PDUs received with mismatched Level 2 password.

Hello error
Number of Level 1 or Level 2 hello PDUs received with mismatched hello password.

LLC2 Service

The following is an example of the display generated by the **s**How -SYS STATistics -LLC2 command:

```
ACCUMULATED VALUES
== LLC2 statistics ======  === 1===  ==== 3=== ==== 5=== === 7===
Test Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
Xid Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
UI-Data Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
Sabme Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
I-Data Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
I-Data Bytes
   Received                0         0         0         0
   Transmitted             0         0         0         0
RR Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
RNR Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
Reject Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
Disc Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
UA Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
DM Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
FRMR Frames
   Received                0         0         0         0
   Transmitted             0         0         0         0
```

The elements of this display are described as follows:

Test Frames

Received Number of test frames received per port.

Transmitted Number of test frames transmitted per port.

Xid Frames

Received Number of Xid frames received per port.

Transmitted Number of Xid frames transmitted per port.

UI-Data Frames

Received Number of Unnumbered Information frames received per port. These frames are typically sent and received by NetBIOS and LAN Network Manager Logical Link Control (LLC) protocols.

Transmitted Number of Unnumbered Information frames transmitted per port.These frames are typically sent and received by NetBIOS and LAN Network Manager LLC protocols.

Sabme Frames

Received Number of set asynchronous balanced mode extended frames received per port.

Transmitted Number of set asynchronous balanced mode extended frames transmitted per port.

I-Data Frames

Received Number of valid I-data frames received per port, not including retransmissions.

Transmitted Number of valid I-data frames transmitted per port, not including retransmissions.

I-Data Bytes

Received Number of valid I-data bytes received per port, not including MAC address bytes.

Transmitted Number of valid I-data bytes transmitted per port, not including MAC address bytes.

RR Frames

Received Number of receiver ready frames received per port.

Transmitted Number of receiver ready frames transmitted per port.

RNR Frames

Received Number of receiver not ready frames received per port.

Transmitted Number of receiver not ready frames transmitted per port.

Reject Frames

Received Number of reject frames received per port.

Transmitted Number of reject frames transmitted per port.

Disc Frames

Received Number of disconnect frames received per port.

Transmitted Number of disconnect frames transmitted per port.

UA Frames

Received Number of unnumbered acknowledgment frames received per port.

Transmitted Number of unnumbered acknowledgment frames transmitted per port.

DM Frames

Received Number of disconnect mode frames received per port.
Transmitted Number of disconnect mode frames transmitted per port.

FRMR Frames

Received Number of frame reject frames received per port.
Transmitted Number of frame reject frames transmitted per port.

MIP Service

The following is an example of a display generated by the SHow -SYS STATistics -MIP command:

IGMP is not a service in the user interface. However, you can still obtain IGMP statistics through the MIP Service.

```
ACCUMULATED VALUES
===== MIP statistics =======   ==1==    ==2==   ==3==   ==4==
Multicast IP datagram          1093     5957K   0       0
Pkts Received (total)          1078     1620    20102   0
   Queries                     0        0       0       0
   Reports                     1078     1620    0       0
   Leaves                      0        0       0       0
Pkts Transmitted               0        0       1943    0
Pkts Discarded (total)         0        0       0       0
   TooShort                    0        0       0       0
   Version Err                 0        0       0       0
   Chksum Err                  0        0       0       0
   Unknown Type                0        0       0       0
```

The elements of this display are described as follows:

Multicast IP Datagram Total number of multicast IP datagrams received.

Pkts Received The total number of Internet Group Management Protocol (IGMP) packets received, including DVMRP packets.

Queries Number of Host Query packets received.
Reports Number of Host Report packets received.
Leaves Number of Host Leave Group packets received.

Pkts Transmitted Total number of IGMP packets transmitted, including DVMRP packets.

Pkts Discarded Total number of packets discarded.

TooShort Number of packets received with the data length too short.
Version Err Number of packets received that have a bad version number.
Chksum Err Number of packets received that have bad checksums.
Unknown Type Number of packets received with unknown type.

MOSPF Service

The following is an example of a display generated by the SHow -SYS STATistics -MOSPF command:

```
ACCUMULATED VALUES
== MOSPF statistics ========
SPF calculations              18
```

```
Cache flushed                13
Rcvd from DVMRP              152610
Sent to DVMRP                8
Prune back DVMRP             3
Resource error               0
== MOSPF statistics ======== === 1=== === 2=== === 3=== === 4===
Receive                      152607    8         0         0
   Good                      0         0         0         0
   No Route                  0         0         0         0
   Bad MAC address           0         0         0         0
   Wrong source              0         0         0         0
   TTL exceeded              0         0         0         0
   Recv Disabled             0         0         0         0
Transmit                     0         0         0         0
   Good                      0         0         152231    0
   TTL too small             0         0         0         0
   Xmit Disabled             0         0         0         0
   Failure                   0         0         0         0
```

The elements of this display are described as follows:

MOSFP Statistics

SPF calculations Number of route computations, using Dijkstra's algorithm.
Cache flushed Number of times forwarding cache was flushed, possibly due to network topology changes.
Rcvd from DVMRP Total number of packets received from DVMRP domains.
Sent to DVMRP Total number of packets transmitted to DVMRP domains.
Prune back DVMRP Number of Prune messages sent back to DVMRP because the received multicast packets have no listener in the MOSPF domain.
Resource error Number of failures in obtaining memory resources.

Receive

Good Number of good receive packets.
No Route No recipient; packets discarded.
Bad MAC Address Unicast or multicast MAC address expected, but the incoming packets were not using the correct MAC address format.
Wrong Source Packets arrived from upstream neighbor other than from the correct upstream neighbor.
TTL exceeded Time-to-live value exceeded on the receive packet.
Recv Disabled MOSPF is disabled on the interface.

Transmit

Good Number of good transmit packets.
TTL too small TTL too small to reach any of the downstream destinations, or TTL smaller than the THreshold (in the MIP service) parameter.
Xmit Disabled MOSPF is disabled on the interface.
Failure Transmit queue overflow; packets discarded.

NLSP Service

The following is an example of the display generated by the SHow -SYS STATistics -NLSP command:

```
ACCUMULATED VALUES
================= NLSP statistics ===================
Corrupted LSP                          0
AreaAddress dropped                    0
SeqNumber overflow                     0
SeqNumber skipped                      0
Own LSP purged                         0
L1 LinkStateData overload              0
L2 LinkStateData overload              0
== NLSP statistics ==================  1=======  2=======  4=======
Adjacency change                       0         0         0
Adjacency reject                       0         0         0
Corrupted LSP rcvd                     0         0         0
PDU sent                               0         0         0
PDU rcvd                               0         0         0
PDU rcvd format err                    0         0         0
L1 DIS change                          0         0         0
L2 DIS change                          0         0         0
Authentication:
   L1 error                            0         0         0
   L2 error                            0         0         0
   Hello error                         0         0         0
```

The elements of this display are described as follows:

NLSP statistics

Corrupted LSP
Number of times an LSP with unacceptable format or bad information was received.

AreaAddress dropped
Too many area addresses in the area, causing a manual area addresses on the router to be dropped.

SeqNumber overflow
The sequence number field in the LSP generated by the router has reached the maximum allowed value (approximately 4 billion), which forces the router to go out of service temporarily.

SeqNumber skipped
Number of times another router claims to own an LSP generated by this router, but with a high sequence number.

Own LSP purged
Number of times another router has purged an LSP generated by this router.

L1 LinkStateData overload
Number of times this router encountered memory resource problems when trying to store a Level 1 LSP PDU.

L2 LinkStateData overload
Number of times this router encountered memory resource overload problems when trying to store a Level 2 LSP PDU.

Adjacency change
Number of times the adjacency state with nearby routers has gone into UP or DOWN state.

Adjacency reject
Number of times an adjacency is rejected to this router because of mismatch in the area address of the two routers.

Corrupted LSP rcvd
Number of times an LSP is received on each interface with an unacceptable format or bad information.

PDU sent
Number of ISIS PDUs sent, including Hello, CSNP, PSNP, and LSP.

PDU rcvd
Number of ISIS PDUs received, including Hello, CSNP, PSNP, and LSP. The counter includes packets received with format errors.

PDU rcvd format err
Number of times an ISIS PDU with an incorrect format was received.

L1 DIS change
Number of times the Level 1 designated intermediate system has changed.

L2 DIS change
Number of times the Level 2 designated intermediate system has changed.

Authentication

L1 error
Number of L1, LSP, CSNP, or PSNP PDUs received with mismatched Level 1 password.

L2 error
Number of L2, LSP, CSNP, or PSNP PDUs received with mismatched Level 2 password.

Hello error
Number of Level 1 or Level 2 Hello PDUs received with mismatched hello password.

NRIP Service

The following is an example of the display generated by the SHow -SYS STATistics -NRIP command:

```
ACCUMULATED VALUES
== NRIP statistics ============ 1========   2========   4=========
RIP Updates(out)                0           0           0
RIP Updates(in)                 0           0           0
RIP Requests(out)               0           0           0
RIP Requests(in)                0           0           0
RIP Replies(out)                0           0           0
RIP Discarded                   0           0           0
```

The elements in the display are described as follows:

NRIP statistics

RIP Updates(out)
Number of IPX RIP broadcasts transmitted by the router since the boot time or the last flush time. Both regular updates and triggered updates are included in this category.

RIP Updates(in)
Number of IPX RIP broadcasts received on a port by the router.

RIP Requests (out)
Number of IPX requests generated by the router.

RIP Requests(in)
Number of IPX requests received by the router.

RIP Replies(out)
Number of RIP replies generated by the router in response to RIP requests. The number of RIP replies can be bigger than the number of RIP requests depending on the current number of IPX RIP table entries.

RIP Discarded
Number of RIP packets dropped by the router because of various errors such as packets received from unknown networks, lost packets, and so forth.

OSPF Service

The following is an example of the display generated by the SHow -SYS STATistics -OSPF command:

```
ACCUMULATED VALUES
========================= OSPF statistics ========================
SPF calculations            1729
Resource error              0

== OSPF statistics ======== 1=======  2=======  3=======  4=======
Hello Rcvd                  0         0         0         0
Hello Xmit                  0         7         0         0
DD Rcvd                     0         0         0         0
DD Xmit                     0         0         0         0
LSR Rcvd                    0         0         0         0
LSR Xmit                    0       0       0       0
LSA Rcvd                    0       0       0       0
LSA Xmit                    0       0       0       0
LSU Rcvd                    0       0       0       0
LSU Xmit                    0       0       0       0

Number of DR Election       0       1       0       0
Adjacency UP Events         0       0       0       0
Adjacency DOWN Events       0       0       0       0
Errors:
  Xmit fail                 0       0       0       0
  Rcv bad packet header     0       0       0       0
  Mismatch HelloTime        0       0       0       0
  Mismatch RouterDeadTim    0       0       0       0
  Mismatch subnet/mask      0       0       0       0
  Mismatch area ID          0       0       0       0
  Unknown packet type       0       0       0       0
  Authentication Error      0       0       0       0
  Packet Checksum Error     0       0       0       0
  LSA Checksum Error        0       0       0       0
```

The elements of this display are described as follows:

OSPF Statistics

SPF calculations Number of times the router has performed the SPF calculation.

Resource error Number of times OSPF failed to obtain buffers for packet transmission or LSA storage.

Hello Rcvd Number of Hello messages received by the router.

Hello Xmit Number of Hello messages sent by the router.

DD Rcvd Number of data description packets received by the router.

DD Xmit Number of data description packets transmitted by the router.

LSR Rcvd Number of link state information request packets received by the router.

LSR Xmit Number of link state information request packets sent by the router.

LSA Rcvd Number of link state acknowledgment packets received by the router.

LSA Xmit Number of link state acknowledgment packets sent by the router.

LSU Rcvd Number of link state update packets received by the router.

LSU Xmit Number of link state update packets sent.

Number of DR Election Number of times designated router election has been performed.

Adjacency UP Events Number of times that an adjacency has gone from Down to Up state.

Adjacency DOWN Events Number of times that an adjacency has gone from Up to Down state.

Errors

Xmit Fail Number of transmission congestions experienced while transmitting OSPF packets. Congestion happens when the transmit queue overflows, and the OSPF packets are dropped.

Rcv bad packet header
Number of errors received by the router because of faulty packet headers.

Mismatch HelloTime
Number of Hello packets received with mismatched HelloTimes. In order for two OSPF systems to establish an adjacency, both must have identical HelloTime values.

Mismatch RouterDeadTime
Number of Hello packets received with mismatched RouterDeadTimes. In order for two OSPF systems to establish an adjacency, both must have identical RouterDeadTime values.

Mismatch subnet/mask
Number of Hello packets received with mismatched subnet or mask. In order for two OSPF systems to establish an adjacency, both must have identical IP subnets and masks.

Mismatch area ID
Number of adjacency rejections due to mismatched area ID. In order for two neighbors to become adjacent, they must both be configured with identical OSPF area IDs.

Unknown packet type
Number of OSPF packets received that are not one of the known (Hello, DD, LSR, LSA and LSU) packet types.

Authentication Error
Number of packets received that fail authentication.

Packet Checksum Error
Number of packets received with checksum errors.

LSA Checksum Error
Number of link state advertisements with checksum errors.

PATH Service

The following is an example of the display generated by the SHow -SYS STATistics -PATH command. This example also applies for the BRidge Service and the PPP Service.

```
ACCUMULATED VALUES
== PATH statistics============  1======= 2======= 3======= 4=======
Rcvd Packets                    9265289  0        0        1297
   Bytes                        2542M    0        0        54488
   Err: CRC                     5        0        0        0
      Framing                   360      0        0        0
      Too Long                  89       0        0        0
      Lost                      0        0        0        0
      Parity                    0        0        0        0
      Break                     0        0        0        0
Xmit Packets                    45445    0        0        55296
   Bytes                        3065816  0        0        3135915
   Err: Deferred                779      0        0        0
      Collision                 704      0        0        0
      Late Collisions           0        0        0        0
      Xcess Collision           0        0        0        0
      Carrier Loss              0        0        0        0
      Underrun                  0        0        0        0
Discard: Buf Overrun            0        0        0        0
   Congestion                   0        0        0        0
Utilization: (%)                7        0        0        0
Rcv Good: pckt/Sec              119      0        0        0
      Byte/Sec                  32826    0        0        0
Xmit Good: Pkt/Sec              0        0        0        0
      Byte/Sec                  0        0        0        0
```

The elements of this display are described as follows:

Rcvd Packets Number of good packets received on the interface.

Bytes Number of good bytes received on the specified interface. Includes headers but not cyclic redundancy check (CRC) bytes.

Err: CRC Number of frames that were received but failed the cyclic redundancy check.

Framing Number of frames that were received but were not on a 16-bit Too Long boundary.

Too Long Number of frames discarded because the packet length was longer than the maximum packet length allowed.

Lost Number of receptions aborted because the CPU could not provide the controller chip with memory quickly enough.

Parity Number of asynchronous parity receive errors.

Break Number of asynchronous break receive errors.

Xmit Packets Number of good packets transmitted.

Failure to terminate your Ethernet network will result in the false detection of transmission on the nonterminated BNC connector. This is due to Ethernet module sensitivity to RF transmissions from nearby boards. If this occurs, the Xmit Packets number displayed in the PATH Service statistics will be incorrect.

Bytes Number of bytes transmitted from a port. Includes headers but not CRC bytes.

Err: Deferred Number of frames that could not be transmitted because of existing traffic on the link. Transmission would have resulted in a collision. A later attempt was made to transmit the frame.

Collision Number of frames that experienced a collision during the first attempt to transmit.

Late Collisions Number of frames that received a collision outside of the preamble.

Xcess Collision Number of frames not discarded after 16 consecutive collisions.

Carrier Loss Number of frames that experienced a loss of the carrier signal during transmission.

Underrun Number of transmissions aborted because the CPU could not provide the controller chip with data fast enough.

Discard

Buf Overrun Number of good frames lost because of memory overrun. This occurs when the system does not have enough memory to transfer the packet internally for further processing.

Congestion Number of frames that could not be transmitted because of transmit queue overflow.

Utilization: The percentage of time the carrier sense signal was active during the specified interval. The percentage of utilization displayed for HSS ports is based on a full-duplex link. For example, a 64 Kbps circuit can transmit and receive 64 Kbps simultaneously. If this link were transmitting at 64 Kbps and receiving nothing, the percentage of utilization would be 50 percent.

Rcv Good:

Pkt/Sec Number of good packets received per second.

Byte/Sec Number of good bytes received per second.

Xmit Good

Pkt/Sec Number of good packets transmitted per second.

Byte/Sec Number of good bytes transmitted per second.

PORT Service

The following is an example of the display generated by the SHow -SYS STATistics -PORT command:

```
ACCUMULATED VALUES
== PORT statistics ========= 1=======  2=======  3=======  4=======
Rcvd : Packets               574285    0         0         282
   Bytes                     299912K   0         0         11858
   Multicast                 35752     0         0         282
   Broadcast                 5080      0         0         0
Xmit : Packets               9801      0         0         11820
   Bytes                     666413    0         0         670004
   Multicast                 722       0         0         8529
   Broadcast                 722       0         0         3288
Filter : Custom              0         0         0         0
Discard:
   Buf Overrun               0         0         0         0
   Congestion                0         0         0         0
DialOnDemand Mode:
   DodCallsMade              0         0         0         0
   DodCallsFail              0         0         0         0
   DodUpTime                 0         0         0         0
   DodPktsOut                0         0         0         0
```

The elements of this display are described as follows:

Rcvd

Packets Number of good packets received on a specified port.

Bytes Number of good bytes received on a specified port.

Multicast Number of multicast packets received. Multicast packets are sent to more than one station on the network.

Broadcast Number of broadcast packets received. Broadcast packets are sent to the entire network.

Xmit

Packets Number of good packets sent by ports.

Bytes Number of good bytes sent by ports.

Multicast Number of multicast packets sent. Multicast packets are sent to more than one station on the network.

Broadcast Number of broadcast packets sent. Broadcast packets are sent to the entire network.

Filter

Custom Number of packets that matched the custom filters and were therefore discarded.

Discard

Buffer Overrun Number of packets discarded because of buffer overrun. This occurs when the system does not have enough memory to transfer the packet internally for further processing.

Congestion Number of packets discarded because of congestion. This occurs when a packet cannot be transmitted within a specified amount of time.

DialOnDemand Mode

DodCallsMade Number of outgoing calls successfully initiated by the port operating in dial-on-demand mode.

DodCallsFail Number of outgoing calls unsuccessfully initiated by the port operating in dial-on-demand mode.

DodUpTime Length of time in seconds that the primary path of a port is up while operating in dial-on-demand mode.

DodPktsOut Number of user-data packets sent out by the port operating in dial-on-demand mode.

PPP Service

The following is an example of the display generated by the SHow -SYS STATistics -PPP command, which displays received and transmitted LCP packets:

```
ACCUMULATED VALUES
== PPP statistics =============  1======  3======  5======  7======
LCP path statistics :
Rcvd Conf-Request-               0        -        -        0
   Conf-Ack                      -        0        -        0
   Conf-Nak                      -        0        -        0
   Conf-Reject                   -        0        -        0
   Term-Request                  -        0        -        0
   Term-Ack                      -        0        -        0
   Code-Reject                   -        0        -        0
   Protocol-Reject               -        0        -        0
   Echo-Request                  -        20       -        420
   Echo-Reply                    -        420      -        420
   Discard-Req                   -        0        -        0
   Link Quality Rpt              -        0        -        0
   Unknown Code                  -        0        -        0
Xmit Conf-Request-               0        -        -        0
   Conf-Ack                      -        0        -        0
   Conf-Nak                      -        0        -        0
   Conf-Reject                   -        0        -        0
   Term-Request                  -        0        -        0
   Term-Ack                      -        0        -        0
   Code-Reject                   -        0        -        0
   Protocol-Reject               -        1        -        0
   Echo-Request                  -        428      -        428
   Echo-Reply                    -        428      -        427
   Discard-Req                   -        0        -        0
   Link Quality Rpt              -        0        -        0
```

The elements of this display are described as follows:

LCP path statistics

Rcvd

Conf-Request Number of received LCP packets of code 1 for configure request.

Conf-Ack Number of received LCP packets of code 2 for configure acknowledgment.

Conf-Nak Number of received LCP packets of code 3 for configure negative acknowledgment.

Conf-Reject Number of received LCP packets of code 4 for configure rejection.

Term-Request Number of received LCP packets of code 5 for terminate request.

Term-Ack Number of received LCP packets of code 6 for terminate acknowledgment.

Code-Reject Number of received LCP packets of code 7 for code rejection.

Protocol-Reject Number of received LCP packets of code 8 for protocol rejection.

Echo-Request Number of received LCP packets of code 9 for echo request.

Echo-Reply Number of received LCP packets of code 10 for echo reply.

Discard-Req Number of received LCP packets of code 11 for discard request.

Link Quality Rpt Number of received LCP packets of code 12 for link quality report.

Unknown Code Number of received LCP packets of unknown code.

Xmit

Conf-Request Number of transmitted LCP packets of code 1 for configure request.

Conf-Ack Number of transmitted LCP packets of code 2 for configure acknowledgment.

Conf-Nak Number of transmitted LCP packets of code 3 for configure negative acknowledgment.

Conf-Reject Number of transmitted LCP packets of code 4 for configure rejection.

Term-Request Number of transmitted LCP packets of code 5 for terminate request.

Term-Ack Number of transmitted LCP packets of code 6 for terminate acknowledgment.

Code-Reject Number of transmitted LCP packets of code 7 for code rejection.

Protocol-Reject Number of transmitted LCP packets of code 8 for protocol rejection.

Echo-Request Number of transmitted LCP packets of code 9 for echo request.

Echo-Reply Number of transmitted LCP packets of code 10 for echo reply.

Discard-Req Number of transmitted LCP packets of code 11 for discard request.

Link Quality Rpt Number of transmitted LCP packets of code 12 for link quality report.

RIPIP Service

The following is an example of the display generated by the SHow -SYS STATistics -RIPIP command:

```
ACCUMULATED VALUES
================= RIPIP Statistics ==== 1======= 2======= 3=======
RIP/IP Statistics:
Incoming Packets                        16347    0        0
   Request Updates                      0        0        0
   Response Updates                     16347    0        0
   Discarded Updates                    0        0        0
Outgoing Packets                        16347    0        0
   Request Updates                      0        0        0
   Regular Responses                    16347    0        0
   Triggered Responses                  0        0        0
```

The elements of this display are described as follows:

Incoming Packets

Request Updates — Number of incoming RIP request packets on a port. RIP request packets request routing information.

Response Updates — Number of incoming RIP response packets on a port. RIP response packets are sent to convey routing information.

Discarded Updates — Number of update packets discarded on a port because they originated from an unconfigured neighbor.

Outgoing Pkts

Request Updates — Number of outgoing request update packets sent on a port. Request updates are sent either when the router is booted or when RIP has been configured.

Regular Responses — Number of outgoing regular update packets sent on a port. Regular updates are router information packets that are sent out by the router at regular intervals.

Triggered Responses — Number of triggered response packets sent on a port. Triggered responses are router information packets sent out immediately when a network becomes unreachable.

RIPXNS Service

The following is an example of the display generated by the SHow -SYS STATistics -RIPXNS command:

```
ACCUMULATED VALUES
== RIPXNS statistics ===========  1======  3======  5======  7======
RIPXNS Statistics
RIP Updates(out)                  0        0        0        0
RIP Updates(in)                   0        0        0        0
RIP Requests(out)                 0        0        0        0
RIP Requests(in)                  0        0        0        0
RIP Replies(out)                  0        0        0        0
RIP Discarded                     0        0        0        0
```

The elements of this display are described as follows:

RIP Updates(out) Number of XNS RIP broadcasts transmitted by the router since the boot-time or last flushing. Both regular updates and triggered updates will be included in this category.

RIP Updates(in) Number of XNS RIP broadcasts received on a port by the router.

RIP Requests(out) Number of RIP requests generated by the router.

RIP Requests(in) Number of RIP requests received by the router.

RIP Replies(out) Number of RIP replies generated by the router in response to RIP requests. The number of RIP replies can be bigger than the number of RIP requests in case RIP replies require more than one packet.

RIP Discarded Number of RIP packets dropped by the router because of various errors such as packets received from unknown networks, request packets received when IDP routing is turned off, lost packets, and so forth.

SAP Service

The following is an example of the display generated by the **s**How -SYS STATistics -SAP command:

```
ACCUMULATED VALUES
== SAP statistics ==================  1=======  2=======  4=======
SAP Updates(out)                      0         0         0
SAP Updates(in)                       0         0         0
SAP Requests(out)                     0         0         0
SAP Requests(in)                      0         0         0
SAP Replies(out)                      0         0         0
SAP Discarded                         0         0         0
```

The elements of the display are described as follows:

SAP Statistics

SAP Updates (out) Number of SAP updates generated by the router since the boot time or the last flush time. Periodic updates or incremental updates on serial interfaces are included in this category. Depending on the current number of SAP table entries, the number of SAP updates can vary.

SAP Updates (in) Number of SAP broadcasts received on a port by the router.

SAP Requests (out) Number of SAP requests generated by the router. Normally the router generates SAP requests on serial interfaces to learn new server information from the other router when a port comes up or new routes are learned from a remote router.

SAP Requests (in) Number of SAP queries received by the router.

SAP Replies (out) Number of SAP replies generated by the router in response to SAP queries. In general, the number of SAP replies is more than the number of SAP requests because most of the time more than one packet is required to satisfy one SAP request.

SAP Discarded Number of SAP packets dropped by the router because of errors such as bad framed packets, and packets with unknown SAP packet types.

SHDlc Service

The following is an example of the display generated by the SHow -SYS STATistics -SHDlc command:

```
ACCUMULATED VALUES
== SHDlc statistics ================== 3=======  3c=======
Frames:
Received                               118767    25688
Transmitted                            23758     128370
Bytes:
Received                               48827832  51376
(cont'd)
ACCUMULATED VALUES(cont'd)
== SHDlc statistics ================== 3=======  3c=======
Transmitted                            47516     13911937
Frames Discarded:
Received                               16        44
Transmitted                            0         5
Circuit Count:
Connected                              61        2
Disconnected                           0         0
```

The elements of the display are described as follows. Statistics correspond to the port numbers (3 and 3c) at the column heads.

Frames

Received Number of SDLC or HDLC frames received from the WAN by DLSw.
Transmitted Number of SDLC or HDLC frames transmitted to the WAN by DLSw.

Bytes

Received Number of bytes received from the WAN by DLSw.
Transmitted Number of bytes transmitted to the WAN by DLSw.

Frames Discarded

Received Number of SDLC or HDLC frames received from the WAN by DLSw that were discarded.
Transmitted Number of SDLC or HDLC frames transmitted to the WAN by DLSw that were discarded

Circuit Count

Connected Number of times the circuit was in connection state.
Disconnected Number of times the circuit was not in connected state.

SMDS Service

The following is an example of the display generated by the SHow -SYS STATistics -SMDS command:

```
ACCUMULATED VALUES
== SMDS statistics ===========  1======= 3======= 5======= 7=======
Packets Received:
   Individual Address           -        0        -        -
   Group Address                -        0        -        -
Packets Transmitted:
   Individual Address           -        0        -        -
   Group Address                -        0        -        -
Error Packets Received:
   Unrecognized IA              -        0        -        -
   Unrecognized GA              -        0        -        -
   Invalid Address Type         -        0        -        -
   Syntactic errors             -        0        -        -
```

The elements of this display are described as follows:

Packets Received

Individual Address Number of individually addressed SIP Level 3 PDUs received.

Group Address Number of group addressed SIP Level 3 PDUs received.

Packets Transmitted

Individual Address Number of individually addressed SIP Level 3 PDUs that have been sent out.

Group Address Number of group addressed SIP Level 3 PDUs that have been sent out.

Error Packets Received

Unrecognized IA Number of SIP Level 3 PDUs received with invalid or unknown individual destination.

Unrecognized GA Number of SIP Level 3 PDUs received with invalid or unknown group addresses.

Invalid Address Type Number of SIP Level 3 PDUs received that had the source or destination address_type fields (the four most significant bits of the address) not equal to the value 0xC or 0xE, or the value is equal to 0xE for the source address.

Syntactic errors Number of SIP Level 3 PDUs received that have errors, including protocol processing and bit errors, but excluding addressing related errors. For more information, refer to RFC 1304.

SNMP Service

The following is an example of the display generated by the SHow -SYS STATistics -SNMP command:

```
ACCUMULATED VALUES
======================== SNMP statistics =======================
Incoming SNMP PDUs              39
   Get Requests                 4
   Get-Next Requests            30
   Set Requests                 0
   Bad PDUs                     5
Outgoing SNMP PDUs              34
   Get Responses                33
   Error Responses              1
   Traps                        0
```

The elements of this display are described as follows:

Incoming SNMP PDUs

Number of PDUs delivered to SNMP.

Get Requests Number of Get-Request PDUs processed by SNMP.

Get-Next Requests Number of Get-Next Request PDUs processed by SNMP.

Set Requests Number of Set-Request PDUs processed by SNMP.

Bad PDUs Number of PDUs delivered to but not processed by SNMP, including unsupported version, unknown community name, not allowed operation by the named community, ASN.1 parsing errors, and unknown PDU type.

Outgoing SNMP PDUs Number of PDUs generated by SNMP.

Get Responses Number of Get-Response PDUs generated by SNMP.
Error Responses Number of valid SNMP PDUs generated by SNMP and for which the value of the error-status is not noError (0), including tooBig (1), noSuchName (2), badValue (3), readOnly (4), and genErr (5).
Traps Number of Trap PDUs generated by SNMP.

SR Service

The following is an example of the display generated by the **s**How -SYS STATistics -SR command:

```
ACCUMULATED VALUES
== SR statistics =============  1======= 3======= 5======= 7=======
RECEIVED:
   All Route Explorer:          0        0        0        0
   Spanning Tree Explorer:      0        0        0        0
   Specifically Routed:         0        0        0        0
   SRT Gateway Packets          0        0        0        0
TRANSMITTED:
   All Route Explorer:          0        0        0        0
   Spanning Tree Explorer:      0        0        0        0
   Specifically Routed:         0        0        0        0
   SRT Gateway Packets          0        0        0        0
ERRORS:
   Bad Routing Info:            0        0        0        0
   Expl RD Limit Exceeded:      0        0        0        0
    "    Frames Too Long:       0        0        0        0
    "    Incorrect Ring In:     0        0        0        0
   SRF Duplicate Ring In:       0        0        0        0
    "  Missing Ring In:         0        0        0        0
    "  Bad Bridge Number:       0        0        0        0
    "  Bad Ring Out:            0        0        0        0
   Discarded SRTG Pkts:         0        0        0        0
   Unknown SRTG Pkts:           0        0        0        0
```

The elements of this display are described as follows:

RECEIVED

All Route Explorer: Number of All Route Explorer frames received.
Spanning Tree Explorer: Number of Spanning Tree Explorer frames received.
Specifically Routed: Number of Specifically Routed frames received.
SRT Gateway Packets: Number or SRT gateway packets received.

TRANSMITTED

All Route Explorer: Number of All Route Explorer frames transmitted.
Spanning Tree Explorer: Number of Spanning Tree Explorer frames transmitted.
Specifically Routed: Number of Specifically Routed frames transmitted.
SRT Gateway Packets: Number of SRT gateway packets transmitted.

ERRORS

Bad Routing Info: Number of frames discarded because of a formatting error in the Routing Information field, for example, bad Largest Frame Size (LFS) or Direction bit (D) set in frame.

Expl RD Limit Exceeded: Number of explorer frames discarded because of the Routing Designator (RD) limit exceeded.

Expl Frames Too Long Number of frames discarded because the size exceeds the largest frame size configured for the inbound interface.

Expl Incorrect Ring In: Number of explorer frames discarded because the last LAN ID of the RI does not equal the LAN-In ID.

SRF Duplicate Ring In: Number of Specifically Routed frames discarded because the LAN-In ID already existed in the RI.

SRF Missing Ring In: Number of Specifically Routed frames discarded because the LAN-In ID is not found in the RI field.

SRF Bad Bridge Number: Number of Specifically Routed frames discarded because the bridge number following the LAN-In ID does not match the bridge number of this bridge.

SRF Bad Ring Out: Number of Specifically Routed frames discarded because a matching LAN-Out ID (that is, the LAN ID that follows the bridge number of the local bridge) is not configured.

Discarded SRTG Pkts: Number of SRTG packets discarded due to incorrect packet formats.

Unknown SRTG Pkts: Number of packets discarded because their protocols are not supported.

STP Service

The following is an example of the display generated by the SHow -SYS STATistics -STP command:

```
ACCUMULATED VALUES
== STP statistics ============ 1======= 3======   5======= 7=======
STP statistics
Forwarding State Count         1        1         0        0
Blocking State Count           0        1         0        0
Bad Config BPDU received       12       1         0        0
Looped Config BPDU receiv      0        0         0        0
Message Age Timeouts           0        1         0        0
```

The elements of this display are described as follows:

STP statistics

Forwarding State Count Number of times the specified port was put in forwarding state.

Blocking State Count Number of times the specified port was put in blocking state because of a possible loop.

Bad Config BPDU received Number of bridge protocol data units (BPDUs) received with information indicating that the transmitting bridge does not recognize another high-priority bridge on the network (possibly because the transmitting bridge just booted up).

Looped Config BPDU received — Number of BPDUs the bridge received its own.

Message Age Timeouts — Number of times a neighboring high-priority bridge has gone out of service.

SYS Service

The following is an example of the display generated by the SHow -SYS DpmSTATistics POrt IP command:

```
DpmSTATistics --- From Source, Per Port, Protocol:IP
Destination ===>
Source
===       1 ======  1B ====== 1C ====== 1D ====== 1E ====== 1F ===
1         0         0         0         592137    591940    591873
1B        0         0         0         592126    591833    591943
1C        0         0         0         592164    591735    591988
1D        0         0         0         0         0         0
1E        0         0         0         0         0         0
1F        0         0         0         0         0         0

===       4 ======  5 ======= 7 ======= CEC ===
1         0         0         0         0
1B        0         0         0         0
1C        0         0         0         0
1D        0         0         0         11
1E        0         0         0         11
1F        0         0         0         11
```

The elements of the display are described as follows:

Port Per port statistics are displayed.

Source Data is displayed about packets transmitted from the specified slots or ports, for example, ports that are the source of the packets.

If the value is all zeroes for a particular source port, that port is not displayed.

Protocol Statistics for the specified protocol, IP, are displayed. In the screen example, 592164 IP packets were forwarded from port 1C to port 1D.

TCP Service

The following is an example of the display generated by the SHow -SYS STATistics -TCP command:

```
ACCUMULATED VALUES
======================== TCP statistics =========================
TCP Packets:     Transmitted   Received      Retransmitted
                 728           245           0
TCP Connections: Initiated     Accepted      Failed           Reset
                 0             2             0                0
```

The elements of this display are described as follows:

TCP Packets

Transmitted	Number of TCP packets transmitted within a specified interval.
Received	Number of TCP packets received within a specified interval.
Retransmitted	Number of TCP packets retransmitted within a specified interval. Packets are retransmitted when the previous attempt fails or when they are timed out.

TCP Connections

Initiated	Number of TCP connections attempted.
Accepted	Number of successful TCP connection attempts.
Failed	Number of failed TCP connection attempts.
Reset	Number of TCP connections reset. Connections are reset if they were aborted because of error.

UDP Service

The following is an example of the display generated by the SHow -SYS STATistics -UDP command.

The UDP Service does not appear in the user interface. However, you can still obtain UDP statistics.

```
ACCUMULATED VALUES
======================== UDP statistics =========================
UDP Statistics:
   Datagrams transmitted                   323
   Good datagrams received                 83707
   Datagrams with errors                   0
   Datagrams with unknown port             6
   ICMP Datagrams received                 0
```

The elements of this display are described as follows:

UDP Statistics

Datagrams transmitted	Number of packets sent to other networks.
Good datagrams received	Number of error-free packets received.
Datagrams with errors	Number of packets containing errors, such as checksum errors unable to give packet to UDP user.
Datagrams with unknown port	Number of packets received for which UDP does not have matching port number in its port table.
ICMP Datagrams received	Number of Internet Control Management Protocol packets received.

UDPHELP Service

The following is an example of the display generated by the SHow -SYS STATistics -UDPHELP command.

The system displays the UDPHELP statistics on a system-wide basis, rather than on a port-by-port basis.

```
ACCUMULATED VALUES
====================== UDPHELP statistics ======================
BOOTP/UDP/IP Broadcast Helper Statistics
   Packets Received           0
   Packets Forwarded          0
   Packets ReBroadcasted      0
   Packets Discarded          0
   Miscellaneous Error        0
```

The elements of this display are described as follows:

BOOTP/UDP/IP Broadcast Helper Statistics

Packets Received	Number of packets received.
Packets Forwarded	Number of packets forwarded to servers.
Packets ReBroadcasted	Number of packets forwarded to all interfaces.
Packets Discarded	Number of packets discarded because the packet did not broadcast a destination address.
Miscellaneous Error	Number of times a packet could not be forwarded because of a combination of heavy traffic on the network and memory constraints.

VIP Service

The following is an example of the display generated by the SHow -SYS STATistics -VIP command:

```
ACCUMULATED VALUES
== VIP statistics ======= 1=======  3=======  5=======  7=======
VINES IP Statistics:
Received                  0         0         0         0
Xmitted                   0         0         0         0
Forwarded                 0         0         0         0
To Client                 0         0         0         0
From Client               0         0         0         0
Discarded                 0         0         0         0

VINES ARP Statistics:
Query Request(In)         0         0         0         0
Query Response(Out)       0         0         0         0
Assignment Req(In)        0         0         0         0
Assignment Resp(Out)      0         0         0         0
ARP Discarded             0         0         0         0

VINES ICP Statistics:
Exception                 0         0         0         0
Metric                    0         0         0         0

VINES RTP Statistics:
Updates(In)               0         0         0         0
Updates(Out)              0         0         0         0
Requests(In)              0         0         0         0
Responses(Out)            0         0         0         0
Redirects(In)             0         0         0         0
Redirects(Out)            0         0         0         0
RTP Discarded(In)         0         0         0         0
Xmit Fail                 0         0         0         0
```

The elements of this display are described as follows:

VINES IP Statistics

Received Number of packets received on a port since boot-up time or last flushing. This number is the total number of packets received from the network including Forwarded packets, Broadcast and Unicast packets addressed to the router and successfully delivered to VIP clients, and some of Discarded packets.

Xmitted Number of packets generated and transmitted by the router since boot-up time or last flushing. There can be only three types of packets (ARP, ICP, and RTP) generated by the router.

Forwarded Number of packets routed successfully to other ports since boot-up time or last flushing. Those packets generated by the router itself are not included in this category.

To client Number of broadcast packets or unicast packets addressed to the router and successfully delivered to proper VIP clients. ARP and ICP packets are not included in this count. So this number is the total number of RTP packets received on the port since boot-up time or last flushing.

From client Number of broadcast packets or unicast packets received from the clients. On the router reside only VINES ARP, ICP, and RTP clients, but ARP and ICP packets are not included in this count; this number is the total number of RTP packets received on the port since boot-up time or last flushing.

Discarded Number of packets discarded by VIP due to various errors such as bad framed packets, packets without any data, packets destined to other networks when VIP routing is turned off, packets addressed to the router but no clients available, etc. This number includes bad packets received from either networks or VIP clients.

VINES ARP Statistics

Query Request(In) Number of ARP Query Requests received.

Query Response(Out) Number of ARP Query Responses generated.

Assignment Req(In) Number of incoming ARP Assignment Requests.

Assignment Resp(Out) Number of Assignment Responses generated.

ARP Discarded Number of ARP packets discarded due to bad checksum, subnetwork number exhaustion, or various transmit failures.

VINES ICP Statistics

Exception Number of exception notifications generated whenever bad packets were detected, but those packets have an error notify bit set by the source node.

Metric Number of metric notifications generated whenever incoming packets have the metric bit set by the source node.

VINES RTP Statistics

Updates(In) Number of RTP broadcasts received on a port by the router.

Updates(Out) Number of RTP broadcasts transmitted by the router since the boot time or last flushing.

Requests(In) Number of RTP requests received by the router.

Replies(Out) Number of RTP responses generated by the router in response to RTP requests. The number of RTP responses can be bigger than the number of RTP requests depending on the current number of RTP table entries.

Redirects(In) Number of RTP Redirect packets received.

Redirects(Out) Number of RTP Redirects generated.

RTP Discarded(In) Number of RTP packets dropped by the router due to various errors such as packets received from unknown networks, lost packets, etc.

Xmit Failure(Out) Number of instances where the router failed to generate RTP packets because of resource depletion.

WE Service

The following is an example of the display generated by the SHow -SYS STATistics -WE command:

```
== WE statistics =======   2=======   3=======   6=======   8=======
Control Frames Sent        -          8199       -          -
Control Bytes Sent         -          106587     -          -
Control Frames Recv        -          8292       -          -
Control Bytes Recv         -          954097     -          -
LMI Frames Sent            -          8199       -          -
LMI Frames Recv            -          8199       -          -
Local Link Down Events     -          0          -          -

Received Frame Errors:
Invalid DLCI Frames        -          0          -          -
Invalid Control Frames     -          0          -          -
Inactive DLCI Discards     -          4          -          -
Small Frames Recv          -          0          -          -
```

The elements of this display are described as follows:

WE Statistics

Control Frames Sent Specifies how many control frames were sent, using the WNI protocol, between the NETBuilder II bridge/router and the WAN Extender. Control frame counts include the LMI frame counts.

Control Bytes Sent Specifies how many control bytes were sent, using the WNI protocol, between the NETBuilder II bridge/router and the WAN Extender.

Control Frames Recv Specifies how many control frames were received, using the WNI protocol, between the NETBuilder II bridge/router and the WAN Extender. Control frame counts include the LMI frame counts.

Control Bytes Received Specifies how many control bytes were received, using the WNI protocol, between the NETBuilder II bridge/router and the WAN Extender.

LMI Frames Sent Specifies how many LMI frames were sent, using the WNI protocol, between the NETBuilder II bridge/router and the WAN Extender.

LMI Frames Recv	Specifies how many LMI frames were received, using the WNI protocol, between the NETBuilder II bridge/router and the WAN Extender.
Local Link Down Events	Specifies the number of local link down events. These events occur when a number of consecutively unanswered Status Enquiry messages exceeds the ports configured ErrorThreshold. Refer to the ErrorThreshold and KeepAliveInt parameters in Chapter 64 in *Reference for NETBuilder Family Software* for more information about Status Enquiry messages.

Received Frame Errors

Invalid DLCI Frames	Specifies how many DLCI frames sent between the NETBuilder II bridge/router and the WAN Extender were invalid.
Invalid Control Frames	Specifies how many control frames sent between the NETBuilder II bridge/router and the WAN Extender were invalid.
Inactive DLCI Discards	Specifies how many discarded data packets (because of a terminated connection) were encountered in transit after termination.
Small Frames Recv	Specifies how many small frames were received between the NETBuilder II bridge/router and the WAN Extender.

X25 Service

The following is an example of the display generated by the SHow [!<port>] -X25 X25STATistics command:

```
Call Request             0        Incoming Call            0
Call Accepted            0        Call Connected           0
Clear Request            0        Clear Indication         0
DTE Clear Conf           0        DCE CLear Conf           0
DTE data                 0        DCE Data                 0
DTE Interrupt            0        DCE Interrupt            0
DTE Interrupt Conf       0        DCE Interrupt Conf       0
DTE RR                   0        DCE RR                   0
DTE RNR                  0        DCE RNR                  0
Reset Request            0        Reset Indication         0
DTE Reset Conf           0        DCE Reset Conf           0
Restart Request          0        Restart Indication       0
DTE Restart Conf         0        DCE Restart Conf         0
DTE Invalid Pkt          0        DCE Invalid Pkt          0
Link Down                0        Link Up                  0
```

The elements of this display are described as follows:

Call Request	Number of Call Request packets sent.
Call Accepted	Number of Calls Accepted packets sent.
Clear Request	Number of Clear Request packets sent.
DTE Clear Conf	Number of Clear Confirmation packets sent.
DTE data	Number of Data packets sent.
DTE Interrupt	Number of Interrupt Request packets sent.

DTE Interrupt Conf	Number of Interrupt Confirmation packets sent.
DTE RR	Number of RR packets sent.
DTE RNR	Number of RNR packets sent.
Reset Request	Number of Reset Request packets sent.
DTE Reset Conf	Number of Reset Confirmation packets sent.
Restart Request	Number of Restart Request packets sent.
DTE Restart Conf	Number of Restart Confirmation packets sent.
DTE Invalid Pkt	Number of Invalid Packets received from clients.
Link Down	Number of times Frame Layer went down.
Incoming Call	Number of Incoming calls received.
Call Connected	Number of Call Connected packets received.
Clear Indication	Number of Clear Indication packets received.
DCE CLear Conf	Number of Clear Confirmation packets received.
DCE Data	Number of Data packets received.
DCE Interrupt	Number of Interrupt packets received.
DCE Interrupt Conf	Number of Interrupt Confirmation packets received.
DCE RR	Number of RRs received.
DCE RNR	Number of RNRs received.
Reset Indication	Number of Reset Indication packets received.
DCE Reset Conf	Number of Reset Confirmation packets received.
Restart Indication	Number of Restart Indication packets received.
DCE Restart Conf	Number of Restart Confirmation packets received.
DCE Invalid Pkt	Number of Unrecognized packets received.
Link Up	Number of times Frame Layer came up.

I

STATIC TABLES

The number of entries you can have in statically configured routing tables depends on the NETBuilder hardware on which you are running your bridge/router software.

Table I-1 lists the different bridge/router features and router types, and the maximum number of routing table entries you can have for each hardware platform.

Table I-1 Number of Entries Allowed in Static Tables

Bridge/Router Feature	NB II (all chassis)	SS II 222, 227, and 228	SS II 422 and 427
Bridge			
Bridge Table	2048	512	512
AppleTalk			
Address Mapping Table	1000	1000	1000
Banyan VINES			
WAN Neighbor Table	128	64	64
DECnet			
WAN Neighbor Table	32/WAN port	32/WAN port	32/WAN port
IP			
Static Routing Table	256	No limit	No limit
Static Address Table	256	No limit	No limit
Secondary IP Addresses	32	No limit	No limit
IP-OSPF			
ExteriorPolicy	64	No limit	No limit
InteriorPolicy	64	No limit	No limit
StaticPolicy	64	No limit	No limit
ReceivePolicy	64	No limit	No limit
Neighbors	16/port	No limit	No limit
Virtual Link	8/port	No limit	No limit
OSI			
End System Table	64	64	64
PrefixRoutes	64	64	64
Neighbors	28	28	28

J

AUDIT TRAIL MESSAGES

Table J-1describes bridge/router audit trail messages identified by record type and status codes. For example, a message that indicates a macro cache overflow includes the record type code MO.

Table J-1 Audit Trail Messages

Service	Record Type Code	Status Code	Description
All services	CC		The user has configured the network. Each CC record type is followed by an additional explanatory code, which indicates what commands or parameters have been used. The record also usually displays changed values.
ARP	AC		Duplicate Internet address detected.
BRidge		FW	The CONTrol parameter has been set to FOrward or NoFOrward as indicated in the record.
		LE	The CONTrol parameter has been set to LEarn or NoLEarn as indicated in the record.
		SRS FWD	The SRcSecurity parameter has been set to Fwd for the specified port.
		SRS BLK	The SRcSecurity parameter has been set to Blk for the specified port.
		SRS NONE	The SRcSecurity parameter has been set to None for the specified port.
		DSS FWD	The DStSecurity parameter has been set to Fwd for the specified port.
		DSS BLK	The DStSecurity parameter has been set to Blk for the specified port.
		DSS NONE	The DStSecurity parameter has been set to None for the specified port.
IP	IR		An ICMP message from the second device has been received by the first device indicated on the audit trail. The two numbers that follow are the type field and code field of the ICMP message. Type 3 messages (known as destination unreachable) contain additional information from the returned (erroneous) IP header. For more information, refer to RFC 792.
	IX		An ICMP message has been transmitted from the first device to the second device indicated on the audit trail. The two numbers that follow are the type field and code field of this ICMP message.
Macro	MO		Macro cache overflow. Macro service may be disrupted for some users.
	MI		Message generated by a macro currently in execution. AUDIT <message> is inside a macro. <Message> can be anything.
PATH	CC	PA ON	The CONTrol parameter has been used to enable the specified path.
		PA OFF	The CONTrol parameter has been used to disable the specified path.
		BA	The BAud parameter has been used to set the baud rate for the specified path to the value indicated in the record.
	NU		Indicates the average per-minute utilization of network capacity (on a scale of 1,000). This record is generated once every 10 minutes. Each record consists of 10 entries, one for each minute. To obtain the percentage of network capacity represented by an entry, divide the reported number by 10.
	PU		The specified path is operating.

(continued)

Table J-1 Audit Trail Messages (continued)

Service	Record Type Code	Status Code	Description
PATH	SUX		Records the average per-minute outgoing traffic as a fraction of the specified serial line capacity (on a scale of 1,000). This record is generated once every 10 minutes. Each record consists of 10 entries, one for each minute. To obtain the percentage of outgoing capacity represented by an entry, divide the reported number by 10.
	SUR		Records the average per-minute incoming traffic as a percentage of the specified serial line capacity (on a scale of 1,000). This record is generated once every 10 minutes. Each record consists of 10 entries, one for each minute. To obtain the percentage of incoming capacity represented by an entry, divide the reported number by 10.
PORT	CC	PORT ON	The specified port has been enabled.
	CC	PORT OFF	The specified port has been disabled.
	CC	ADD PA	Paths have been added to the specified port.
	CC	DEL PA	Paths have been deleted from the specified port.
PPP	AC		Duplicate Internet address detected.
Serial Line Protocol	PD	NOACK	The specified path is not operating because acknowledgments for probe packets are not being received.
SLIP	AC		Duplicate Internet address detected.
STP	BC		The specified bridge has been booted.
	CC	STPPRI	The PortPriority parameter has been set to the value indicated.
	CC	STPPRI	The BridgePriority parameter has been set to the value indicated.
	CC	STPCOST	The PathCost parameter has been set to the value indicated.
	BK		The specified port is in blocking mode.
	FW		The specified port is in forwarding mode.
	CC	STSTP ON	The Spanning Tree Protocol has been turned on.
	CC	STSTP OFF	The Spanning Tree Protocol has been turned off.
	CC	STHR ON	HopReduce has been selected for the Spanning Tree Protocol.
	CC	STHR OFF	NoHopReduce has been selected for the Spanning Tree Protocol.
SYS	None	None	<title of statistics> exceeded per minute threshold <threshold> <title of statistics> exceeded per hour threshold <threshold> <title of statistics> exceeded per day threshold <threshold> <title of statistics> exceeded accumulation threshold These messages indicate that normal levels of network activity have been exceeded for a specific statistic. No record type code or status code appears in the message.
	DLT	SN	A data link test was initiated from the local bridge to a wide area bridge. The record indicates whether the test was run at the maximum transmission rate, which is the default transmission rate, or at a user-defined transmission rate. The record also contains the address of the wide area bridge.
		ST	A data link test was initiated from a wide area bridge to a local bridge. The record also contains the address of the wide area bridge.
		DONE	A data link test has been completed.
	PE		Five unsuccessful attempts to enter a password on the internetwork bridge have been made during the five minutes before the message was recorded.

K

REGULAR EXPRESSIONS

This appendix describes the regular expressions used for creating and displaying AS-path-based filters (AsPolicyAll, AsPolicyExt, AsPolicyInt, AsPolicyPeer, and DisplayFilter parameters) in the BGP Service and for altering SHow and SHowDefault displays with the GREP command.

You can use regular expressions to specify a general string. This general string can then be used for pattern matching.

A regular expression is a formula for generating a set of strings. If a particular string can be generated by a given regular expression, that string and regular expression match. In many ways, a regular expression is a program, and the regular expression matches the strings the program generates.

A regular expression consists of different components described in Table K-1, each of which is used to build the regular expression string-generating program.

Table K-1 Regular Expression Components

Regular Expression	Function
c	Use an ordinary ASCII character (excluding the special characters) to match that character. For example, c matches a lowercase c.
.	Use a period (.) to match any character except NEWLINE.
*	Use a regular expression followed by an asterisk (*) to match zero or more occurrences of the one-character regular expression. If there is any choice, the longest leftmost string that permits a match is chosen.
+	Use a regular expression followed by plus sign (+) to match one or more occurrences of the one-character regular expression. If there is any choice, the longest leftmost string that permits a match is chosen.
?	Use a regular expression followed by a question mark (?) to match zero or one occurrences of the one-character regular expression. If there is any choice, the longest leftmost string that permits a match is chosen.
|	Use two regular expressions separated by vertical bar (|) or NEWLINE to match either the first or the second (logical OR operation).
( )	Use a regular expression enclosed in parentheses to match the regular expression.
\. * \[\\	Use a backslash (\) followed by any special character (period, asterisk, left square bracket, backslash) to match the special character. These characters are special except when they are enclosed within square brackets ([ ]). For example, * matches an asterisk (*).

(continued)

Table K-1 Regular Expression Components (continued)

Regular Expression	Function
[string]	Use a non-empty string of characters enclosed in square brackets to match any one character in that string. If the first character of the string is a caret (^), the one-character regular expression matches any character except NEWLINE and the remaining characters in the string. The caret has this special meaning only if it occurs first in the string. The hyphen (-) indicates a range of consecutive ASCII characters; for example, [0-9] is equivalent to [0123456789]. The hyphen loses this special meaning if it occurs first (after an initial ^, if any) or last in the string. The right square bracket (]) does not terminate a string when it is the first character within it (after an initial ^, if any); for example, [ ]a-f] matches either a right square bracket or one of the letters a through f inclusive. The period, asterisk, left square bracket, and backslash represent themselves within such a string of characters.
Concatenation: The remaining regular expressions are for concatenation, a regular expression that matches the concatenation of the strings matched by each component of the regular expression.	
\<	Use the sequence \< in a regular expression to constrain the one-character regular expression immediately following it only to match something at the beginning of a "word;" for example, either at the beginning of a line, or just before a letter, digit, or underline and after a character not one of these.
\>	Use the sequence \> in a regular expression to constrain the one-character regular expression immediately following it only to match something at the end of a "word;" for example, either at the end of a line, or just before a character, which is neither a letter, digit, nor underline.
\(and \)	Use a regular expression enclosed between the character sequences \(and \) to match whatever the unadorned regular expression matches.
\{m\} \{m,\} \{m,n\}	Use a regular expression followed by \{m\}, \{m,\}, or \{m,n\} to match a range of occurrences of the regular expression. The values of m and n must be non-negative integers less than 256. \{m\} matches exactly m occurrences. \{m,\} matches at least m occurrences. \{m,n\} matches any number of occurrences between m and n inclusive. Whenever a choice exists, the regular expression matches as many occurrences as possible.
\n	Use the expression \n to match the same string of characters that was matched by an expression enclosed between \(and \) earlier in the same regular expression. n is a digit; the subexpression specified begins with the nth occurrence of \ (counting from the left). For example, the expression ^\(.*\)\1$ matches a line consisting of two repeated appearances of the same string.
^	Use the caret (^) at the beginning of a regular expression to constrain the regular expression to match an initial segment of a line.

(continued)

Table K-1 Regular Expression Components (continued)

Regular Expression	Function
$	Use the dollar sign ($) at the end of a regular expression to constrain the regular expression to match a final segment of a line. For example, ^entire regular expression $ constrains the regular expression to match the entire line.

AS Filter Examples

This section provides examples of autonomous system (AS) filters using regular expressions. The following syntax is used.

```
ADD -BGP AsFilter <AsfilterID> "<regular expression>"
```

Blank spaces are represented here as underscores (_). When two spaces are shown together, a space has been inserted between the underscores, for example _ _. You must enter a blank space for each underscore shown in the examples.

Example 1 To create filter 1 that identifies an AS-path attribute containing AS25, enter:

```
ADD -BGP ASFilter 1 "_25_"
```

Example 2 To create filter 2 that identifies an AS-path attribute containing AS35 and AS50 (in this order), enter:

```
ADD -BGP ASFilter 2 "_35_.*_50_"
```

The ".*" indicates a single character followed by any number of unspecified characters.

Example 3 To create filter 3 that identifies an AS-path attribute containing AS35 and AS50 (in any order), enter:

```
ADD -BGP ASFilter 3 "_35_.*_50_|_50_.*_35_"
```

The "|" indicates a logical OR operation.

Example 4 To create filter 4 that identifies an AS-path attribute containing the AS sequence <AS5, AS46, AS32>, enter:

```
ADD -BGP ASFilter 4 "<_5_ _46_ _32_>"
```

Example 5 To create filter 5 that identifies an AS-path attribute containing the AS set [AS5, AS46, AS32], enter:

```
ADD -BGP ASFilter 5 "[_5_ _32_ _46_]"
```

AS sets are always sorted from lowest AS to highest AS.

GREP Command Examples

This section provides examples of commands used with the GREP filter and regular expressions. The following syntax is used:

```
<COMMAND> <parameters> [<options>] | GREP [-v] [-i] <grep pattern>
```

You can use the GREP command with the SHow and SHowDefault commands to alter the output to display the information you specify in the GREP pattern.

You cannot apply multiple GREP commands to a single UI command. The following command to show all IP routes that have the number 192 and 128 is not supported:

```
SHow -IP AllRoutes | GREP 192 | GREP 128
```

Example 1 To display all IP routes, you normally enter the SHow command. To show only those routes that have the number 152 in them, you can pipe the SHow or SHowDefault output to the GREP command by entering:

`SHow -IP AllRoutes | GREP 152`

The following display appears:

```
129.213.152.0  255.255.252.0  129.213.200.109  1  UP     RIP
```

The number 152 in regular expression form is a string of ASCII characters that are matched, generating the information you specified.

Example 2 To display all IP routes that do not have the number 152 in them, pipe the output to the GREP command by entering:

`SHow -IP AllRoutes | GREP -V 152`

The following display appears:

```
---------------------------- IP Routing Table --------------------------
Total Routes = 12, Total Direct Networks = 1
Destination     Mask           Gateway          Metric  Status  TTL  Source

0.0.0.0         0.0.0.0        129.213.200.109  2       UP      170  RIP
                               129.213.200.103  3       UP      150  RIP
129.213.16.0    255.255.252.0  129.213.200.109  1       UP      170  RIP
                               129.213.200.108  1       UP      170  RIP
129.213.32.0    255.255.252.0  129.213.200.109  1       UP      170  RIP
129.213.48.0    255.255.252.0  129.213.200.109  1       UP      170  RIP
                               129.213.200.102  1       UP      170  RIP
129.213.72.0    255.255.252.0  129.213.200.109  1       UP           RIP
129.213.96.0    255.255.252.0  129.213.200.103  1       UP           RIP
129.213.200.0   255.255.252.0  129.213.203.16   0       UP      --   Connected
129.213.240.0   255.255.252.0  129.213.200.109  1       UP           RIP
```

The -v option for GREP performs the NOT (or invert) operation on the display information and lists all the IP routes that do not have the number 152 in them. For descriptions of the GREP options, refer to the GREP command description in Chapter 1 in *Reference for NETBuilder Family Software*

Example 3 To display all IP routes containing the number 72 or 96 in them, pipe the output to the GREP command by entering:

`How -IP AllRoutes | GREP 72 | 96`

The following display appears:

```
129.213.152.0  255.255.252.0  129.213.200.109  1  UP     RIP
```

The first "|" represents the pipe to the GREP command. The "|" between the numbers is the regular expression for logical OR. In this example, the output matches any string that contains the numbers 72 or 96.

L

X.3 PARAMETERS AND PAD PROFILES

This appendix provides the X.3-to-TERM Service session parameter mappings as well as the 3Com implementation of the Consultive Committee for International Telegraph and Telephone (CCITT) Simple Standard packet assembler/disassembler (PAD) Profile Number 90 parameter settings.

X.3-to-TERM Service Parameter Equivalence

Table L-1 lists the standard X.3 profile parameters and equivalent parameters, that currently operate on a bridge/router which functions as a connection service gateway. For information on X.3 profile parameters, refer to CCITT Recommendations X.3 and X.29.

Table L-1 X.3-to-TERM Service Parameter Equivalence

PAD Parameter	X.3 Profile Parameters	TERM Service Parameters	Default Setting
Parameter 1	PAD recall using a character	ECMChar	^A
Parameter 2	Echo	ECHOData	ON
Parameter 3	Selection of data forwarding character	DataForward	CR, ESC, EDiting, Term, FormEf, COntrol
Parameter 4	Selection of idle timer delay	IdleTimer	1
Parameter 5	Ancillary device control	FlowCtrlFrom	Xon_Xoff
Parameter 6	Control of PAD service signal	None	Not applicable (cannot be configured by user)
Parameter 7	PAD on receipt of break	BReakAction	InBand
Parameter 8	Discard output	FlushVC	OFF
Parameter 9	Padding after carriage return	None	None
Parameter 10	Line folding	None	Not applicable (cannot be configured by user)
Parameter 11	Binary speed of start-stop DTE	BAud	9600
Parameter 12	Flow control of the PAD	FlowCtrlTo	Xon_Xoff
Parameter 13	Linefeed insertion after carriage return	LFInsertion	None
Parameter 14	Padding after linefeed	None	None
Parameter 15	Editing	LocalEDit	OFF
Parameter 16	Character delete	ERAse	^?
Parameter 17	Line delete	LineERase	^X
Parameter 18	Line display	ReprintLine	^R
Parameter 19	Editing PAD service signals	None	Not applicable (cannot be configured by user)
Parameter 20	Echo mask	ECHOMask	None
Parameter 21	Parity treatment	PARIty	None
Parameter 22	Page wait	None	Not applicable (cannot be configured by user)

CCITT Simple Standard PAD Profile

Table L-2 lists the default values of the CCITT Simple Standard PAD Profile. You can select these profiles to use with incoming and outgoing extended connections. With outgoing extended connections, you can alter the settings of these parameters to create customized profiles (configuration files) as described in Chapter 49.

The 3Com implementation of the CCITT Simple Standard PAD Profile is based on the CCITT Simple Standard PAD Profile Number 90, but does not exactly match the official CCITT definition. In the 3Com implementation of Profile Number 90, the value of PAD parameter number 19 has been changed to 2 and parameter number 6 has been changed to 5.

Table L-2 CCITT Simple Standard PAD Profile of CCITT PAD Profile 90

PAD Parameter Number	PAD Parameter Name	Value	Meaning
1	PAD recall using a character	1	Escape from data transfer
2	Echo	1	Local echo
3	Selection of data forwarding character	126	All characters in column 0 and 1 and character DEL
4	Selection of idle timer delay	0	No idle timer delay
5	Ancillary device control	1	Use of XON/XOFF (data transfer)
6	Control of PAD service signal	5	Pad service signals and the prompt PAD service signal are transmitted in the standard format
7	PAD on receipt of break	2	Reset
8	Discard output	0	Normal data delivery
9	Padding after carriage return	0	No padding after carriage return
10	Line folding	0	No line folding
11	Binary speed of start-stop mode DTE	14	Baud rate (9600)
12	Flow control of the PAD	1	Use of XON/XOFF for flow control
13	Linefeed insertion after carriage return	0	No linefeed insertion
14	Padding after linefeed	0	No padding after linefeed
15	Editing	0	No editing in the data transfer state
16	Character delete	127	Character DEL
17	Line delete	24	ASCII 18
18	Line display	18	ASCII 12
19	Editing PAD service signals	2	Editing PAD service signals for display terminals
20	Echo mask	0	All characters echoed
21	Parity treatment	0	None
22	Page wait	0	Disabled

M

WIDE AREA NETWORK SETUP INFORMATION

This appendix provides information to help you set up your wide area serial ports.

NETBuilder II I/O Module Placement

Do not insert a token ring I/O module into the NETBuilder II chassis directly above a HSS V.35 3-Port module with part number 06-0107-000.

CAUTION: *This module placement can cause overheating. Any other placement of the token ring I/O module and the HSS V.35 3-Port module is acceptable.*

This module placement problem does not occur with HSS V.35 3-Port module part number 06-0124-xxx.

T3 Plus Interoperability

In the following manuals for the BMX45S Bandwidth Manager from T3 Plus Network, Inc., the installation instructions incorrectly describe how to configure the BMX45S to work with a 3Com NETBuilder bridge/router:

- *BMX45 T3 Bandwidth Manager: User Manual (DOS/Windows)*, part number 010-10148-0001, Rev. E
- *BMX45 T3 Bandwidth Manager: User Manual (UNIX/SNMP NMS),* part number 010-10373-0001, Rev. B

In these manuals, Table 2-3, "Typical Strapping Option Requirements," describes how to configure BMX45S with 3Com, NCS, Cisco, and Bay Networks products. The 3Com NETBuilder description is incorrect; the transmit clock setting should be set to EXTERNAL.

HSS Port Utilization Percentage

The percentage of utilization displayed for HSS, HSSI, and HSS V.35 3-Port WAN ports is based on a full-duplex link. For example, a 64-kbps circuit can transmit and receive 64 kbps simultaneously. If this link were transmitting at 64 kbps and receiving nothing, the percentage of utilization would be 50 percent.

To display utilization information, enter:

```
SHow -SYS STATistics -PATH
```

Serial Line Connectivity

The following sections provide information about serial line connections, clocking, cables, and data rates.

External Device Connections

When configuring your NETBuilder II bridge/router for remote communications with external clocking devices, 3Com recommends that you set the -PATH

CONNector and CLock parameters to match the external devices before physically connecting the two devices.

External Device Cable Length

At serial line data rates greater than 56 kbps on RS-232 interfaces, 3Com recommends using cables less than 25 feet long. The HSS V.35 3-Port WAN interface includes an 8-foot external adapter cable. A cable attached to this adapter should be no more than 5 feet long.

Serial Line Clocking

When using a NETBuilder II bridge/router or SuperStack II NETBuilder bridge/router with a modem or channel service unit/data service unit (CSU/DSU), you must determine which piece of equipment provides the clocking signal. For a bridge/router using an external device over dial-up lines, the external device must provide clocking. For a bridge/router using an external device over a leased line, either device can provide clocking.

If you configure an external device to provide clocking to a bridge/router, the external device must use the return clock provided by the bridge/router. Neither the external device nor the cable used to connect it to the bridge/router should loop back the transmit clock to the receive clock.

If you connect two SuperStack II NETBuilder bridge/routers, or a SuperStack II NETBuilder bridge/router to a NETBuilder II bridge/router with an HSS 3-Port WAN interface, you must use a modem eliminator and set the -PATH CLock parameter to External on both devices. Contact your 3Com supplier for a suggested list of modem eliminators.

Serial Line Supported Data Rates

Version 8.3 software supports serial line data rates above 2,048 kbps for external clocking devices only. If the selected data rate is above 2,048 kbps with internal clocking (Test Mode) specified, the actual data rate used by the bridge/router is 2,048 kbps.

When using external clocking devices, set the data rate of the serial line on the bridge/router as close as possible to the external device. Packet processing is optimized to this value. Statistical reporting of line utilization is based on the data rate you configure.

N

APPN CONFIGURATION EXAMPLES

This appendix provides examples of how to configure Advanced Peer-to-Peer Networking (APPN) on the 3Com network node so that sessions to and from other commonly used IBM platforms can take place. For information on basic APPN configuration steps, refer to Chapter 10.

Unless otherwise noted, the examples in this appendix assume that the NETBuilder II bridge/router and the corresponding IBM platforms are configured for Intermediate Session Routing (ISR) only.

AS/400 Configuration

This section provides examples of how to configure the 3Com APPN network node to communicate to and from an AS/400 in both token ring and Frame Relay environments.

If you change transmission group (TG) characteristics using the LinkStaCHar parameter on the NETBuilder II bridge/router, you must also change the corresponding TG characteristics on the AS/400 to match. If you change TG characteristics on the AS/400 for a link to the NETBuilder II system, you must define a link station on the NETBuilder II system to the AS/400 and modify the TG class to match.

Example 1: Token Ring Over Physical Ports

Figure N-1 is an example of an AS/400 and a 3Com NETBuilder II system connected in a token ring environment. Table N-1 lists the parameters that must be configured on each platform if the AS/400 initiates the connection to the NETBuilder II system. Table N-2 lists the parameters that must be configured differently on each platform if the NETBuilder II system initiates the connection to the AS/400 (all other parameters are configured as shown in Table N-2).

If the AS/400 initiates the connection, the AS/400 initial connection setting must be set to DIAL, and the adjacent link station to the AS/400 does not need to be configured on the NETBuilder II system. If the NETBuilder II system initiates the session request, then the adjacent link station must be configured, and the initial connection setting on the AS/400 must be set to ANSwer.

When configuring the remote media access control (MAC) addresses, you configure MAC addresses in noncanonical format on the AS/400 and canonical format on the NETBuilder II system. In the tables, the MAC addresses are shown in the respective formats that must be used on each platform.

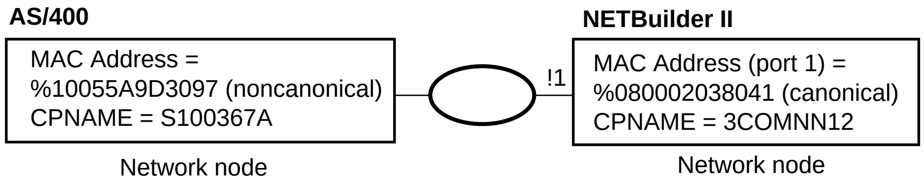


Figure N-1 Token Ring Configuration Between NETBuilder II System and AS/400

Table N-1 AS/400 Parameters to Initiate Token Ring Connection with NETBuilder II System (ISR Only)

AS/400 Parameters for Connection with 3Com Network Node	NETBuilder II Commands for Responding to Connection Requests from AS/400 Network Node
Change Line Description: Line Description: TOKENRING1* Local adapter address: 10005A9D3097	**Enable the port:** `SETDefault !1 -PORT CONTrol = Enabled` **Set APPN parameters:**
Change Controller Description: Controller Description: 3COMNN12 Option: *Basic Category of Controller: *APPC Link type: *LAN Maximum frame size: 1033‡ APPN/HPR capable: *NO Active switched line: TOKENRING1* Remote network identifier: US3COMHQ Remote control point: 3COMNN12 Exchange Identifier: E06xxxxx† Initial connection: *DIAL Dial initiation: *LINKTYPE LAN remote adapter address: 100040C00182	`SETDefault -APPN LocalNodeName = US3COMHQ.3COMNN12 xxxxx`† `SETDefault !1 -APPN PortDef = LLC2 1033`‡ `0 80 HPR=No`

* The line description is the name you assign to the line. The line description must match the name used for the active switched line parameter.
† 3Com network nodes send the exchange identifier E06xxxxx. The digits xxxxx are set in LocalNodeName, and the AS/400 exchange identifier must match. If you do not configure the ID number, it defaults to 00000.
‡ The maximum frame size setting on the AS/400 should match the maximum BTU size value on the NETBuilder II network node. However, if these values do not match, the value is negotiated and the lower value is used.

Table N-2 NETBuilder II Parameters to Initiate Token Ring Connection with AS/400 (ISR Only)

AS/400 Parameters for Responding to Connection Requests from 3Com Network Node	NETBuilder II Commands for Connection with AS/400 Network Node
Change Controller Description: Initial connection: *ANS	**Define AS/400 as an adjacent link station through token ring:** `ADD !1 -APPN AdjLinkSta NN 1033 NC10005A9D3097* 04`† `HPR=NO`

* The MAC address here matches the Local Adapter Address of the AS/400 in Table N-1. The address is entered here in noncanonical format.
† SAP 04 represents the SAP of the AS/400. Many IBM devices use SAP 04. To verify the correct SAP, consult IBM documentation.

If the AS/400 is an end node, then a different configuration is required. Table N-3 lists the different configuration necessary on both sides if the AS/400 is an end node. If the AS/400 is an end node, for example, then you do not have to configure the AS/400 as an adjacent link station on the 3Com network node because the AS/400 calls into the 3Com network node. Unless listed here, the configuration on the AS/400 is the same as in Table N-2, since the AS/400 initiates the connection with the network node.

Table N-3 Settings if AS/400 is an End Node

AS/400 Acting as End Node	NETBuilder II Commands
Change Network Attributes: Node Type = *ENDNODE Network node servers Server network ID: US3COMHQ Control point name: 3COMNN12	None

The previous examples assume that both the NETBuilder II bridge/router and the AS/400 are both ISR nodes only. Table N-4 lists how you would enter the commands differently to configure High Performance Routing (HPR) support for both nodes.

Table N-4 Settings to Configure Differently for Both Nodes to Support HPR

Parameters for AS/400	NETBuilder II Commands

Change Controller Description:
APPN/HPR capable: *YES
	`SETDefault !1 -APPN PortDef = LLC2 1033 0 80 HPR=Yes`
	`ADD !1 -APPN AdjLinkSta NN 1033 NC10005A9D3097 HPR=Yes`

Example 2: Frame Relay over Physical Ports

Figure N-2 is an example of an AS/400 and a 3Com NETBuilder II system connected in a Frame Relay environment using physical ports. Table N-5 lists the parameters that must be configured on each platform if the AS/400 initiates the connection to the NETBuilder II system. Table N-6 lists the parameters that must be configured differently on each platform if the NETBuilder II system initiates the connection to the AS/400 (all other parameters would be configured the same as shown in Table N-5).

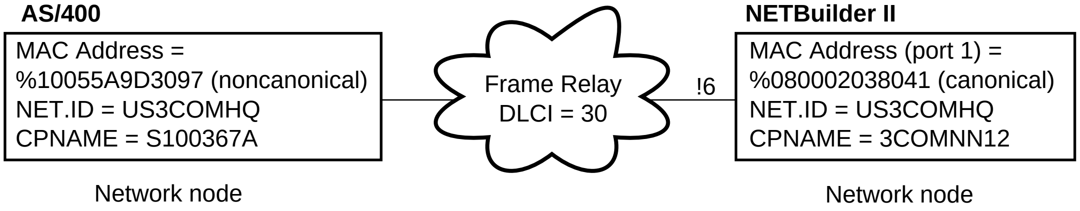


Figure N-2 Frame Relay Configuration Between NETBuilder II System and AS/400 (Physical Port)

Table N-5 AS/400 Parameters to Initiate Frame Relay Connection with NETBuilder II (Physical Ports)

AS/400 Parameters	NETBuilder II Commands
Network Interface (NWI) Description:	**Set Path parameters:**
Network interface description: FRAMERELAY	`SETDefault !6 -PATH BAud = 1536`
Category of NWI: *FR	`SETDefault !6 -PATH CONTrol = Enabled`
Option: *BASIC	**Set up port for Frame Relay:**
Line speed: 1536000	`SETDefault !6 -PORT OWNer = FrameRelay`
LMI mode: *NONE	`SETDefault !6 -FR CONTrol = NoLMI`

(continued)

Table N-5 AS/400 Parameters to Initiate Frame Relay Connection with NETBuilder II (Physical Ports) (continued)

Change Line Description:
Line Description: FR1
Attached nonswitched NWI: FRAMERELAY
DLC Identifier: 30
Exchange Identifier: 0560367A*
Maximum frame size: 1033
Change Controller Description:
Controller Description: 3COMNN12DIAL
Option: *Basic
Category of Controller: *APPC
Link type: *FR
Change Controller Description: (cont.)
Maximum frame size: 1033
Remote network identifier: US3COMHQ
Exchange Identifier: E0600000†‡
Initial connection: *DIAL
Dial initiation: *LINKTYPE
	Set data link type to be Frame Relay:
	`SETDefault !6 -APPN PortDef = FR 1033 0 80`
	Enable the port:
	`SETDefault !6 -PORT CONTrol = Enabled`
	Set APPN parameters:
	`SETDefault -APPN LocalNodeName = US3COMHQ.3COMNN12 00000`†

* This is the Exchange Identifier for the AS/400.
† The last five hex digits in these two entries must match.
‡ This is the exchange identifier for the NETBuilder II system.

Table N-6 NETBuilder II Parameters to Initiate Frame Relay Connection with AS/400 (Physical Ports)

AS/400 Parameters	NETBuilder II Commands
Change Controller Description: Initial connection: *ANS	**Define the AS/400 as an adjacent link station through Frame Relay:** `ADD !6 -APPN AdjLinkSta NN 1033 30 4 USCOMHQ.S100367A 0367A`*

* The last five hex digits must match the Exchange Identifier in the AS/400 line description.

Table N-7 lists parameters to set differently from those shown in Table N-5 if you are using a modem, switch, or modem eliminator. For this example, set the baud rate in the modem eliminator to 64000 (the actual speed depends on the modem, switch, or modem eliminator being used).

Table N-7 Parameters for Modem, Modem Eliminator, or Switch that Provides Clocking

AS/400 Parameters	NETBuilder II Commands

Change NWI Description:
Line speed: 64000*
	Set baud rate to match that of modem, modem eliminator, or switch:
	`SETDefault !6 -PATH BAud = 64`*
	Use clock from modem eliminator:
	`SETDefault !6 -PATH CLock = External`

* The line speed setting on the AS/400 should match the path baud rate setting on the NETBuilder II network node.

Example 3: Frame Relay over Virtual Ports

Figure N-3 is an example of an AS/400 and a 3Com NETBuilder II system connected in a Frame Relay environment using physical ports. Table N-8 lists the parameters that must be configured on each platform if the AS/400 initiates the connection to the NETBuilder II system. Table N-9 lists the parameters that must be configured differently if the NETBuilder II system initiates the connection to the AS/400.

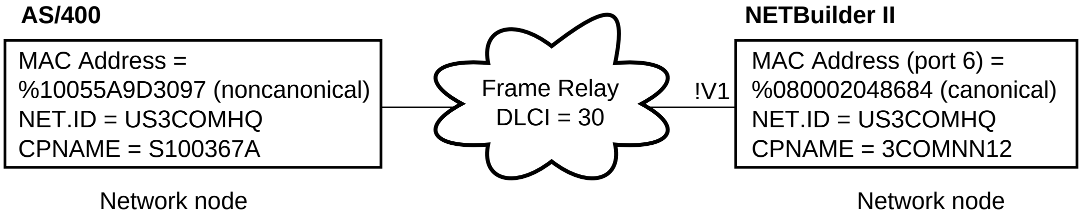


Figure N-3 Frame Relay Configuration Between NETBuilder II System and AS/400 (Virtual Port)

Table N-8 AS/400 Parameters to Initiate Frame Relay Connection with NETBuilder II (Virtual Ports)

AS/400 Parameters	NETBuilder II Commands
Network Interface (NWI) Description:	**Set Path parameters:**
Network interface description: FRAMERELAY	`SETDefault !6 -PATH BAud = 1536`
Category of NWI: *FR	`SETDefault !6 -PATH CLock = TestMode`
Option: *BASIC	**Set up port for Frame Relay:**
Line speed: 1536000	`SETDefault !6 -PORT OWNer = FrameRelay`
LMI mode: *NONE	`SETDefault !6 -FR CONTrol = NoLMI`
Change Line Description:	**Set virtual port:**
Line Description: FR1	`ADD !V1 -PORT VirtualPort 6@30`
Attached nonswitched NWI: FRAMERELAY	**Set port type to be Frame Relay:**
DLC Identifier: 30	`SETDefault !6 -APPN PortDef = FR 1033 0 80`
Exchange Identifier: 0560367A*	**Enable the port:**
Maximum frame size: 1033	`SETDefault !V1 -PORT CONTrol = Enabled`
Change Controller Description:	**Set APPN parameters:**
Controller Description: 3COMNN12DIAL	`SETDefault -APPN LocalNodeName = US3COMHQ.3COMNN12 0000`†
Option: *Basic	
Category of Controller: *APPC	
Link type: *FR	
Maximum frame size: 1033	
Remote network identifier: US3COMHQ	

* This is the Exchange Identifier for the AS/400.
† The last five hex digits in these two entries must match.

Table N-9 NETBuilder II Parameters to Initiate Frame Relay Connection with AS/400 (Virtual Ports)

AS/400 Parameters	NETBuilder II Commands
Change Controller Description: Initial connection: [*]ANS	**Define the AS/400 as an adjacent link station through Frame Relay:** `ADD !V1 -APPN AdjLinkSta NN 1033 30 4 USCOMHQ.S100367A 0367A`[*]

[*] The last five hex digits must match the Exchange Identifier in the AS/400 line description.

IBM PC Support/400 Example

This section provides examples of how to configure the 3Com APPN network node to communicate to and from PCs running PC Support/400.

Example 4: Setting Up Connections with a DOS PC

Figure N-4 is an example in which a DOS PC client is trying to access a logical unit on an AS/400 server, with the NETBuilder II system acting as the network node server for the PC. Table N-10 lists the commands you need to configure on the PC and on the NETBuilder II system for the PC to initiate the BINDs.

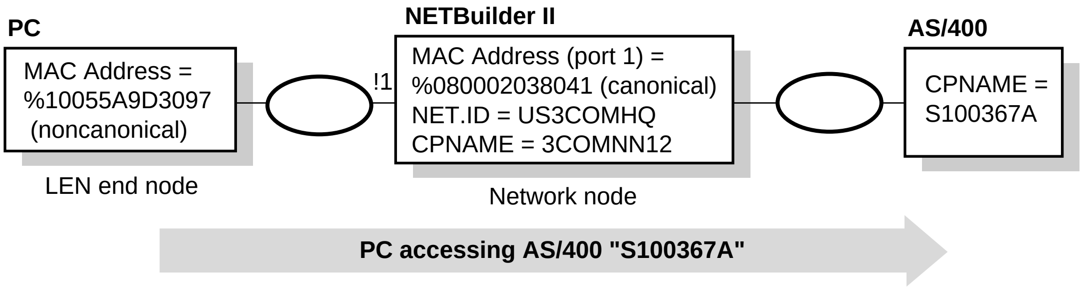

PC
MAC Address = %10055A9D3097 (noncanonical)
LEN end node
!1
NETBuilder II
MAC Address (port 1) = %080002038041 (canonical)
NET.ID = US3COMHQ
CPNAME = 3COMNN12
Network node
AS/400
CPNAME = S100367A
PC accessing AS/400 "S100367A"

Figure N-4 Token Ring Configuration Between NETBuilder II System and DOS PC

Table N-10 DOS PC Configuration to Initiate Connection with NETBuilder II System

DOS PC Commands for initiating Connection with Server via 3Com Network Node	NETBuilder II Commands for Responding to Connection Requests from DOS PC End Node
SFLR 1,I,S100367A[*]	**Set data link type for token ring:**
UPDT I:\QIWSFL2,C:\PCS,S,,,PC Support/400	`SETDefault !1 -APPN PortDef = LLC2 1033 0 80`
RTYP ITRN	**Enable the port:**
RTLN US3COMHQ.USER[†]	`SETDefault !1 -PORT CONTrol = Enabled`
TRLI S100367A, 100040C00182[‡]	**Set APPN parameters:**
ADRS PUBS[**], S100367A[††]	`SETDefault -APPN LocalNodeName = US3COMHQ.3COMNN12 0000`

[*] The value underlined on line SFLR must match the value underlined on line TRLI.
[†] US3COMHQ.USER is the LU name for PC Support/400.
[‡] This is the MAC address of the NETBuilder II system in noncanonical format.
[**] This is the name of a second AS/400.
[††] This line is used only if the NETBuilder II system is connecting to more than one AS/400.

Configuration for DLUs/DLUr

This section provides an example of the Virtual Telecommunications Access Method (VTAM) host configuration parameters and how they must match parameters on the Dependent Logical Unit Requester (DLUr) and physical unit (PU) 2x nodes.

This example is for Intermediate Session Routing only. For information on configuring HPR for VTAM, refer to the IBM document VTAM V4.3: High Performance Routing (HPR) Early User Experiences (SG24-4507-00).

Figure N-5 is an example in which a VTAM host is serving as the dependent LU server (DLUs) for a PU 2.x node with dependent LUs. The NETBuilder II bridge/router serves as the network node DLUr.

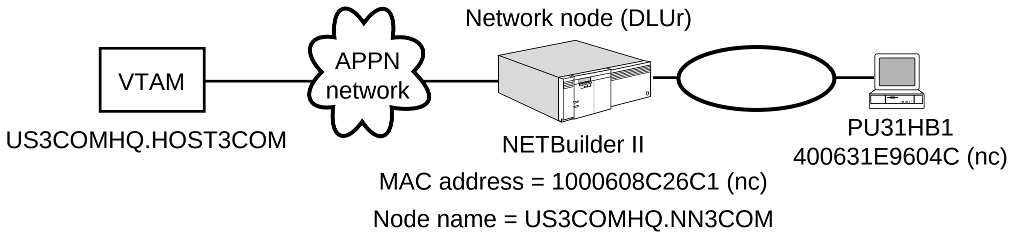


Figure N-5 VTAM Host Configuration for DLUs/DLUr

Table N-11 is an example configuration with a VTAM host as a DLUs node and a network node acting as the DLUr. The table depicts how parameters must match to make the configuration work. This configuration assumes that VTAM is configured for APPN and is at least level 4.2 or higher and that DLUr and DLUs can establish LU6.2 sessions with each other. Both DLCADDR statements are required for DLUs initiated activation.

Table N-11 VTAM Configuration for DLUr

PU Definition on VTAM	DLUr Link Station on DLUr Node	Host Link Definition on PU 2.x

PU31HB1 PU ADDR=01
ANS=CONT,
DLOGMOD=D4C32782,
...
IDBLK=05D,
IDNUM=24001,
...
HB1PATH PATH PID=1,
DLURNAME=PEBBLE,
DLCADDR=(1,C,INTPU),
DLCADDR=(2,X,05D24001)
	`ADD !1 DlurLinkStation 1033 Ncmac 400031E9604C PU31HB1 Dlus = HOST3COM`	Local node ID = 05D24001
		LAN Destination Address = 1000608C26C1 (nc)
		MAC Address of DLUr

APPN Sense Codes

This section lists APPN sense codes. Table N-12 lists the APPN primary return sense codes.

Table N-12 APPN Primary Return Sense Codes

Sense Codes	Hex
OK	(0x0000)
PARAMETER_CHECK	(0x0100)
STATE_CHECK	(0x0200)
ALLOCATION_ERROR	(0x0300)
DEALLOC_ABEND	(0x0500)
DEALLOC_ABEND_PROG	(0x0600)
DEALLOC_ABEND_SVC	(0x0700)
DEALLOC_ABEND_TIMER	(0x0800)
DEALLOC_NORMAL	(0x0900)
PROG_ERROR_NO_TRUNC	(0x0C00)
PROG_ERROR_TRUNC	(0x0D00)
PROG_ERROR_PURGING	(0x0E00)
CONV_FAILURE_RETRY	(0x0F00)
CONV_FAILURE_NO_RETRY	(0x1000)
SVC_ERROR_NO_TRUNC	(0x1100)
SVC_ERROR_TRUNC	(0x1200)
SVC_ERROR_PURGING	(0x1300)
UNSUCCESSFUL	(0x1400)
CNOS_PARTNER_LU_REJECT	(0x1800)
CONVERSATION_TYPE_MIXED	(0x1900)
NODE_STOPPING	(0x1A00)
NODE_NOT_STARTED	(0x1B00)
CANCELLED	(0x2100)
BACKED_OUT	(0x2200)
CONVERSATION_ENDED	(0x4200)
THREAD_BLOCKING	(0xF006)
INDICATION	(0x0210)
ACTIVATION_FAIL_RETRY	(0x0310)
ACTIVATION_FAIL_NO_RETRY	(0x0410)
LU_SESS_LIMIT_EXCEEDED	(0x0510)
FUNCTION_NOT_SUPPORTED	(0x0610)
TP_BUSY	(0x02F0)
COMM_SUBSYSTEM_ABENDED	(0x03F0)
COMM_SUBSYSTEM_NOT_LOADED	(0x04f0)
UNEXPECTED_SYSTEM_ERROR	(0x11F0)
INVALID_VERB	(0xFFFF)

Table N-13 lists the APPN secondary return sense codes.

Table N-13 APPN Secondary Return Sense Codes

Sense Codes	Hex
ALLOCATE_NOT_PENDING	(0x09050000L)
ALLOCATION_FAILURE_NO_RETRY	(0x04000000L)
ALLOCATION_FAILURE_RETRY	(0x05000000L)
INVALID_NODE_TYPE_FOR_HPR	(0xC8020000L)
BAD_CONV_ID	(0x02000000L)
BAD_CONV_TYPE	(0x11000000L)
BAD_ERROR_DIRECTION	(0x05010000L)
BAD_LL	(0xF1000000L)
BAD_REMOTE_LU_ALIAS	(0x03000002L)
BAD_RETURN_CONTROL	(0x14000000L)
BAD_RETURN_STATUS_WITH_DATA	(0xD7000000L)
BAD_SECURITY	(0x13000000L)
BAD_SYNC_LEVEL	(0x12000000L)
BAD_TP_ID	(0x01000000L)
BAD_TYPE	(0x50020000L)
BO_NO_RESYNC	(0x00002408L)
BO_RESYNC	(0x01002408L)
CONFIRMED_BAD_STATE	(0x41000000L)
CONFIRM_BAD_STATE	(0x32000000L)
CONFIRM_NOT_LL_BDY	(0x33000000L)
CONFIRM_ON_SYNC_LEVEL_NONE	(0x31000000L)
COS_NAME_NOT_DEFD	(0x10080000L)
CP_OR_SNA_SVCMG_UNDELETABLE	(0xF3010000L)
CPSVCMG_ALREADY_DEFD	(0x21020000L)
DEALLOC_BAD_TYPE	(0x51000000L)
DEALLOC_CONFIRM_BAD_STATE	(0x53000000L)
DEALLOC_FLUSH_BAD_STATE	(0x52000000L)
DEALLOC_LOG_LL_WRONG	(0x57000000L)
DEALLOC_NOT_LL_BDY	(0x55000000L)
DEF_PLU_INVALID_FQ_NAME	(0x74020000L)
DEL_MODE_DEFAULT_SPCD	(0xF4010000L)
DLC_ACTIVE	(0x01100000L)
DUPLICATE	(0x8D020000L)
DUPLICATE_CP_NAME	(0x02100000L)
DUPLICATE_DEST_ADDR	(0x03100000L)
DUPLICATE_TG_NUMBER	(0x15530000L)
DLC_DEACTIVATING	(0x86020000L)
ALREADY_STARTING	(0xC0010000L)
DUPLICATE_ADJ_NODE_ID	(0x04100000L)

(continued)

Table N-13 APPN Secondary Return Sense Codes (continued)

Sense Codes	Hex
DUPLICATE_PORT	(0x10100000L)
DUPLICATE_PORT_NUMBER	(0x05100000L)
DUPLICATE_PORT_NAME	(0x06100000L)
FLUSH_NOT_SEND_STATE	(0x61000000L)
INVALID_AUTO_ACT_SUPP	(0xB5020000L)
INVALID_CN_NAME	(0x21080000L)
INVALID_CNOS_SLIM	(0x17020000L)
INVALID_COS_SNASVCMG_MODE	(0x1C020000L)
INVALID_CP_NAME	(0xCA010000L)
INVALID_DATA_TYPE	(0x05070000L)
INVALID_DEFAULT_RU_SIZE	(0x1D020000L)
INVALID_DLC	(0x10050000L)
INVALID_DLC_NAME	(0x07100000L)
INVALID_DLC_TYPE	(0x08100000L)
INVALID_FQ_LU_NAME	(0xFD010000L)
INVALID_FQ_OWNING_CP_NAME	(0xDB020000L)
INVALID_LIMITED_RESOURCE	(0xCE010000L)
INVALID_LINK_ACTIVE_LIMIT	(0x09100000L)
INVALID_LINK_NAME	(0xC1010000L)
INVALID_LINK_NAME_SPECIFIED	(0xB0020000L)
INVALID_LU_ALIAS	(0xB1020000L)
INALID_MAX_NEGOT_SESS_LIM	(0x14020000L)
INVALID_MIN_CONWINNERS	(0x1E020000L)
INVALID_MODE_NAME	(0x15020000L)
INVALID_NAME_LEN	(0xC5020000L)
INVALID_NETID_LEN	(0xC6020000L)
INVALID_NODE_TYPE	(0xC4020000L)
INVALID_NUM_LS_SPECIFIED	(0xB2020000L)
INVALID_NUM_PORTS_SPECIFIED	(0x0B100000L)
INVALID_NUMBER_OF_NODE_ROWS	(0x02080000L)
INVALID_NUMBER_OF_TG_ROWS	(0x09080000L)
INVALID_PORT_NAME	(0x0C100000L)
INVALID_PORT_TYPE	(0x0D100000L)
INVALID_RECV_PACING_WINDOW	(0x16020000L)
INVALID_TARGET_PACING_CNT	(0x18020000L)
INVALID_TG_CHARS	(0x18030000L)
INVALID_TG_NUMBER	(0x15500000L)
INVALID_MAX_RU_SIZE_UPPER	(0x19020000L)
INVALID_SET_PROT	(0x00070000L)
INVALID_NEW_PROT	(0x01070000L)
INVALID_SET_UNPROT	(0x02070000L)

(continued)

Table N-13 APPN Secondary Return Sense Codes (continued)

Sense Codes	Hex
INVALID_NEW_UNPROT	(0x03070000L)
INVALID_SET_USER	(0x04070000L)
INVALID_SNASVCMG_MODE_LIMIT	(0x1A020000L)
INVALID_UNINT_PLU_NAME	(0x7C020000L)
INVALID_WILDCARD_NAME	(0x8C020000L)
INVALID_STATS_TYPE	(0x06070000L)
INVALID_TABLE_TYPE	(0x07070000L)
LINK_ACT_BY_LOCAL	(0x15100000L)
LINK_ACT_BY_REMOTE	(0x14100000L)
LINK_DEACTIVATED	(0x13100000L)
LINK_DEACT_IN_PROGRESS	(0x12100000L)
LINK_NOT_DEFD	(0x17100000L)
LOCAL_CP_NAME	(0xD7010000L)
LS_ACTIVE	(0xDA010000L)
MISSING_CP_NAME	(0x15510000L)
MISSING_CP_TYPE	(0x15520000L)
MISSING_TG_NUMBER	(0x15550000L)
MODE_NAME_NOT_DEFD	(0xF5010000L)
MODE_SESS_LIM_EXCEEDS_NEG	(0x20020000L)
MODE_UNDELETABLE	(0xF6010000L)
NO_PORTS_DEFINED_ON_DLC	(0x0F100000L)
NO_USE_OF_SNASVCMG	(0x17000000L)
NO_USE_OF_SNASVCMG_CPSVCMG	(0x17000000L)
NODE_ROW_WGT_LESS_THAN_LAST	(0x04080000L)
PARALLEL_TGS_NOT_ALLOWED	(0x15570000L)
PIP_LEN_INCORRECT	(0x16000000L)
PORT_ACTIVE	(0x0E100000L)
PORT_DEACTIVATED	(0x08070000L)
PS_CREATION_FAILURE	(0x18100000L)
P_TO_R_INVALID_TYPE	(0xA1000000L)
P_TO_R_NOT_LL_BDY	(0xA2000000L)
P_TO_R_NOT_SEND_STATE	(0xA3000000L)
RCV_AND_POST_BAD_FILL	(0xD5000000L)
RCV_AND_POST_BAD_STATE	(0xD100000L)
RCV_AND_POST_NOT_LL_BDY	(0xD200000L)
RCV_AND_WAIT_BAD_FILL	(0xB500000L)
RCV_AND_WAIT_BAD_STATE	(0xB100000L)
RCV_AND_WAIT_NOT_LL_BDY	(0xB200000L)
RCV_IMMD_BAD_FILL	(0xC400000L)
RCV_IMMD_BAD_STATE	(0xC100000L)
R_T_S_BAD_STATE	(0xE100000L)

(continued)

Table N-13 APPN Secondary Return Sense Codes (continued)

Sense Codes	Hex
SECURITY_NOT_VALID	(0x51600F08L)
SEND_DATA_CONFIRM_SYNC_NONE	(0xF5000000L)
SEND_DATA_INVALID_TYPE	(0xF4000000L)
SEND_DATA_NOT_LL_BDY	(0xF6000000L)
SEND_DATA_NOT_SEND_STATE	(0xF2000000L)
SEND_ERROR_BAD_TYPE	(0x03010000L)
SEND_ERROR_LOG_LL_WRONG	(0x02010000L)
SNA_DEFD_COS_CANT_BE_CHANGE	(0x0A080000L)
SNA_DEFD_COS_CANT_BE_DELETE	(0x11080000L)
STOP_PORT_PENDING	(0x11100000L)
TG_NUMBER_IN_USE	(0x15540000L)
TG_ROW_WGT_LESS_THAN_LAST	(0x05080000L)
TRANS_PGM_NOT_AVAIL_NO_RTRY	(0x00004C08L)
TRANS_PGM_NOT_AVAIL_RETRY	(0x31604B08L)
TP_NAME_NOT_RECOGNIZED	(0x21600810L)
UNKNOWN_PARTNER_MODE	(0x18000000L)
UNRECOGNIZED_DEACT_TYPE	(0x0E050000L)
LU_NAME_WILDCARD_NAME_CLASH	(0x8E020000L)
TP_ACTIVE	(0x19100000L)
MODE_ACTIVE	(0x1A100000L)
PLU_ACTIVE	(0x1B100000L)
INVALID_PLU_NAME	(0x1C100000L)
INVALID_SET_NEGOTIABLE	(0x1D100000L)
INVALID_MODE_NAME_SELECT	(0x1E100000L)
INVALID_RESPONSIBLE	(0x1F100000L)
INVALID_DRAIN_SOURCE	(0x20100000L)
INVALID_DRAIN_TARGET	(0x21100000L)
INVALID_FORCE	(0x22100000L)
INVALID_CLEANUP_TYPE	(0x24100000L)
INVALID_COS_NAME	(0x25100000L)
INVALID_SESSION_LIMIT	(0x26100000L)
INVALID_DRAIN	(0x27100000L)
INVALID_PRLL_SESS_SUPP	(0x28100000L)
INVALID_LU_NAME	(0x29100000L)
MODE_NOT_RESET	(0x2A100000L)
MODE_RESET	(0x2B100000L)
CNOS_REJECT	(0x2C100000L)
CNOS_COMMAND_RACE_REJECT	(0x5F010000L)
CNOS_MODE_NAME_REJECT	(0x57010000L)
INVALID_OP_CODE	(0x2D100000L)
EXCEEDS_MAX_ALLOWED	(0x5C010000L)

(continued)

Table N-13 APPN Secondary Return Sense Codes (continued)

Sense Codes	Hex
DEACT_CG_INVALID_CGID	(0x6C020000L)
INVALID_SESSION_ID	(0x12050000L)
LU_NAU_ADDR_ALREADY_DEFD	(OX12020000L)
DIR_ENTRY_PARENT	(0x38100000L)
NODE_ALREADY_STARTED	(0xZ3910000L)
NODE_FAILED_TO_START	(0x3A100000L)
LU_ALREADY_DEFINED	(0x3B100000L)
PORT_INACTIVE	(0x3D100000L)
ACTIVATION_LIMITS_REACHED	(0x3E100000L)
PARALLEL_TGS_NOT_SUPPORTED	(0x3F100000L)
DLC_INACTIVE	(0x40100000L)
NO_LINKS_DEFINED	(0x41100000L)
STOP_DLC_PENDING	(0x42100000L)
INVALID_LS_ROLE	(0x43100000L)
INVALID_BTU_SIZE	(0x44100000L)
LAST_LINK_ON_ACTIVE_PORT	(0x45100000L)
DYNAMIC_LOAD_ALREADY_REGD	(0x46100000L)
INVALID_LIST_OPTION	(0x47100000L)
INVALID_RES_NAME	(0x48100000L)
INVALID_RES_TYPE	(0x49100000L)
INVALID_ADJ_NNCP_NAME	(0x4A100000L)
INVALID_NODE	(0x4B100000L)
INVALID_ORIGIN_NODE	(0x4C100000L)
INVALID_TG	(0x4D100000L)
INVALID_FQPCID	(0x4E100000L)
INVALID_POOL_NAME	0x4F1000000L)
INVALID_NAU_ADDRESS	(0x50100000L)
INVALID_ENABLE_POOL	(0x50300000L)
LU_NAME_POOL_NAME_CLASH	(0x51100000L)
INVALID_PRIORITY	(0x52100000L)
INVALID_DNST_LU_NAME	(0x53100000L)
INVALID_HOST_LU_NAME	(0x54100000L)
PU_NOT_DEFINED	(0x55100000L)
INVALID_PU_NAME	(0x56100000L)
INVALID_MAX_IFRM_RCVD	(0x57100000L)
INVALID_SYM_DEST_NAME	(0x58100000L)
INVALID_LENGTH	(0x59100000L)
INVALID_ISR_THRESHOLDS	(0x5A100000L)
INVALID_NUM_LUS	(0x5B100000L)
CANT_DELETE_ADJ_ENDNODE	(0x5C100000L)
INVALID_RESOURCE_TYPE	(0x5D100000L)

(continued)

Table N-13 APPN Secondary Return Sense Codes (continued)

Sense Codes	Hex
PU_CONC_NOT_SUPPORTED	(0x5E100000L)
DLUR_NOT_SUPPORTED	(0x5F100000L)
INVALID_RTP_CONNECTION	(0x60100000L)
PATH_SWITCH_IN_PROGRESS	(0x61100000L)
HPR_NOT_SUPPORTED	(0x62100000L)
RTP_NOT_SUPPORTED	(0x63100000L)
COS_TABLE_FULL	(0x64100000L)
INVALID_DAYS_LEFT	(0x65100000L)
CONVERSATION_TYPE_MISMATCH	(0x34600810L)
PIP_NOT_ALLOWED	(0x31600810L)
SYNC_LEVEL_NOT_SUPPORTED	(0x41600810L)
PLU_ALIAS_CANT_BE_CHANGED	(0xB3020000L)
PLU_ALIAS_ALREADY_USED	(0xB4020000L)
LU_ALIAS_CANT_BE_CHANGED	(0xB8020000L)
LU_ALIAS_ALREADY_USED	(0xB9020000L)
UNKNOWN_USER	(0x32100000L)
NO_PROFILES	(0x33100000L)
TOO_MANY_PROFILES	(0x36100000L)
INVALID_UPDATE_TYPE	(0x37100000L)
INVALID_USERID	(0x90020000L)
INVALID_PASSWORD	(0x91020000L)
INVALID_PROFILE	(0x93020000L)
INVALID_DLUS_NAME	(0x00900000L)
NO_DEFAULT_DLUS_DEFINED	(0x01900000L)
INVALID_PU_ID	(0x02900000L)
PU_ALREADY_ACTIVATING	(0x03900000L)
PU_ALREADY_DEACTIVATING	(0x04900000L)
PU_ALREADY_ACTIVE	(0x05900000L)
PU_NOT_ACTIVE	(0x06900000L)
DLUS_REJECTED	(0x07900000L)
DLUS_CAPS_MISMATCH	(0x08900000L)
PU_FAILED_ACTPU	(0x09900000L)
PU_NOT_RESET	(0x0A900000L)
PU_OWNS_LUS	(0x0B900000L)
INVALID_FILTER_OPTION	(0x0C900000L)
INVALID_STOP_TYPE	(0x0D900000L)
PU_ALREADY_DEFINED	(0x0E900000L)
DEPENDENT_LU_NOT_SUPPORTED	(0x0F900000L)
INVALID_DSPU_NAME	(0x12900000L)
DSPU_ALREADY_DEFINED	(0x13900000L)
INVALID_SOLICT_SSCP_SESS	(0x14900000L)

(continued)

Table N-13 APPN Secondary Return Sense Codes (continued)

Sense Codes	Hex
INVALID_BACK_LEVEL_SUPPORT	(0x15000000L)
INVALID_BKUP_DLUS_NAME	(0x15900000L)
INVALID_EFFECTIVE_CAPACITY	(0x24080000L)
INVALID_TIME_COST	(0xD6010000L)
INVALID_TP_NAME	(0xA0020000L)
INVALID_BYTE_COST	(0xD1010000L)
DEF_LINK_INVALID_SECURITY	(0x22080000L)
INVALID_PROPAGATION_DELAY	(0x23080000L)
INVALID_USER_DEF_1	(0xC3010000L)
INVALID_USER_DEF_2	(0xC4010000L)
INVALID_USER_DEF_3	(0xC5010000L)
AS_NEGOTIATED	(0x07000000L)
AS_SPECIFIED	(0x00000000L)
FORCED	(0xB7020000L)
INVALID_LS_NAME	(0xB7030000L)
INVALID_LFSID_SPECIFIED	(0xB7040000L)
INVALID_FILTER_TYPE	(0xB7050000L)
INVALID_MESSAGE_TYPE	(0xB7060000L)
CANT_DELETE_CP_LU	(0xB7070000L)
ALL_RESOURCES_NOT_DEFINED	(0xB7090000L)
INVALID_LIST_TYPE	(0xB70A0000L)

O

IBM TRACE FACILITY

This appendix describes how to set up filters to capture traces of data link switching (DLSw), Logical Link Control type 2 (LLC2), or Synchronous Data Link Control (SDLC) packets to troubleshoot IBM network environment problems. The trace facility uses mnemonic filtering masks to filter specific types of packets for tracing purposes. For more information about mnemonic filtering, refer to Chapter 4. For more information about parameters in the FIlter Service, refer to Chapter 23 in *Reference for NETBuilder Family Software*.

Tracing IBM Data Traffic

You can trace IBM data traffic of the following packet frame types:

- DLSw
- LLC2
- SDLC

This appendix is divided into sections showing how to trace each packet frame type.

In the examples in this chapter, all MAC addresses must be entered in noncanonical format.

Tracing DLSw Packets

To set up a trace for DLSw packets, you set up mnemonic filters and masks, follow these steps:

1 Set up the mask using:

```
ADD -FIlter MASK <maskname> <location> <pattern>
 <location>: = <mnemonic format>
 <mnemonic format>: = <protocol>.<field> <protocol>:= DLSW
 <field>:= DLSwLclMAC | DLSwLclSAP | DLSwRmtMAC |
 DLSwRmtSAP | IPADDRess <maskname> is an arbitrary string of 15
 printable characters
```

You can set up the field in the mask in several ways to trace specific types of packets from the following locations:

- DLSw local MAC address
- Remote MAC address
- DLSw local SAP
- DLSw remote SAP
- A specific IP address

Table O-1 lists the possible fields and the appropriate matching value. For examples of how to set up these types of masks, see the specific examples beginning on page O-3.

Table O-1 Field Values for DLSw Traces

Field	Description	Matching Value
DlswLclMAC	Local MAC address	<MAC address>
DlswLclSAP	Local SAP	<hexadecimal value>
DlswRmtMAC	Remote MAC address	<MAC address>
DlswRmtSAP	Remote SAP	<hexadecimal value>
IPADDRess	IP address of the DLSw tunnel	<IP address>

You can display these values by entering:

```
SHow -FIlter MNemonics DLSw
```

2 Set up the filter policy using:

```
ADD -FIlter POLicy <policyname><action> <masks>
```

Specify the action as TRace.

For the <masks> value, you can select one of two built-in masks and/or masks you have defined. Table O-2 lists the built-in masks for tracing different types of packets.

Table O-2 Built-in Masks for Tracing DLSw Packets

Built-in Mask	Equivalent	Packet Type
DLSCTL	DLSW.1=72	DLSw Control Message
DLSWI	DLSW.1=16	DLSw Information Message

For examples of how to set specific DLSw masks and policies, refer to “DLSw Filter Examples” on page O-3.

When setting policies for DLSw, the only action allowed is TRace.

3 Set the maximum number of bytes to be captured in the trace using:

```
SETDefault -DLSw MaxTRaceData = <max_bytes_captured> (0-76)
```

This parameter sets the number of bytes captured over and above the DLSw message headers. The number of bytes you capture determines the quality of the trace data. The more bytes you capture, the more information you will receive. The number you specify will be rounded up to the nearest multiple of four when determining how many bytes to capture. For example, if you set the value to 29, the maximum number of bytes to be captured is rounded up to 32.

4 Set the filter selection by entering:

```
SETDefault FIlter SELection = DLSW
```

5 Enable the FIlter Service by entering one of the following commands:

```
SETDefault -FIlter CONTrol = (Enabled, MatchOne)
```

or

```
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

For more information about parameters in the FIlter Service, refer to Chapter 23 in *Reference for NETBuilder Family Software*.

Displaying DLSw Trace Data

To display the trace data, enter:

```
SHow -DLSw TRaceData
```

DLSw Filter Examples

This section provides examples for setting up different filters for tracing DLSw packets.

Example 1

Tracing DLSw Packets from a Local MAC Address

To trace DLSw packets from DLSw local MAC address %600631244F6F with a mask named DLSW1and policy name EX1, enter:

```
ADD -FIlter MASK DLSW1 DLSW.DlswLclMac = %00631244F6F
ADD -FIlter POLicy EX1 TRace DLSW1
```

Example 2

Tracing DLSw Packets from a Local SAP

To trace DLSw packets from DLSw local SAP %04 with a mask named DLSW2 and policy name EX2, enter:

```
ADD -FIlter MASK DLSW2 DLSW.DlswLclSap = %04
ADD -FIlter POLicy EX2 TRace DLSW2
```

Example 3

Tracing DLSw Packets from a Remote MAC Address

To trace DLSw packets from DLSw remote MAC address %6020000C0E854 with a mask named DLSW3 and policy name EX3, enter:

```
ADD -FIlter MASK DLSW3 DLSW.DlswRmtMac = %020000C0E854
ADD -FIlter POLicy EX3 TRace DLSW3
```

Example 4

Tracing DLSw Packets from a Remote SAP

To trace DLSw packets from DLSw remote SAP %04 with a mask named DLSW4 and policy name EX4, enter:

```
ADD -FIlter MASK DLSW4 DLSW.DlswRmtSap = %04
ADD -FIlter POLicy EX4 TRace DLSW4
```

Example 5

Tracing DLSw Packets from an IP Address

To trace DLSw packets from IP address 129.213.240.230 with a mask named DLSW5 and policy name EX5, enter:

```
ADD -FIlter MASK DLSW5 DLSW.IPADDRess = 129.213.240.230
ADD -FIlter POLicy EX5 TRace DLSW5
```

Example 6

Tracing DLSw Control Message Packets from a Local MAC Address

To trace DLSw control message packets from DLSw local MAC address %600631244F6F with a mask named DLSW6 and policy name EX6, enter:

```
ADD -FIlter MASK DLSW6 DLSW.DlswLclMac = %00631244F6F
ADD -FIlter POLicy EX6 TRace DLSWCTL,DLSW6
```

Example 7

Tracing DLSw Control Message Packets from a Local SAP

To trace DLSw control message packets from DLSw local SAP %04 with a mask named DLSW7 and policy name EX7, enter:

```
ADD -FIlter MASK DLSW7 DLSW.DlswLclSap = %04
ADD -FIlter POLicy EX7 TRace DLSWCTL,DLSW7
```

Example 8

Tracing DLSw Control Message Packets from a Remote MAC Address

To trace DLSw control message packets from DLSw remote MAC address %6020000C0E854 with a mask named DLSW8 and policy name EX8, enter:

```
ADD -FIlter MASK DLSW8 DLSW.DlswRmtMac = %020000C0E854
ADD -FIlter POLicy EX8 TRace DLSWCTL,DLSW8
```

Example 9 Tracing DLSw Control Message Packets from a Remote SAP

To trace DLSw control message packets from DLSw remote SAP %04 with a mask named DLSW9 and policy name EX9, enter:

```
ADD -FIlter MASK DLSW9 DLSW.DlswRmtSap = %04
ADD -FIlter POLicy EX9 TRace DLSWCTL,DLSW9
```

Example 10 Tracing DLSw Control Message Packets from an IP Address

To trace DLSw control message packets from IP address 129.213.240.230 with a mask named DLSW10 and policy name EX10, enter:

```
ADD -FIlter MASK DLSW10 DLSW.IPADDRess = 129.213.240.230
ADD -FIlter POLicy EX10 TRace DLSWCTL,DLSW10
```

Example 11 Tracing DLSw Information Message Packets from a Local MAC Address

To trace DLSw information message packets from DLSw local MAC address %00631244F6F with a mask named DLSW11 and policy name EX11, enter:

```
ADD -FIlter MASK DLSW11 DLSW.DlswLclMac = %00631244F6F
ADD -FIlter POLicy EX11 TRace DLSWI,DLSW11
```

Example 12 Tracing DLSw Information Message Packets from a Local SAP

To trace DLSw information message packets from DLSw local SAP %04 with a mask named DLSW12 and policy name EX12, enter:

```
ADD -FIlter MASK DLSW12 DLSW.DlswLclSap = %04
ADD -FIlter POLicy EX12 TRace DLSWI,DLSW12
```

Example 13 Tracing DLSw Information Message Packets from a Remote MAC Address

To trace DLSw information message packets from DLSw remote MAC address %6020000C0E854 with a mask named DLSW13 and policy name EX13, enter:

```
ADD -FIlter MASK DLSW13 DLSW.DlswRmtMac = %020000C0E854
ADD -FIlter POLicy EX13 TRace DLSWI,DLSW11
```

Example 14 Tracing DLSw Information Message Packets from a Remote SAP

To trace DLSw information message packets from DLSw remote SAP %04 with a mask named DLSW14 and policy name EX14, enter:

```
ADD -FIlter MASK DLSW14 DLSW.DlswRmtSap = %04
ADD -FIlter POLicy EX14 TRace DLSWI,DLSW14
```

Example 15 Tracing DLSw Information Message Packets from an IP Address

To trace DLSw information message packets from IP address 129.213.240.230 with a mask named DLSW15 and policy name EX15, enter:

```
ADD -FIlter MASK DLSW15 DLSW.IPADDRess = 129.213.240.230
ADD -FIlter POLicy EX15 TRace DLSWI,DLSW15
```

Tracing LLC2 Frames

To set up a trace for LLC2 frames, you set up mnemonic filters and masks, follow these steps:

1 Set up the mask using:

```
ADD -FIlter MASK <maskname> <location> <pattern>
 <location>: = <mnemonic format>
```

```
<mnemonic format>: = <protocol>.<field> <protocol>:= LLC2
 <field>:= FrameType | LlcLclMAC | LlcLclSAP |
 LlcRmtMAC | LlcRmtSAP <pattern>:= <comparison><match>
 <match>:= LlcInfoFrame | LlcUnnFrame | LlcSupFrame
```

You can set up the field in the mask in several ways to trace specific types of packets: For examples of how to set up these types of masks, see the specific examples following this section. Table O-3 lists the field options available for tracing LLC2 packets from different origins and targets. For examples of how to set up these types of masks, refer to the specific examples in "LLC2 Filter Examples" on page O-6.

Table O-3 Field Values for LLC2 Traces

Field	Description	Matching Value
LlcLclMAC	LocalMAC address	<MAC address>
LlcLclSAP	Local SAP	<hexadecimal value>
LlcRmtMAC	Remote MAC address	<MAC address>
LlcRmtSAP	Remote SAP	<hexadecimal value>
FrameType	LLC2 frame type	<frame_type mnemonic>

If you specify FrameType as the field value, when you set up the pattern, you set up the comparison and match. For the match, you specify the frame type mnemonic you want matched. Table O-4 lists the frame type mnemonic options for tracing LLC2 packets.

Table O-4 Frame Type Mnemonics for LLC2 Traces

Frame Type Mnemonic	Equivalent	Packet Type
LlcInfoFrame	LLC2.FrameType = %0	LLC2 information frames
LlcUnnFrame	LLC2.FrameType = %3	LLC2 unnumbered frames
LlcSupFrame	LLC2.FrameType = %1	LLC2 supervisory frames

You can display these values by entering:

```
SHow -FIlter MNemonics LLC
```

2. Set up the filter policy using:

```
ADD -FIlter POLicy <policyname><action> <masks> <context>
```

Specify the action as TRace.

For examples of how to set specific LLC2 masks and policies, refer to "LLC2 Filter Examples" on page O-6.

When setting policies for LLC2, the only action allowed is TRace.

3. Set the maximum number of bytes to be captured in the trace using:

```
SETDefault -LLC2 MaxTRaceData = <max_bytes_captured> (0-76)
```

This parameter sets the number of bytes captured over and above the LLC2 address and control bytes. The number of bytes you capture determines the quality of the trace data. The more bytes you capture, the more information you receive. The number you specify is rounded up to the nearest multiple of four.

For example, if you set the value to 29, the maximum number of bytes to be captured is rounded up to 32.

4 Set the filter selection by entering:

```
SETDefault -FIlter SELection = LLC
```

5 Enable the FIlter Service by entering one of the following commands:

```
SETDefault -FIlter CONTrol = (Enabled, MatchOne)
```

or

```
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

For more information about parameters in the FIlter Service, refer to Chapter 23 in *Reference for NETBuilder Family Software*.

Displaying LLC2 Trace Data

After you have conducted your trace, you can display the trace data by entering:

```
SHow -LLC2 TRaceData
```

LLC2 Filter Examples

This section provides examples for setting up different filters for tracing LLC2 packets.

Example 1

Tracing LLC2 Packets from a Local MAC Address

To trace LLC2 packets from local MAC address %6080002057Ab0 with a mask named LLC2_16 and policy name LLC2TRACE1, enter:

```
ADD -FIlter MASK LLC2_16 LLC2.LLC2LclMac = %080002057AB0
ADD -FIlter POLicy LLC2TRACE1 TRace LLC,LLC2_16
```

Example 2

Tracing LLC2 Packets from a Local SAP

To trace LLC2 packets from local SAP %08 with a mask named LLC2_17 and policy name LLC2TRACE2, enter:

```
ADD -FIlter MASK LLC2_17 LLC2.LLC2LclSap = %08
ADD -FIlter POLicy LLC2TRACE2 TRace LLC,LLC2_17
```

Example 3

Tracing LLC2 Packets from a Remote MAC Address

To trace LLC2 packets from remote MAC address %600608C23EBBC with a mask named LLC2_18 and policy name LLC2TRACE3, enter:

```
ADD -FIlter MASK LLC2_18 LLC2.LLC2RmtMac = %00608C23EBBC
ADD -FIlter POLicy LLC2TRACE3 TRace LLC,LLC2_18
```

Example 4

Tracing LLC2 Packets from a Remote SAP

To trace LLC2 packets from remote SAP %1C with a mask named LLC2_19 and policy name LLC2TRACE4, enter:

```
ADD -FIlter MASK LLC2_19 LLC2.LLC2RmtSap = %1C
ADD -FIlter POLicy LLC2TRACE4 TRace LLC,LLC2_19
```

Example 5

Tracing LLC2 Information Frames from a Local MAC Address

To trace LLC2 information frames from local MAC address %6080002057AB0 with masks LLC2_20 and LLC2_20A and policy name LLC2TRACE5, enter:

```
ADD -FIlter MASK LLC2_20 LLC2.LLC2LclMac = %080002057AB0
ADD -FIlter MASK LLC2_20A LLC2.LLC2FrameType = LlcInfoFrame
ADD -FIlter POLicy LLC2TRACE5 TRace LLC2_20,LLC2_20A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 6

Tracing LLC2 Information Frames from a Local SAP

To trace LLC2 information frames from local SAP %08 with masks LLC2_21 and LLC2_21A and policy name LLC2TRACE6, enter:

```
ADD -FIlter MASK LLC2_21 LLC2.LLC2LclSap = %08
ADD -FIlter MASK LLC2_21A LLC2.LLC2FrameType = LlcInfoFrame
ADD -FIlter POLicy LLC2TRACE6 TRace LLC2_21,LLC2_21A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 7

Tracing LLC2 Information Frames from a Remote MAC Address

To trace LLC2 information frames from remote MAC address %600608C23EBBC with masks LLC2_22 and LLC2_22A and policy name LLC2TRACE7, enter:

```
ADD -FIlter MASK LLC2_22 LLC2.LLC2RmtMac = %080002057AB0
ADD -FIlter MASK LLC2_22A LLC2.LLC2FrameType = LlcInfoFrame
ADD -FIlter POLicy LLC2TRACE7 TRace LLC2_22,LLC2_22A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 8

Tracing LLC2 Information Frames from a Remote SAP

To trace LLC2 information frames from remote SAP %1C with masks LLC2_23 and LLC2_23A and policy name LLC2TRACE8, enter:

```
ADD -FIlter MASK LLC2_23 LLC2.LLC2RmtSap = %1C
ADD -FIlter MASK LLC2_23A LLC2.LLC2FrameType = LlcInfoFrame
ADD -FIlter POLicy LLC2TRACE8 TRace LLC2_23,LLC2_23A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 9

Tracing LLC2 Unnumbered Frames from a Local MAC Address

To trace LLC2 unnumbered frames from local MAC address %6080002057AB0 with masks LLC2_24 and LLC2_24A and policy name LLC2TRACE9, enter:

```
ADD -FIlter MASK LLC2_24 LLC2.LLC2LclMac = %080002057AB0
ADD -FIlter MASK LLC2_24A LLC2.LLC2FrameType = LlcUnnFrame
ADD -FIlter POLicy LLC2TRACE9 TRace LLC2_24,LLC2_24A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 10

Tracing LLC2 Unnumbered Frames from a Local SAP

To trace LLC2 unnumbered frames from local SAP %08 with masks LLC2_25 and LLC2_25A and policy name LLC2TRACE10, enter:

```
ADD -FIlter MASK LLC2_25 LLC2.LLC2LclSap = %08
ADD -FIlter MASK LLC2_25A LLC2.LLC2FrameType = LlcUnnFrame
ADD -FIlter POLicy LLC2TRACE10 TRace LLC2_25,LLC2_25A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 11

Tracing LLC2 Unnumbered Frames from a Remote MAC Address

To trace LLC2 unnumbered frames from remote MAC address %600608C23EBBC with masks LLC2_26 and LLC2_26A and policy name LLC2TRACE11, enter:

```
ADD -FIlter MASK LLC2_26 LLC2.LLC2RmtMac = %080002057AB0
ADD -FIlter MASK LLC2_26A LLC2.LLC2FrameType = LlcUnnFrame
ADD -FIlter POLicy LLC2TRACE7 TRace LLC2_26,LLC2_26A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 12 Tracing LLC2 Unnumbered Frames from a Remote SAP

To trace LLC2 unnumbered frames from remote SAP %1C with masks LLC2_27 and LLC2_27A and policy name LLC2TRACE12, enter:

```
ADD -FIlter MASK LLC2_27 LLC2.LLC2RmtSap = %1C
ADD -FIlter MASK LLC2_27A LLC2.LLC2FrameType = LlcUnnFrame
ADD -FIlter POLicy LLC2TRACE12 TRace LLC2_27,LLC2_27A
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Tracing SDLC Frames

To set up a trace for SDLC frames, you set up mnemonic filters and masks, follow these steps:

1 Set up the mask using:

```
ADD -FIlter MASK <maskname> <location> <pattern>
  <location>: = <mnemonic format>
  <mnemonic format>: = <protocol>.<field> <protocol>:= SDLC
  <field>:= FrameType | PollADDRess <pattern>:= <comparison>
  <match> <match>:= SDLCInfoFrame | SDLCUnnFrame | SDLCSupFrame
```

You can set up the field in the mask in several ways to trace specific types of packets. For examples of how to set up these types of masks, see the specific examples following this section. Table O-5 lists the field options available for tracing SDLC packets from different origins and targets. For examples of how to set up these types of masks, refer to the specific examples in “SDLC Filter Examples” on page O-9.

Table O-5 Field Values for SDLC Traces

Field	Description	Matching Value
FrameType	SDLC frame type	<frame_type mnemonic>
PollADDRess	SDLC Poll Address	<hexadecimal value>

If you specify FrameType as the field value, when you set up the pattern, you set up the comparison and match. For the match, you specify the frame type mnemonic you want matched. Table O-6 lists the frame type mnemonic options for tracing LLC2 packets.

Table O-6 Frame Type Mnemonics for SDLC Traces

Frame Type Mnemonic	Equivalent	Packet Type
SDLCInfoFrame	SDLC.FrameType = %0	SDLC Info frames
SDLCUnnFrame	SDLC.FrameType = %3	SDLC unnumbered frames
SDLCSupFrame	SDLC.FrameType = %1	SDLC supervisor frames

You can display these values by entering:

```
SHow -FIlter Mnemonics SDLC
```

2 Set up the filter policy using:

```
ADD -FIlter POLicy <policyname><action> <masks> <context>
```

Specify the action as TRace.

For examples of how to set specific SDLC masks and policies, refer to “SDLC Filter Examples.”

3 Set the maximum number of bytes to be captured in the trace using:

```
SETDefault -SDLC MaxTRaceData = <max_bytes_captured> (0-76)
```

This parameter sets the number of bytes captured over and above the SDLC address and control bytes. The number of bytes you capture determines the quality of the trace data. The more bytes you capture, the more information you receive. The number you specify is rounded up to the nearest multiple of four. For example, if you set the value to 29, the maximum number of bytes to be captured is rounded up to 32.

4 Set the filter selection by entering:

```
SETDefault -FIlter SELection = SDLC
```

5 Enable the FIlter Service by entering one of the following commands:

```
SETDefault -FIlter CONTrol = (Enabled, MatchOne)
```

or

```
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

For more information about parameters in the FIlter Service, refer to Chapter 23 in *Reference for NETBuilder Family Software*.

Displaying SDLC Trace Data

After you have conducted your trace, you can display the trace data by entering:

```
SHow -SDLC TraceData
```

SDLC Filter Examples

This section provides examples for setting up different filters for tracing SDLC packets.

Example 1

Tracing SDLC Packets from a Poll Address

To trace SDLC packets from poll address %C1 with mask SDLC1 and policy SDLCTRACE1 on port 2, enter:

```
ADD -FIlter MASK SDLC1 SDLC.PollADDRess = %C1
ADD -FIlter POLicy SDLCTRACE1 TRace SDLC1 at !2
```

Example 2

Tracing SDLC Information Frames

To trace SDLC information frames from poll address %C1 with masks SDLC1 and SDLC2 and policy SDLCTRACE2 on port 2, enter:

```
ADD -FIlter MASK SDLC1 SDLC.PollADDRess = %C1
ADD -FIlter MASK SDLC2 SDLC.FrameType = SDLCInfoFrame
ADD -FIlter POLicy SDLCTRACE2 TRace SDLC1,SDLC2 at !2
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 3

Tracing SDLC Unnumbered Frames

To trace SDLC unnumbered frames from poll address %C1 with masks SDLC1 and SDLC3 and policy SDLCTRACE3 on port 2, enter:

```
ADD -FIlter MASK SDLC1 SDLC.PollADDRess = %C1
ADD -FIlter MASK SDLC3 SDLC.FrameType = SDLCUnnFrame
ADD -FIlter POLicy SDLCTRACE3 TRace SDLC1,SDLC3 at !2
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

Example 4

Tracing SDLC Unnumbered Frames

To trace SDLC supervisory frames from poll address %C1 with masks SDLC1 and SDLC4 and policy SDLCTRACE4 on port 2, enter:

```
ADD -FIlter MASK SDLC1 SDLC.PollADDRess = %C1
ADD -FIlter MASK SDLC4 SDLC.FrameType = SDLCSupFrame
ADD -FIlter POLicy SDLCTRACE4 TRace SDLC1,SDLC4 at !2
SETDefault -FIlter CONTrol = (Enabled, CheckAll)
```

P

ABBREVIATIONS AND ACRONYMS

This appendix provides a list of the abbreviations and acronyms used in this guide and corresponding NETBuilder documentation.

	Abbreviation/Acronym	Meaning
A	AAL	ATM adaptation layer
	AARP	AppleTalk Address Resolution Protocol
	ABR	area border router
	AC	access control (Access Control when referring to service name)
	AEP	AppleTalk Echo Protocol
	AFP	AppleTalk Filing Protocol
	AFI	authority format identifier
	AMP	Adapter Management Protocol
	ANR	Automatic Network Routing
	ANSI	American National Standards Institute
	API	application program interface
	APPC	Advanced Program-to-Program Communication
	APPN	Advanced Peer-to-Peer Networking
	ARE	All Routes Explorer
	ARP	Address Resolution Protocol
	ARPANET	Advanced Research Projects Agency Network
	AS	autonomous system
	ASBR	Autonomous System Boundary Router
	ASN	autonomous system number
	ATG	address translation gateway
	ATM	Asynchronous Transfer Mode
	ATP	AppleTalk Transaction Protocol
B	BAN	Boundary Access Node
	BBS	bulletin board service
	BDR	backup designated router
	BGP	Border Gateway Protocol
	BMA	broadcast multi-access
	BNN	Boundary Network Node
	BOD	bandwidth-on-demand
	BPDU	Bridge Protocol Data Unit
	BRI	basic rate interface
	BSC	Binary Synchronous Communication

	Abbreviation/Acronym	Meaning
	BSD	Berkeley Software Distribution
	BSI	British Standards Institute
	BTU	basic transmission unit
	BUS	Broadcast and Unknown Server
C	CBPDU	Configuration Bridge Protocol Data Unit
	CBT	Core-Based Trees
	CEC	Communications Engine Card
	CC	configuration change
	CCITT	Consultative Committee for International Telegraph and Telephone
	CCS	**1** common channel signaling (ISDN)
		2 compact configuration store
	CD	**1** carrier detect (signal)
		2 compact disc
		3 collision detection
	CERT	Computer Emergency Response Team
	CHAP	Challenge Handshake Authentication Protocol
	CIDR	Classless Interdomain Routing Protocol
	CLNP	Connectionless Network Protocol
	CLNS	Connectionless Network Service
	CN	connection network
	COS	class of service
	CP	control point
	CR	carriage return
	CRC	cyclic redundancy check
	CS	communications server
	CSMA	carrier sense multiple access
	CSMA/CD	carrier sense multiple access/collision detection
	CSNP	Complete Sequence Number Protocol Data Unit
	CSU	channel service unit
	CTS	clear to send
	CU	control unit
	CUG	closed user group
D	DCD	data carrier detected
	DCE	**1** data communications equipment (EIA expansion)
		2 data circuit-terminating equipment (CCITT)
		3 Distributed Computing Environment (OSF)
	DD	double-density
	DDP	Datagram Delivery Protocol
	DEB	Destination Explicit Blocking
	DEF	Destination Explicit Forwarding
	DHCP	Dynamic Host Configuration Protocol
	DIB	Directory Information Base

	Abbreviation/Acronym	Meaning
	DIS	Designated Intermediate System
	DIT	Directory Information Tree
	DLC	data link control
	DLCI	data link connection identifier
	DLSw	data link switching
	DLT	data link test
	DLUr	dependent LU requester
	DLUs	dependent LU server
	DN	**1** distinguished name
		2 directory number
	DNS	Domain Name Service
	DOD	dial-on-demand (3Com)
	DR	designated router
	DSA	Directory System Agent
	DSAP	destination service access point
	DSP	domain specific part
	DSPU	downstream physical unit
	DSR	data set ready
	DSU	digital service unit
	DTE	data terminal equipment
	DTR	data terminal ready
	DUA	Directory User Agent
	DVMRP	Distance Vector Multicast Routing Protocol
	DXI	data exchange interface
E	EBCDIC	Extended Binary Coded Decimal Interchange Code
	ECM	enter command mode
	ECS	Ether Connect System
	ED	extra-density
	ELAN	Emulated LAN
	EN	end node (APPN)
	ERP	Echo Reply Protocol
	ERQ	echo request
	ES	end system
	ESH	end system hello
	ES-IS	End System-to-Intermediate System
	ETSI	European Telecommunications Standards Institute
F	FAP	File Access Protocol
	FDDI	Fiber Distributed Data Interface
	FEP	front end processor
	FIT	fully initializing terminal
	FS	frame status
	FSE	full status enquiry

	Abbreviation/Acronym	Meaning
	FTAM	File Transfer Access and Management
	FTP	File Transfer Protocol
G	GOSIP	Government Open Systems Interconnection Profile
	GSA	Government Services Administration
H	HD	high-density
	HDLC	high-level data link control
	HPR	High Performance Routing (APPN)
	HSS	high-speed serial
	HSSI	High-Speed Serial Interface
I	IANA	Internet Assigned Numbers Authority
	ICD	International Code Designator
	ICMP	Internet Control Message Protocol
	ICP	Internet Control Protocol
	IDI	initial domain identifier
	IDP	**1** initial domain part (OSI)
		2 Internet Datagram Protocol
	IEN	Internet Engineering Notes
	IETF	Internet Engineering Task Force
	IGMP	Internet Group Management Protocol
	IGP	Interior Gateway Protocol
	IIH	Intermediate System-to-Intermediate System hello packet
	IISIS	Integrated Intermediate System-to-Intermediate System
	ILMI	Interim Local Management Interface
	IP	Internet Protocol
	IPC	interprocessor communication
	IPX	Internetwork Packet Exchange
	IS	intermediate system
	ISDN	Integrated Services Digital Network
	ISH	intermediate system hello
	IS-IS	Intermediate System-to-Intermediate System
	ISO	International Organization for Standardization
	ISR	Intermediate Session Routing (APPN)
	ITCM	Integrated T1 Controller Module
	ITU-TSS	International Telecommunications Union–Telecommunications Standards Sector
L	LAA	LAN Address Administration
	LAP	Link Access Procedure
	LAPB	Link Access Procedure, Balanced
	LANE	LAN Emulation Client
	LAT	local area transport
	LCN	logical channel number
	LCP	Link Control Protocol
	LEM	Link Error Monitor

	Abbreviation/Acronym	Meaning
	LEN	low-entry networking (APPN)
	LEC	LAN Emulation Client
	LECS	LAN Emulation Configuration Server
	LES	LAN Emulation Server
	LF	linefeed
	LFS	largest frame size
	LIS	logical IP subnetwork
	LLC	Logical Link Control
	LLC2	Logical Link Control, type 2
	LMF	Line Management Function
	LMI	Local Management Interface
	LNM	LAN Net Manager
	LS	link state
	LSA	link state advertisement
	LSP	**1** Link State Protocol
		2 link state packets
	LSR	link state information request
	LSU	link state update
	LU	logical unit
	LUNI	LAN Emulation User Network Interface
M	MAC	**1** media access control
		2 media access controller (FDDI)
	MAU	**1** multistation access unit (token ring)
		2 medium access unit (Ethernet)
	MIB	management information base
	MIC	media interface connector
	MIP	Multicast Internet Protocol
	MLN	multiple logical networks
	MLP	Multilink Protocol
	MOSPF	Multicast Open Shortest Path First
	MP	multiprocessor
	MPATM	multiprotocol ATM
	MSB	most significant bit
	MTU	maximum transmission unit
N	NA	Neighbor Acquisition
	NBMA	non-broadcast multi-access interfaces
	NBP	**1** Name Binding Protocol (AppleTalk)
		2 NetBIOS Protocol (3Com)
	NCE	network connection endpoint
	NCP	**1** Network Control Protocol
		2 NetWare Core Protocol (Novell)
		3 Network Control Program (SNA)

	Abbreviation/Acronym	Meaning
	NCS/AT	Network Control Server/AT
	NET	Network Entity Title
	NetBIOS	Network Basic Input/Output System
	NFS	Network File System
	NLPID	Network Layer Protocol Identifier
	NLSP	NetWare Link Services Protocol
	NMI	nonmaskable interrupt
	NMS	Network Management System
	NMU	Network Management Utilities
	NN	network node (APPN)
	NPDU	network protocol data unit
	NR	neighbor reachability
	NRIP	NetWare Routing Information Protocol (Novell)
	NRZ	non-return to zero
	NRZI	non-return to zero inverted
	NSA	National Security Agency
	NSAP	network service access point
	NSF	**1** National Specific Facilities
		2 National Science Foundation
	NT1	network termination 1
O	OSI	Open System Interconnection
	OSIAPPL	Open System Interconnection Applications
	OSPF	Open Shortest Path First
P	PAD	packet assembler/disassembler
	PAP	Password Authentication Protocol
	PCM	**1** physical connection management
		2 pulse code modulation (ISDN)
	PDN	public data network
	PDU	protocol data unit
	PEP	partitioned emulation programming
	PLG	Phone Line Gateway
	PLU	primary logical unit
	PMF	parameter management frame
	PPM	port and path module
	PPP	Point-to-Point Protocol
	PRI	primary rate interface
	PSAP	presentation service access point
	PSDN	packet switching data network
	PSNP	Partial Sequence Number PDU
	PU	physical unit
	PVC	permanent virtual circuit
Q	QOS	quality of service

	Abbreviation/Acronym	Meaning
R	RARP	Reverse Address Resolution Protocol
	RD	**1** route designator
		2 received data (signal)
	RDP	Router Discovery Protocol
	RFC	Request for Comments
	RH	request/response header
	RI	routing information
	RIB	routing information database
	RIF	routing information field
	RII	routing information indicator
	RIP	Routing Information Protocol
	RIPIP	Routing Information Protocol for IP
	RIPXNS	Routing Information Protocol for XNS
	RLSD	received line signal detector
	RMA	Return Materials Authorization
	RMON	Remote Monitoring
	RPB	Reverse Path Broadcasting
	RPF	Reverse Path Forwarding
	RPM	Reverse Path Multicasting
	RSCV	Route Selection Control Vector
	RTMP	Routing Table Maintenance Protocol
	RTP	**1** Routing Table Protocol
		2 routing update packets
		3 Rapid Transport Protocol (APPN HPR)
	RTS	request to send
	RU	request/response unit
S	SAP	**1** Service Advertising Protocol (NetWare)
		2 service access point (OSI and SNA)
	SAS	single-attached station
	SDC	synchronous data compression
	SDLC	Synchronous Data Link Control
	SEB	Source Explicit Blocking
	SEF	Source Explicit Forwarding
	SIO	serial input/output
	SIP	SMDS Interface Protocol
	SLU	secondary logical unit
	SMDS	Switched Multimegabit Data Service
	SMT	Station Management
	SMTP	Simple Mail Transfer Protocol
	SNA	Systems Network Architecture
	SNAP	Subnetwork Access Protocol
	SNI	**1** Subscriber Network Interface

	Abbreviation/Acronym	Meaning
		2 System Network Interconnection
	SNMP	Simple Network Management Protocol
	SNPA	Subnetwork Point of Attachment
	SPF	shortest path first
	SPID	Service Profile Identifiers
	SPT	shortest path tree
	SPX	sequenced packet exchange
	SQL	Structured Query Language
	SR	source route (Source Route when referring to the service)
	SRF	specifically routed frame
	SRT	source-route transparent
	SRTG	source route transparent bridging gateway
	SSAP	source service access point
	SSCP	session services control point
	STA	Spanning Tree Algorithm
	STE	Spanning Tree Explorer
	STP	Spanning Tree Protocol
	SVC	switched virtual circuit
T	TA	terminal adapter
	TACACS	Terminal Access Controller Access Control System
	TCAPPL	Transmission Control Protocol Applications
	TCP	Transmission Control Protocol
	TCP/IP	Transmission Control Protocol/Internet Protocol
	TERM	Terminal (service name)
	TFTP	Trivial File Transfer Protocol
	TG	transmission group
	TH	transmission header
	TOS	type of service
	TRPB	Truncated Reverse Path Broadcasting
	TTL	time-to-live
	TUBA	TCP and UDP with Bigger Addresses
U	UDP	User Datagram Protocol
	UME	User-to-Network Interface Management Entity
	UNI	user-to-network interface
V	VC	virtual circuit (X.25)
		virtual connection (Frame Relay)
		virtual channel (ATM)
	VCC	virtual channel connection
	VCI	virtual channel identifier (APPN)
	VCID	virtual circuit identifier
	VIP	VINES Internet Protocol
	VPI	virtual path identifier

	Abbreviation/Acronym	Meaning
	VPN	virtual private network
	VRN	virtual routing node
	VTP	Virtual Terminal Protocol
X	XID	exchange identification
	XNS	Xerox Network Systems
Z	ZIP	Zone Information Protocol

Q

TECHNICAL SUPPORT

3Com provides easy access to technical support information through a variety of services. This appendix describes these services.

Information contained in this appendix is correct at time of publication. For the very latest, we recommend that you access 3Com Corporation's World Wide Web site as described below.

Online Technical Services

3Com offers worldwide product support 24 hours a day, 7 days a week, through the following online systems:

- World Wide Web site
- 3Com Bulletin Board Service (3ComBBS)
- 3ComFacts^{SM} automated fax service
- 3ComForum on CompuServe® online service

World Wide Web Site

Access the latest networking information on 3Com Corporation's World Wide Web site by entering our URL into your Internet browser:

`http://www.3Com.com/`

This service features the latest information about 3Com solutions and technologies, customer service and support, news about the company, *NetAge*® Magazine, and more.

3Com Bulletin Board Service

3ComBBS contains patches, software, and drivers for all 3Com products, as well as technical articles. This service is available through analog modem or digital modem (ISDN) 24 hours a day, 7 days a week.

Access by Analog Modem

To reach the service by modem, set your modem to 8 data bits, no parity, and 1 stop bit. Call the telephone number nearest you:

Country	Data Rate	Telephone Number
Australia	up to 14400 bps	61 2 9955 2073
Brazil	up to 14400 bps	55 11 547 9666
France	up to 14400 bps	33 1 6986 6954
Germany	up to 28800 bps	4989 62732 188
Hong Kong	up to 14400 bps	852 2537 5608
Italy (fee required)	up to 14400 bps	39 2 27300680
Japan	up to 14400 bps	81 3 3345 7266
Mexico	up to 28800 bps	52 5 520 7853
P. R. of China	up to 14400 bps	86 10 684 92351
Singapore	up to 14400 bps	65 534 5693
Taiwan	up to 14400 bps	886 2 377 5840
U.K.	up to 28800 bps	44 1442 438278
U.S.A.	up to 28800 bps	1 408 980 8204

Access by Digital Modem

ISDN users can dial in to 3ComBBS using a digital modem for fast access up to 56 Kbps. To access 3ComBBS using ISDN, use the following number:

408 654 2703

3ComFacts Automated Fax Service

3Com Corporation's interactive fax service, 3ComFacts, provides data sheets, technical articles, diagrams, and troubleshooting instructions on 3Com products 24 hours a day, 7 days a week.

Call 3ComFacts using your Touch-Tone telephone using one of these international access numbers:

Country	Telephone Number
Hong Kong	852 2537 5610
U.K.	44 1442 438279
U.S.A.	1 408 727 7021

Local access numbers are available within the following countries:

Country	Telephone Number	Country	Telephone Number
Australia	1800 678 515	Netherlands	06 0228049
Belgium	0800 71279	New Zealand	0800 446 398
Denmark	800 17319	Norway	800 11062
Finland	98 001 4444	Portugal	0505 442 607
France	05 90 81 58	Russia (Moscow only)	956 0815
Germany	0130 81 80 63	Singapore	800 6161 463
Hong Kong	800 933 486	Spain	900 964 445
Italy	1678 99085	Sweden	020 792954
Malaysia	1800 801 777	U.K.	0800 626403

3ComForum on CompuServe Online Service

3ComForum is a CompuServe-based service containing patches, software, drivers, and technical articles about all 3Com products, as well as a messaging section for peer support. To use 3ComForum, you need a CompuServe account.

To use 3ComForum:

1. Log on to CompuServe.
2. Type **`go threecom`**
3. Press [Return] to see the 3ComForum main menu.

Support from Your Network Supplier

If additional assistance is required, contact your network supplier. Many suppliers are authorized 3Com service partners who are qualified to provide a variety of services, including network planning, installation, hardware maintenance, application training, and support services.

When you contact your network supplier for assistance, have the following information ready:

- Diagnostic error messages
- A list of system hardware and software, including revision levels
- Details about recent configuration changes, if applicable

If you are unable to contact your network supplier, see the following section on how to contact 3Com.

Support from 3Com

If you are unable to receive support from your network supplier, technical support contracts are available from 3Com.

Contact your local 3Com sales office to find your authorized service provider using one of these numbers:

Regional Sales Office	Telephone Number
3Com Corporation	
P.O. Box 58145	800 NET 3Com *or* 1 408 764 5000
5400 Bayfront Plaza	408 764 5001 (fax)
Santa Clara, California	
95052-8145	
U.S.A.	
3Com Asia Limited	
Australia	61 2 9937 5000 (Sydney)
	61 3 9866 8022 (Melbourne)
China	8610 68492568 (Beijing)
	86 21 63740220 Ext 6115 (Shanghai)
Hong Kong	852 2501 1111
India	91 11 644 3974
Indonesia	6221 572 2088
Japan	81 6 536 3303 (Osaka)
	81 3 3345 7251 (Tokyo)
Korea	822 2 319 4711
Malaysia	60 3 732 7910
New Zealand	64 9 366 9138
Phillippines	632 892 4476
Singapore	65 538 9368
Taiwan	886 2 377 5850
Thailand	662 231 8151 4
3Com Benelux B.V.	
Belgium	32 2 725 0202
Netherlands	31 30 6029700
3Com Canada	
Calgary	403 265 3266
Montreal	514 683 3266
Ottawa	613 566 7055
Toronto	416 498 3266
Vancouver	604 434 3266
3Com European HQ	49 89 627320
3Com France	33 1 69 86 68 00

Regional Sales Office	Telephone Number
3Com GmbH	
Austria	43 1 513 4323
Czech Republic/Slovak Republic	420 2 21845 800
Germany	49 30 34 98790 (Berlin)
(Central European HQ)	49 89 627320 (Munich)
Hungary	36 1 250 83 41
Poland	48 22 6451351
Switzerland	41 31 996 14 14
3Com Ireland	353 1 820 7077
3Com Latin America	
U.S. Headquarters	408 326 2093
Northern Latin America	305 261 3266 (Miami, Florida)
Argentina	541 312 3266
Brazil	55 11 546 0869
Chile	562 633 9242
Colombia	571 629 4110
Mexico	52 5 520 7841/7847
Peru	51 1 221 5399
Venezuela	58 2 953 8122
3Com Mediterraneo	
Italy	39 2 253011 (Milan)
	39 6 5279941 (Rome)
Spain	34 1 383 17 00
3Com Middle East	971 4 349049
3Com Nordic AB	
Denmark	45 39 27 85 00
Finland	358 0 435 420 67
Norway	47 22 18 40 03
Sweden	46 8 632 56 00
3Com Russia	007 095 258 09 40
3Com Southern Africa	27 11 807 4397
3Com UK Ltd.	44 131 220 8228 (Edinburgh)
	44 161 873 7717 (Manchester)
	44 162 889 7000 (Marlow)

Returning Products for Repair

Before you send a product directly to 3Com for repair, you must first obtain a Return Materials Authorization (RMA) number. Products sent to 3Com without RMA numbers will be returned to the sender unopened, at the sender's expense.

To obtain an RMA number, call or fax:

Country	Telephone Number	Fax Number
U.S.A. and Canada	1 800 876 3266, option 2	408 764 7120
Latin America	1 408 326 2927	408 764 7120
Europe, South Africa, and Middle East	44 1442 438125	44 1442 435822
Outside Europe, U.S.A., and Canada	1 408 326 2926	1 408 764 7120

04/22/97

INDEX

3Com Corporation LIMITED WARRANTY

HARDWARE

3Com warrants its hardware products to be free from defects in workmanship and materials, under normal use and service, for the following lengths of time from the date of purchase from 3Com or its Authorized Reseller:

Network adapters	Lifetime
Other hardware products (unless otherwise specified in the warranty statement above)	1 year
Spare parts and spares kits	90 days

If a product does not operate as warranted above during the applicable warranty period, 3Com shall, at its option and expense, repair the defective product or part, deliver to Customer an equivalent product or part to replace the defective item, or refund to Customer the purchase price paid for the defective product. All products that are replaced will become the property of 3Com. Replacement products may be new or reconditioned. Any replaced or repaired product or part has a ninety (90) day warranty or the remainder of the initial warranty period, whichever is longer.

3Com shall not be responsible for any software, firmware, information, or memory data of Customer contained in, stored on, or integrated with any products returned to 3Com for repair, whether under warranty or not.

SOFTWARE

3Com warrants that the software programs licensed from it will perform in substantial conformance to the program specifications therefor for a period of ninety (90) days from the date of purchase from 3Com or its Authorized Reseller. 3Com warrants the media containing software against failure during the warranty period. No updates are provided. The sole obligation of 3Com with respect to this express warranty shall be (at the discretion of 3Com) to refund the purchase price paid by Customer for any defective software products, or to replace any defective media with software which substantially conforms to applicable 3Com published specifications. Customer assumes responsibility for the selection of the appropriate applications program and associated reference materials. 3Com makes no warranty or representation that its software products will work in combination with any hardware or applications software products provided by third parties, that the operation of the software products will be uninterrupted or error free, or that all defects in the software products will be corrected. For any third-party products listed in the 3Com software product documentation or specifications as being compatible, 3Com will make reasonable efforts to provide compatibility, except where the noncompatibility is caused by a "bug" or defect in the third party's product.

STANDARD WARRANTY SERVICE

Standard warranty service for *hardware* products may be obtained by delivering the defective product, accompanied by a copy of the dated proof of purchase, to the 3Com Corporate Service Center or to an Authorized 3Com Service Center during the applicable warranty period. Standard warranty service for *software* products may be obtained by telephoning the 3Com Corporate Service Center or an Authorized 3Com Service Center, within the warranty period. Products returned to the 3Com Corporate Service Center must be preauthorized by 3Com with a Return Material Authorization (RMA) number marked on the outside of the package, and sent prepaid, insured, and packaged appropriately for safe shipment. The repaired or replaced item will be shipped to Customer, at the expense of 3Com, not later than thirty (30) days after receipt of the defective product by 3Com.

WARRANTIES EXCLUSIVE

IF A 3COM PRODUCT DOES NOT OPERATE AS WARRANTED ABOVE, CUSTOMER'S SOLE REMEDY FOR BREACH OF THAT WARRANTY SHALL BE REPAIR, REPLACEMENT, OR REFUND OF THE PURCHASE PRICE PAID, AT THE OPTION OF 3COM. TO THE FULL EXTENT ALLOWED BY LAW, THE FOREGOING WARRANTIES AND REMEDIES ARE EXCLUSIVE AND ARE IN LIEU OF ALL OTHER WARRANTIES, TERMS, OR CONDITIONS, EXPRESS OR IMPLIED, EITHER IN FACT OR BY OPERATION OF LAW, STATUTORY OR OTHERWISE, INCLUDING WARRANTIES, TERMS, OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND SATISFACTORY QUALITY. 3COM NEITHER ASSUMES NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR IT ANY OTHER LIABILITY IN CONNECTION WITH THE SALE, INSTALLATION, MAINTENANCE, OR USE OF ITS PRODUCTS.

3COM SHALL NOT BE LIABLE UNDER THIS WARRANTY IF ITS TESTING AND EXAMINATION DISCLOSE THAT THE ALLEGED DEFECT IN THE PRODUCT DOES NOT EXIST OR WAS CAUSED BY CUSTOMER'S OR ANY THIRD PERSON'S MISUSE, NEGLECT, IMPROPER INSTALLATION OR TESTING, UNAUTHORIZED ATTEMPTS TO REPAIR OR MODIFY, OR ANY OTHER CAUSE BEYOND THE RANGE OF THE INTENDED USE, OR BY ACCIDENT, FIRE, LIGHTNING, OR OTHER HAZARD.

LIMITATION OF LIABILITY

TO THE FULL EXTENT ALLOWED BY LAW, 3COM ALSO EXCLUDES FOR ITSELF AND ITS SUPPLIERS ANY LIABILITY, WHETHER BASED IN CONTRACT OR TORT (INCLUDING NEGLIGENCE), FOR INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL, OR PUNITIVE DAMAGES OF ANY KIND, OR FOR LOSS OF REVENUE OR PROFITS, LOSS OF BUSINESS, LOSS OF INFORMATION OR DATA, OR OTHER FINANCIAL LOSS ARISING OUT OF OR IN CONNECTION WITH THE SALE, INSTALLATION, MAINTENANCE, USE, PERFORMANCE, FAILURE, OR INTERRUPTION OF ITS PRODUCTS, EVEN IF 3COM OR ITS AUTHORIZED RESELLER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, AND LIMITS ITS LIABILITY TO REPAIR, REPLACEMENT, OR REFUND OF THE PURCHASE PRICE PAID, AT THE OPTION OF 3COM. THIS DISCLAIMER OF LIABILITY FOR DAMAGES WILL NOT BE AFFECTED IF ANY REMEDY PROVIDED HEREIN SHALL FAIL OF ITS ESSENTIAL PURPOSE.

Some countries, states, or provinces do not allow the exclusion or limitation of implied warranties or the limitation of incidental or consequential damages for certain products supplied to consumers or the limitation for personal injury, so the above limitations and exclusions may be limited in their application to you. This warranty gives you specific legal rights which may vary depending on local law.

GOVERNING LAW

This Limited Warranty shall be governed by the laws of the State of California.

3Com Corporation, 5400 Bayfront Plaza, Santa Clara, CA 95052-8145 (408) 764-5000

3/6/96