

NCU USER GUIDE

VOLUME II: DETAILED NETWORK DEFINITION

NCU User Guide

Volume II: Detailed Network Definition

Wellfleet Communications, Inc.

Copyright © 1991 Wellfleet Communications, Inc.

All Rights Reserved

Printed in USA

August, 1991

Please address questions about technical matters to our 24-Hour Customer Support Line:

Inside Massachusetts: 617-275-2400

Outside Massachusetts: 1-800-2LANWAN

Please address comments about this manual to:

Technical Publications

Wellfleet Communications, Inc.

15 Crosby Drive

Bedford, MA 01730

Tel: 617-275-2400

Fax: 617-275-5001

Information presented in this document is subject to change without notice.

SYBASE™ is a registered trademark of Sybase, Inc.

AppleTalk is a registered trademark of Apple Computer, Inc.

DECnet, VAX, and VT-100 are registered trademarks of Digital Equipment Corporation.

Ethernet and XNS are registered trademarks of Xerox Corporation.

IPX is a registered trademark of Novell, Inc.

MS-DOS is a registered trademark of Microsoft Corporation.

Sun Workstation, SPARC Station, Sun OS, and OpenWindows are registered trademarks of Sun Microsystems, Inc.

UNIX is a registered trademark of AT&T Bell Laboratories.

X Window System is a registered trademark of the Massachusetts Institute of Technology.

Table of Contents

Preface	xvi
Purpose of this Guide	xvi
Audience	xvi
Organization	xvi
Associated Documents	xvii
Related Documents	xvii
Required Hardware and Software	xviii
Document Conventions	xviii
1 Using the Detailed Configuration Editor	1-1
1.1 Accessing the Configuration-Default Parameters in NCU	1-1
1.2 Accessing the Configuration Parameters of a Single Node	1-2
2 Editing System Parameters	2-1
2.1 Accessing System Parameters	2-1
2.2 Editing Global Parameters	2-1
2.2.1 Adding Global Parameters	2-2
2.2.2 Changing Global Parameters	2-3
2.2.3 Deleting Global Parameters	2-5
2.3 Editing Session Parameters	2-6
2.3.1 Editing User-Session Parameters	2-6
2.3.1.1 Adding a User Session	2-7
2.3.1.2 Changing a User Session	2-7
2.3.1.3 Deleting a User Session	2-10
2.3.2 Editing Printer-Session Parameters	2-11
2.3.2.1 Adding a Printer Session	2-11
2.3.2.2 Changing a Printer Session	2-11
2.3.2.3 Deleting a Printer Session	2-14
2.3.3 Editing Telnet-Session Parameters	2-15
2.3.3.1 Adding a Telnet Session	2-15
2.3.3.2 Changing a Telnet Session	2-15
2.3.3.3 Deleting a Telnet Session	2-17

2.3.4	Editing Disk-Log Session Parameters	2-18
2.3.4.1	Adding a Disk-Log Session	2-18
2.3.4.2	Changing a Disk-Log Session	2-18
2.3.4.3	Deleting a Disk-Log Session	2-21
3	Editing Circuits	3-1
3.1	Accessing Node Circuit Parameters	3-1
3.2	Editing Ethernet/LAN-Circuit Parameters	3-4
3.3	Editing Token-Ring/LAN-Circuit Parameters	3-6
3.4	Editing FDDI/LAN-Circuit Parameters.....	3-9
3.5	Editing Synchronous-Line Parameters	3-12
3.6	Editing T1-Line Parameters	3-15
3.7	Editing E1-Line Parameters	3-17
3.8	Editing Point-to-Point Circuit Parameters	3-20
3.9	Editing X.25 Circuit Parameters	3-26
3.9.1	Editing X.25 Point-to-Point Service Parameters	3-27
3.9.1.1	Configuring Point-To-Point Virtual Circuits	3-31
3.9.1.1.1	Adding Point-To-Point Virtual Circuits	3-31
3.9.1.1.2	Modifying Point-To-Point Virtual Circuits	3-34
3.9.1.1.3	Deleting Point-To-Point Virtual Circuits	3-34
3.9.2	Editing X.25 DDN Service Parameters	3-34
3.9.3	Editing X.25 PDN Service Parameters	3-39
3.9.3.1	Building the X.25 PDN Address Map	3-44
3.9.3.1.1	Adding Map Entries	3-44
3.9.3.1.2	Editing Map Entries	3-48
3.9.3.1.3	Deleting Map Entries	3-48
3.9.4	Configuring Bitmap Values	3-48
4	Editing Circuit Groups	4-1
4.1	Accessing Circuit Group Parameters.....	4-2
4.1.1	Adding Circuit Groups	4-3
4.1.2	Modifying Circuit Groups	4-7
4.1.3	Deleting Circuit Groups	4-9
5	Editing TCP/IP Parameters	5-1
5.1	TCP/IP Overview	5-1
5.2	Accessing IP Parameters.....	5-3
5.3	Editing IP Basic Parameters	5-4
5.3.1	Adding IP Basic Parameters	5-4
5.3.2	Deleting IP Basic Parameters	5-9
5.4	Editing IP Interfaces.....	5-9

5.4.1	Changing Interface-Specific Parameters	5-9
5.4.1.1	Editing Port Filters	5-17
5.4.1.1.1	Adding Port Filters	5-18
5.4.1.1.2	Deleting Port Filters	5-21
5.4.1.1.3	Updating Port Filters	5-23
5.4.2	Deleting IP Interface-Specific Parameters	5-23
5.5	Editing Routing Parameters	5-24
5.5.1	Editing RIP Parameters	5-24
5.5.1.1	Adding RIP Basic Parameters	5-24
5.5.1.2	Updating RIP Basic Parameters	5-26
5.5.1.3	Deleting RIP Basic Parameters	5-26
5.5.1.4	Adding RIP Interfaces	5-26
5.5.1.5	Updating RIP Interfaces	5-30
5.5.1.6	Deleting RIP Interfaces	5-30
5.5.2	Editing EGP Parameters	5-30
5.5.2.1	Adding EGP Basic Parameters	5-30
5.5.2.2	Updating EGP Basic Parameters	5-32
5.5.2.3	Deleting EGP Basic Parameters	5-32
5.5.2.4	Adding EGP Neighbors	5-32
5.5.2.5	Updating EGP Neighbors	5-35
5.5.2.6	Deleting EGP Neighbors	5-35
5.5.3	Editing OSPF Parameters	5-35
5.5.3.1	Adding OSPF Basic Parameters	5-38
5.5.3.2	Updating OSPF Basic Parameters	5-40
5.5.3.3	Deleting OSPF Basic Parameters	5-40
5.5.3.4	Configuring OSPF Areas and Backbone Connections	5-41
5.5.3.4.1	Adding Backbone Connections	5-41
5.5.3.4.1.1	Adding Backbone Networks	5-42
5.5.3.4.1.2	Adding Backbone Virtual Links	5-45
5.5.3.4.1.3	Updating Backbone Virtual Links	5-47
5.5.3.4.1.4	Deleting Backbone Virtual Links	5-47
5.5.3.4.2	Adding OSPF Areas	5-48
5.5.3.4.2.1	Adding Area Networks	5-49
5.5.3.4.3	Adding OSPF Interfaces	5-49
5.5.3.4.4	Updating OSPF Interfaces	5-53
5.5.3.4.5	Deleting OSPF Interfaces	5-53
5.5.3.5	Updating OSPF Areas	5-53
5.5.3.6	Deleting OSPF Areas	5-53
5.5.4	Editing Static and Default Routes	5-54
5.5.4.1	Adding Static Routes	5-56
5.5.4.2	Adding Default Routes	5-59
5.5.5	Updating Static and Default Routes	5-60
5.5.6	Deleting Static and Default Routes	5-60
5.5.7	Editing Adjacent Host Routes	5-61

5.5.7.1	Adding Adjacent Host Routes	5-62
5.5.7.2	Updating Adjacent Host Routes	5-65
5.5.7.3	Deleting Adjacent Host Routes	5-65
5.5.8	Configuring Filters	5-66
5.5.8.1	Configuring Address Filters	5-66
5.5.8.1.1	Adding Address Filters	5-66
5.5.8.1.2	Updating Address Filters	5-69
5.5.8.1.3	Deleting Address Filters	5-69
5.5.8.2	Configuring Routing-Pool Filters	5-69
5.5.8.2.1	Adding Import-Route Filters	5-72
5.5.8.2.1.1	Adding a RIP Import Filter	5-73
5.5.8.2.1.2	Adding an EGP Import Filter	5-75
5.5.8.2.1.3	Adding an OSPF Import Filter	5-76
5.5.8.2.2	Updating Import-Route Filters	5-79
5.5.8.2.3	Deleting Import-Route Filters	5-79
5.5.8.2.4	Adding Export-Route Filters	5-79
5.5.8.2.4.1	Adding a RIP Export Filter	5-80
5.5.8.2.4.2	Adding an EGP Export Filter	5-82
5.5.8.2.4.3	Adding an OSPF Export Filter	5-84
5.5.8.2.5	Updating Export-Route Filters	5-86
5.5.8.2.6	Deleting Export-Route Filters	5-86
5.6	Configuring IP Applications.....	5-87
5.6.1	Configuring TFTP	5-88
5.6.2	Configuring the SNMP Agent	5-90
5.6.2.1	Adding SNMP Communities	5-91
5.6.2.1.1	Adding SNMP Community Members	5-93
5.6.2.1.2	Deleting SNMP Community Members	5-94
5.6.2.2	Updating SNMP Communities	5-94
5.6.2.3	Deleting SNMP Communities	5-95
5.6.3	Configuring BOOTP	5-95
5.6.3.1	Adding the BOOTP Server	5-95
5.6.3.1.1	Adding the BOOTP Client	5-97
5.6.3.1.2	Updating the BOOTP Client	5-99
5.6.3.1.3	Deleting the BOOTP Client	5-99
5.6.3.2	Updating the BOOTP Server	5-99
5.6.3.3	Deleting the BOOTP Server	5-100
5.7	Configuring TCP	5-100
6	Editing Bridge Parameters	6-1
6.1	Bridge Overview.....	6-1
6.1.1	Transparent Bridges	6-2
6.1.2	Source-Routing Bridges	6-3
6.1.2.1	How Source Routing Works	6-4

6.1.3	Source-Routing/Transparent Bridges	6-8
6.1.4	Spanning-Tree Algorithm	6-9
6.1.5	Filtering	6-12
6.2	Accessing Bridge Parameters.....	6-14
6.3	Editing Bridge Basic Parameters	6-15
6.3.1	Modifying Bridge Basic Parameters	6-16
6.3.1.1	Setting Spanning Tree Parameters.	6-18
6.3.2	Deleting Bridge Basic Parameters	6-21
6.4	Configuring Bridge Interfaces	6-21
6.4.1	Modifying Bridge Interfaces	6-21
6.4.1.1	Configuring Filters	6-23
6.4.1.1.1	Adding MAC-Level Source and Destination Address Filters	6-25
6.4.1.1.2	Adding Ethernet Filters	6-31
6.4.1.1.3	Adding 802.2 LLC Filters	6-37
6.4.1.1.4	Adding 802.2 SNAP Filters	6-45
6.4.1.1.5	Adding Novell Filters	6-52
6.4.1.1.6	Adding User-Defined Filters	6-57
6.4.1.1.7	Constructing Filter Lists	6-66
6.4.1.1.7.1	Constructing MAC Address Filter Lists	6-69
6.4.1.1.7.2	Constructing Ethernet Type Filter Lists	6-71
6.4.1.1.7.3	Constructing SAP Filter Lists	6-72
6.4.1.1.7.4	Constructing Protocol ID/Organization Code Filter Lists	6-74
6.4.1.1.7.5	Modifying Filter Lists	6-76
6.4.1.1.7.6	Deleting Filter Lists	6-76
6.4.1.1.8	Forwarding Filtered Traffic	6-76
6.4.1.1.9	Deleting Filters	6-78
6.4.1.2	Configuring the Load-Balance Option	6-78
6.4.2	Deleting Bridge Interfaces	6-81
7	Editing DECnet Parameters	7-1
7.1	DECnet Overview	7-1
7.2	Accessing DECnet Parameters	7-2
7.3	Editing DECnet Basic Parameters	7-3
7.3.1	Modifying DECnet Basic Parameters	7-3
7.3.2	Deleting DECnet Basic Parameters	7-8
7.4	Configuring DECnet Interfaces	7-8
7.4.1	Modifying DECnet Interfaces	7-8
7.4.2	Deleting DECnet Interfaces	7-10
8	Editing XNS Parameters	8-1
8.1	XNS Overview.....	8-1
8.2	Accessing XNS Parameters.....	8-6

8.3	Editing XNS Basic Parameters	8-7
8.3.1	Modifying XNS Basic Parameters	8-7
8.3.2	Deleting XNS Basic Parameters	8-8
8.4	Configuring XNS Interfaces	8-8
8.4.1	Modifying XNS Interfaces	8-8
8.4.2	Deleting XNS Interfaces	8-10
8.5	Configuring Static Routes	8-10
8.5.1	Adding Static Routes	8-11
8.5.2	Updating Static Routes	8-12
8.5.3	Deleting Static Routes	8-13
9	Editing IPX Parameters	9-1
9.1	IPX Overview	9-1
9.1.1	Service Advertising Protocol	9-4
9.1.2	NetBIOS	9-5
9.2	Accessing IPX Parameters	9-6
9.3	Editing IPX Basic Parameters	9-7
9.3.1	Modifying IPX Basic Parameters	9-7
9.3.2	Deleting IPX Basic Parameters	9-8
9.4	Configuring IPX Interfaces	9-8
9.4.1	Modifying IPX Interfaces	9-8
9.4.2	Configuring Interface-Specific Filters and NetBIOS Static Routes	9-14
9.4.2.1	Configuring SAP Filters	9-14
9.4.2.1.1	Configuring SAP Network-Level Filters	9-15
9.4.2.1.1.1	Adding SAP Network-Level Filters	9-16
9.4.2.1.1.2	Updating SAP Network-Level Filters	9-17
9.4.2.1.1.3	Deleting SAP Network-Level Filters	9-17
9.4.2.1.2	Configuring SAP Server-Level Filters	9-17
9.4.2.1.2.1	Adding SAP Server-Level Filters	9-17
9.4.2.1.2.2	Updating SAP Server-Level Filters	9-19
9.4.2.1.2.3	Deleting SAP Server-Level Filters	9-19
9.4.2.2	Configuring NetBIOS Static Routes	9-20
9.4.2.2.1	Adding NetBIOS Static Routes	9-21
9.4.2.3	Updating NetBIOS Static Routes	9-22
9.4.2.4	Deleting NetBIOS Static Routes	9-23
9.4.3	Deleting IPX Interfaces	9-23
9.5	Configuring Static Routes	9-23
9.5.1	Adding Static Routes	9-23
9.5.2	Updating Static Routes	9-25
9.5.3	Deleting Static Routes	9-25

10	Editing AppleTalk Parameters	10-1
10.1	AppleTalk Overview	10-1
10.2	Accessing AppleTalk Parameters	10-6
10.3	Editing AppleTalk Basic Parameters.....	10-6
10.3.1	Modifying AppleTalk Basic Parameters	10-7
10.3.2	Deleting AppleTalk Basic Parameters	10-9
10.4	Configuring AppleTalk Interfaces.....	10-9
10.4.1	Modifying AppleTalk Interfaces	10-9
10.4.1.1	Configuring Seed Routers	10-13

List of Figures

Figure 1-1.	WELLFLEET NCU Window	1-2
Figure 1-2.	EDIT NODE CONFIGURATION Window for Default Settings	1-3
Figure 1-3.	CONFIGURATION NODE LIST Window for NEUSCurrent	1-4
Figure 1-4.	EDIT NODE CONFIGURATION Window for BOS	1-5
Figure 2-1.	EDIT NODE CONFIGURATION Window for Default Settings	2-2
Figure 2-2.	GLOBAL PARAMETERS Window for DEFAULT_NODE	2-3
Figure 2-3.	Software Version Values List	2-4
Figure 2-4.	USER SESSION Window for BOS	2-8
Figure 2-5.	PRINTER SESSION Window	2-12
Figure 2-6.	TELNET SESSION Window	2-16
Figure 2-7.	LOG SESSION Window	2-19
Figure 3-1.	EDIT NODE CONFIGURATION Window for a Single Node	3-2
Figure 3-2.	CIRCUITS Window with Ethernet Connector Selected	3-3
Figure 3-3.	LAN CIRCUIT CONFIGURATION Window	3-5
Figure 3-4.	TOKEN CIRCUIT CONFIGURATION Window	3-7
Figure 3-5.	FDDI CIRCUIT CONFIGURATION Window	3-10
Figure 3-6.	SYNC LINE Window	3-13
Figure 3-7.	T1 LINE Window	3-16
Figure 3-8.	E1 LINE Window	3-18
Figure 3-9.	POINT TO POINT CIRCUIT Window	3-21
Figure 3-10.	HDLC Frame Format	3-22
Figure 3-11.	Satellite Broadcast (Sample Topology)	3-24
Figure 3-12.	X.25 POINT TO POINT Window	3-27
Figure 3-13.	X.25 VIRTUAL CIRCUITS Window	3-31
Figure 3-14.	ADD X.25 VIRTUAL CIRCUIT Window	3-32
Figure 3-15.	X.25 DDN Window	3-35
Figure 3-16.	X.25 PDN Window	3-41
Figure 3-17.	X.25 PDN ADDRESS MAP Window	3-45
Figure 3-18.	ADD PDN ADDRESS MAP Window	3-45
Figure 3-19.	X.25 BITMAP VALUES Window	3-49

Figure 4-1.	Multiple Circuit Group Assignment	4-2
Figure 4-2.	EDIT NODE CONFIGURATION Window for a Single Node	4-3
Figure 4-3.	CIRCUIT GROUPS Window	4-4
Figure 4-4.	NAME CIRCUIT GROUP Window	4-4
Figure 4-5.	SELECT NETWORK Window	4-5
Figure 4-6.	CIRCUIT GROUP Window	4-6
Figure 4-7.	CIRCUITS Window	4-7
Figure 4-8.	RENAME CIRCUIT GROUP Window	4-8
Figure 5-1.	The Four Layers in the TCP/IP Model	5-3
Figure 5-2.	EDIT NODE CONFIGURATION Window for Default Settings	5-5
Figure 5-3.	NODE IP CONFIGURATION Window	5-6
Figure 5-4.	END-NODE Operation	5-8
Figure 5-5.	Virtual Host Configuration	5-8
Figure 5-6.	IP INTERFACE DEFINITION Window	5-10
Figure 5-7.	TCP Segment Format	5-17
Figure 5-8.	UDP Datagram Format	5-18
Figure 5-9.	IP PORT FILTERS Window	5-20
Figure 5-10.	ADD PORT FILTER Window	5-20
Figure 5-11.	IP PORT FILTERS Window with Port Selected	5-22
Figure 5-12.	NODE IP CONFIGURATION Window	5-23
Figure 5-13.	RIP Window	5-25
Figure 5-14.	RIP Window with Interface Selected	5-27
Figure 5-15.	RIP INTERFACE DEFINITION Window	5-28
Figure 5-16.	EGP Window	5-31
Figure 5-17.	EGP NEIGHBORS window	5-34
Figure 5-18.	Sample OSPF Topology	5-37
Figure 5-19.	Routing Hierarchy	5-40
Figure 5-20.	OSPF Window	5-42
Figure 5-21.	OSPF AREA Window	5-43
Figure 5-22.	OSPF NETWORKS Window	5-44
Figure 5-23.	ADD OSPF NETWORK Window	5-44
Figure 5-24.	OSPF VIRTUAL LINK Window	5-46
Figure 5-25.	ADD VIRTUAL LINK Window	5-47
Figure 5-26.	OSPF INTERFACES Window with Interface Selected	5-50
Figure 5-27.	OSPF INTERFACE Window	5-51
Figure 5-28.	Sample Default Route Topology	5-55
Figure 5-29.	STATIC ROUTES Window	5-56
Figure 5-30.	STATIC ROUTE DEFINITION Window	5-57
Figure 5-31.	STATIC ROUTES Window with Route Selected	5-61
Figure 5-32.	ADJACENT HOST ROUTES Window	5-62
Figure 5-33.	ADJACENT HOST ROUTE Window	5-63
Figure 5-34.	IP Ethernet Encapsulation	5-64

Figure 5-35.	IP 802.2 Encapsulation	5-64
Figure 5-36.	IP SNAP Encapsulation	5-65
Figure 5-37.	ADDRESS FILTERS Window	5-67
Figure 5-38.	ADD ADDRESS FILTER Window	5-68
Figure 5-39.	Routing Information Data Flow	5-71
Figure 5-40.	IMPORT ROUTE FILTERS Window	5-72
Figure 5-41.	IMPORT FROM RIP Window	5-74
Figure 5-42.	IMPORT FROM EGP Window	5-76
Figure 5-43.	IMPORT FROM OSPF Window	5-77
Figure 5-44.	EXPORT ROUTE FILTERS Window	5-80
Figure 5-45.	EXPORT ROUTE TO RIP Window	5-81
Figure 5-46.	EXPORT ROUTE TO EGP Window	5-83
Figure 5-47.	EXPORT ROUTE TO OSPF Window	5-85
Figure 5-48.	TFTP Window	5-89
Figure 5-49.	SNMP Window	5-90
Figure 5-50.	SNMP COMMUNITY Window	5-92
Figure 5-51.	SNMP ACCESS Window	5-94
Figure 5-52.	BOOTP Window	5-96
Figure 5-53.	BOOTP CLIENT Window	5-98
Figure 5-54.	TCP Window	5-100
Figure 6-1.	SRT Bridge Routing Designators	6-5
Figure 6-2.	Multi-Ring Source-Routed Network	6-6
Figure 6-3.	Routing Designator 1	6-6
Figure 6-4.	Routing Designator 2	6-7
Figure 6-5.	Routing Designator 3	6-7
Figure 6-6.	Sample SRT Topology	6-8
Figure 6-7.	Parallel Bridge Topology	6-10
Figure 6-8.	Spanning Tree (Loop-Free) Logical Topology	6-12
Figure 6-9.	Ethernet Encapsulation	6-12
Figure 6-10.	802.2 Encapsulation	6-13
Figure 6-11.	SNAP Encapsulation	6-13
Figure 6-12.	Novell Proprietary Encapsulation	6-13
Figure 6-13.	EDIT NODE CONFIGURATION Window for Default Settings	6-15
Figure 6-14.	BRIDGE Window with Spanning Tree Enable set to NO	6-17
Figure 6-15.	BRIDGE Window with Spanning Tree Enable set to YES	6-19
Figure 6-16.	BRIDGE CIRCUIT GROUPS Window	6-22
Figure 6-17.	BRIDGE INTERFACE FILTERS Window	6-24
Figure 6-18.	BRIDGE FILTER Window	6-26
Figure 6-19.	MAC ADDRESS FILTER Window	6-28
Figure 6-20.	ETHERNET TYPE FILTER Window	6-36
Figure 6-21.	802.2 LLC FILTER Window	6-42
Figure 6-22.	802.2 SNAP FILTER Window	6-50

Figure 6-23.	MAC ADDRESS FILTER Window	6-59
Figure 6-24.	USER DEFINED FIELDS Window.....	6-63
Figure 6-25.	USER DEFINED FILTER Window	6-64
Figure 6-26.	FILTER VALUES Window	6-65
Figure 6-27.	ADD FILTER USER VALUE Window	6-65
Figure 6-28.	BRIDGE FILTER LISTS Window.....	6-68
Figure 6-29.	LIST NAME Window	6-69
Figure 6-30.	MAC ADDRESS LIST Window.....	6-70
Figure 6-31.	ADD MAC ADDRESS RANGE Window	6-70
Figure 6-32.	ETHERNET TYPE LIST Window.....	6-71
Figure 6-33.	ETHERNET TYPE RANGE Window	6-72
Figure 6-34.	SAP LIST Window	6-73
Figure 6-35.	SAP RANGE Window	6-74
Figure 6-36.	PROTOCOL/ORG ID LIST Window	6-75
Figure 6-37.	PROTOCOL/ORG ID RANGE Window	6-75
Figure 6-38.	BRIDGE FORWARD TABLE Window	6-77
Figure 6-39.	BRIDGE GROUPS Window	6-77
Figure 6-40.	BRIDGE LOAD BALANCE Window	6-79
Figure 6-41.	SELECT CIRCUIT Window	6-80
Figure 6-42.	SELECT PROTOCOL Window.....	6-80
Figure 7-1.	EDIT NODE CONFIGURATION Window for Default Settings	7-3
Figure 7-2.	DECNET REDIRECTOR Window.....	7-5
Figure 7-3.	Sample DECnet Circuit Costs	7-7
Figure 7-4.	DECNET INTERFACE Window	7-9
Figure 8-1.	XNS Protocol Architecture	8-3
Figure 8-2.	XNS Internet Packet Format	8-5
Figure 8-3.	EDIT NODE CONFIGURATION Window for DEFAULT_NODE	8-6
Figure 8-4.	XNS REDIRECTOR Window	8-7
Figure 8-5.	XNS INTERFACE Window	8-9
Figure 8-6.	XNS STATIC ROUTES Window	8-11
Figure 8-7.	ADD XNS STATIC ROUTE Window	8-12
Figure 9-1.	IPX Internet Packet Format	9-3
Figure 9-2.	EDIT NODE CONFIGURATION Window for DEFAULT_NODE	9-6
Figure 9-3.	IPX REDIRECTOR Window	9-7
Figure 9-4.	IPX INTERFACE Window.....	9-9
Figure 9-5.	Ethernet Encapsulation.....	9-10
Figure 9-6.	Novell Proprietary Encapsulation	9-11
Figure 9-7.	802.2 Encapsulation	9-11

Figure 9-8.	Sample IPX Internet.....	9-12
Figure 9-9.	IPX SAP NETWORK FILTERS Window.....	9-15
Figure 9-10.	ADD SAP NETWORK FILTER Window.....	9-16
Figure 9-11.	SAP SERVER LEVEL FILTERS Window.....	9-18
Figure 9-12.	ADD SAP SERVER FILTER Window.....	9-18
Figure 9-13.	NETBIOS BROADCAST STATIC ROUTES Window.....	9-21
Figure 9-14.	ADD NETBIOS STATIC ROUTES Window.....	9-22
Figure 9-15.	IPX STATIC ROUTES Window.....	9-24
Figure 9-16.	ADD IPX STATIC ROUTE Window.....	9-24
Figure 10-1.	AppleTalk Local Router.....	10-2
Figure 10-2.	AppleTalk Half Router.....	10-2
Figure 10-3.	AppleTalk Backbone Router.....	10-3
Figure 10-4.	Layered Model of AppleTalk Routing Protocols.....	10-4
Figure 10-5.	EDIT NODE CONFIGURATION Window for DEFAULT_NODE	10-7
Figure 10-6.	APPLETALK Window.....	10-8
Figure 10-7.	APPLETALK INTERFACE Window with Seed Router Set to YES	10-10
Figure 10-8.	APPLETALK INTERFACE Window with Seed Router Set to NO	10-11
Figure 10-9.	APPLETALK ZONE Window.....	10-15

List of Tables

Table 3-1.	X.25 Bitmap Values	3-49
Table 5-1.	Military-Standard Protocols	5-2
Table 5-2.	Internet Request for Comments (RFCs) for the IP Router	5-4
Table 5-3.	TCP/UDP Well-Known Ports	5-19
Table 5-4.	OSPF Routers	5-36
Table 5-5.	Import- and Export-Filter Configuration Rules	5-70
Table 6-1.	Encapsulation/Media Matrix	6-14
Table 6-2.	Pre-defined Filter Fields	6-14
Table 6-3.	Suggested Spanning Tree Parameter Values	6-20
Table 6-4.	Public Ethernet Type Field Values	6-67
Table 7-1.	Suggested DECnet Circuit Costs	7-6
Table 8-1.	XNS Packet-Type Values	8-5
Table 8-2.	XNS Well-Known Sockets	8-5
Table 9-1.	IPX Packet-Type Values	9-4
Table 9-2.	IPX Well-Known Sockets	9-5
Table 10-1.	Functions of AppleTalk Protocols	10-5
Table 10-2.	Probe Implementation for Non-Seed Routers	10-12
Table 10-3.	Probe Implementation for Seed Routers	10-12

Preface

Purpose of this Guide

This guide describes how to install and use Wellfleet's network-management system. *Network Configuration Utility (NCU), Version 1.0* allows you to configure an entire network of Wellfleet bridges and routers from a single software platform.

Audience

Volume I of this guide, *Simplified Network Definition*, is intended for untrained network operators that do not have a detailed understanding of bridging and routing technologies. Volume I describes how to configure a basic network.

Volume II of this guide, *Detailed Network Definition*, is intended for experienced network managers. Volume II describes how to set optional parameters associated with more complex networks.

Organization

This guide consists of two volumes, as follows:

- *Volume I: Simplified Network Definition* is organized, as follows:
 - Chapter 1 introduces the capabilities of the NCU program.
 - Chapter 2 describes how to install NCU.
 - Chapter 3 describes the NCU user interface.
 - Chapter 4 describes how to maintain network configurations.
 - Chapter 5 describes how to define network nodes.
 - Chapter 6 describes how to generate configuration reports.
 - Appendix A lists the configuration default settings in NCU.
 - Appendix B describes how to use SYBASE Transact-SQL™ commands.
 - Appendix C describes how to use SYBASE SQL Server™ system procedures.
 - Appendix D provides information about the system tables in NCU.

- Appendix E describes how to use certain UNIX®-level SYBASE utility programs.
- Appendix F provides diagrams of database tables.

- *Volume II: Detailed Network Definition* is organized, as follows:
 - Chapter 1 describes how to use the detailed configuration editor
 - Chapter 2 describes how to edit system parameters.
 - Chapter 3 describes how to edit circuits.
 - Chapter 4 describes how to edit circuit groups.
 - Chapter 5 describes how to edit TCP/IP parameters.
 - Chapter 6 describes how to edit Bridge parameters.
 - Chapter 7 describes how to edit DECnet parameters.
 - Chapter 8 describes how to edit XNS parameters.
 - Chapter 9 describes how to edit IPX parameters.
 - Chapter 10 describes how to edit AppleTalk parameters.

Associated Documents

This manual references the Wellfleet internetworking documentation set. It also references the following Sybase documents which you should have received with NCU:

- *Sun/UNIX Supplementals*
- *SYBASE Installation Guide for Sun OS*
- *SYBASE Installation Guide for Sun UNIX*
- *SQL Toolset System-Specific Notes for Sun OS*

Related Documents

This section lists the Wellfleet internetworking and network-management document sets and related technical documents:

- Internetworking Documentation Set
 - *Configuration Guide*, Volume I (Part # 101157G) and Volume II (Part # 102938A)

Describes how to configure a network by describing the network to the node; includes information on configuring the network's current topology and future changes.

-
- *Installation Guide* (Part # 101159)
Describes how to install the node hardware and how to boot the system.
 - *Operator's Guide* (Part # 101160)
Describes how to operate the node. It includes information on the Network Control Language (NCL) Interpreter and the system information base, and it lists all event messages generated during system operations.
 - Network-Management Document Set
 - *SNMP-NMS User's Guide* (Part # 101161)
Describes how to install and use Wellfleet's SNMP-base network-management software.
 - *NCU User's Guide*, Volume I (Part # 103098) and Volume II (Part # 103239)
Describes how to install Wellfleet's Network Configuration Utility, which allows you to configure remotely all Wellfleet nodes in your network from a central management station.

Required Hardware and Software

NCU runs under the X Window System™ or OpenWindows™ environments on a Sun Microsystems' Sun 3 or SPARC workstation configured with at least 8MBytes of RAM and at least 40 MBytes free disk space.

Document Conventions

This manual uses the following document conventions:

Convention	Function
<i>config.dat</i>	Denotes UNIX® files and directory names
WELLFLEET NCU	Helvetica, bold characters denote window names, window fields, and parameter settings.
[RETURN]	Courier, bold characters enclosed in brackets denote keyboard characters.

1 Using the Detailed Configuration Editor

NCU provides two mechanisms for editing configuration parameters:

1. One mechanism () allows you to edit the configuration-default parameters in NCU (Appendix A in *Volume 1: Simplified Network Definition* of the *NCU User Guide* lists the default settings for configuration parameters).

The changes that you make using *affect every node you configure afterwards.*

2. The other mechanism () allows you to edit the configuration parameters of a single node.

The changes that you make using *affect only that node.*

Note

When you first install the program, you may wish to change the defaults in NCU (by using the mechanism); however, for the most part, you use the mechanism to access configuration parameters.

This section describes how to access each mechanism.

1.1 Accessing the Configuration-Default Parameters in NCU

You access the configuration-default parameters in NCU, as follows:

1. Select in the WELLFLEET NCU window to display the menu depicted in Figure 1-1.
2. Select to display the EDIT NODE CONFIGURATION window for the configuration-default parameters in NCU.

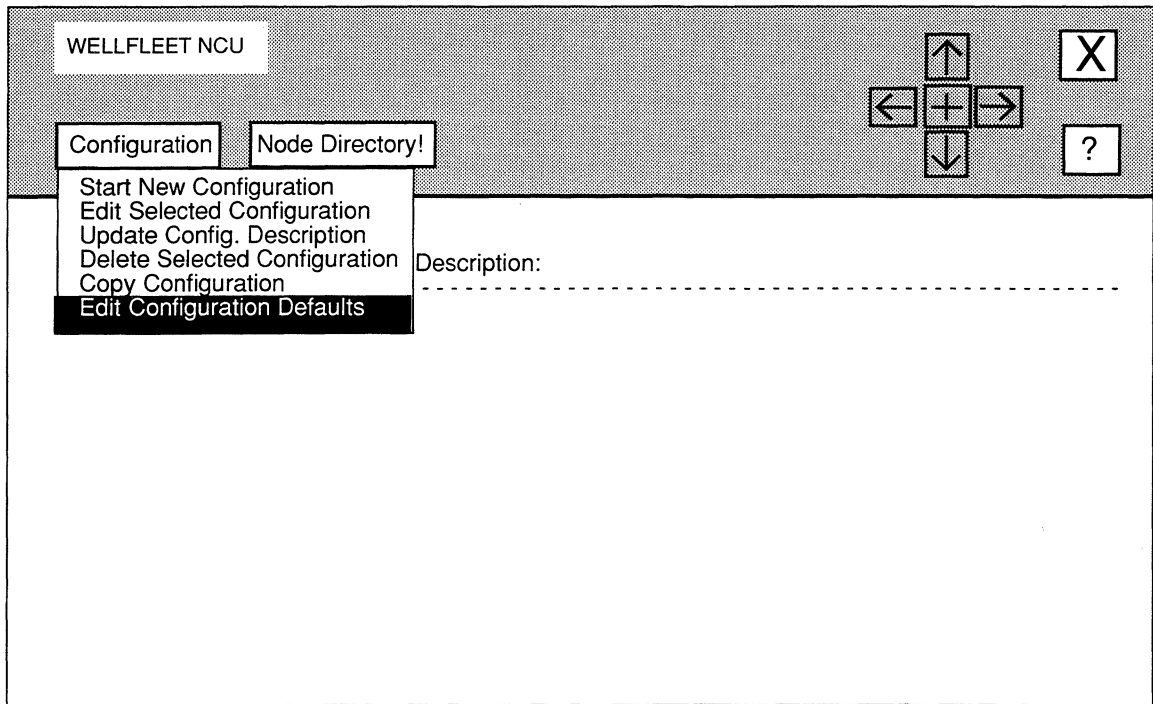


Figure 1-1. WELLFLEET NCU Window

The **EDIT NODE CONFIGURATION** window for the configuration-default parameters (see Figure 1-2) displays **DEFAULT_NODE** at **Node**, **DEFAULTS** at **Configuration Name**, and **Default configuration values** at **Comment**.

You may now proceed to the appropriate sections of this guide (refer to the Table of Contents for section numbers) for instructions on how to reset configuration parameters. Any changes that you make will affect every node you configure afterwards.

1.2 Accessing the Configuration Parameters of a Single Node

You access the configuration parameters of a single node, as follows:

1. **Display the CONFIGURATION NODE LIST window for the configuration associated with the node.**

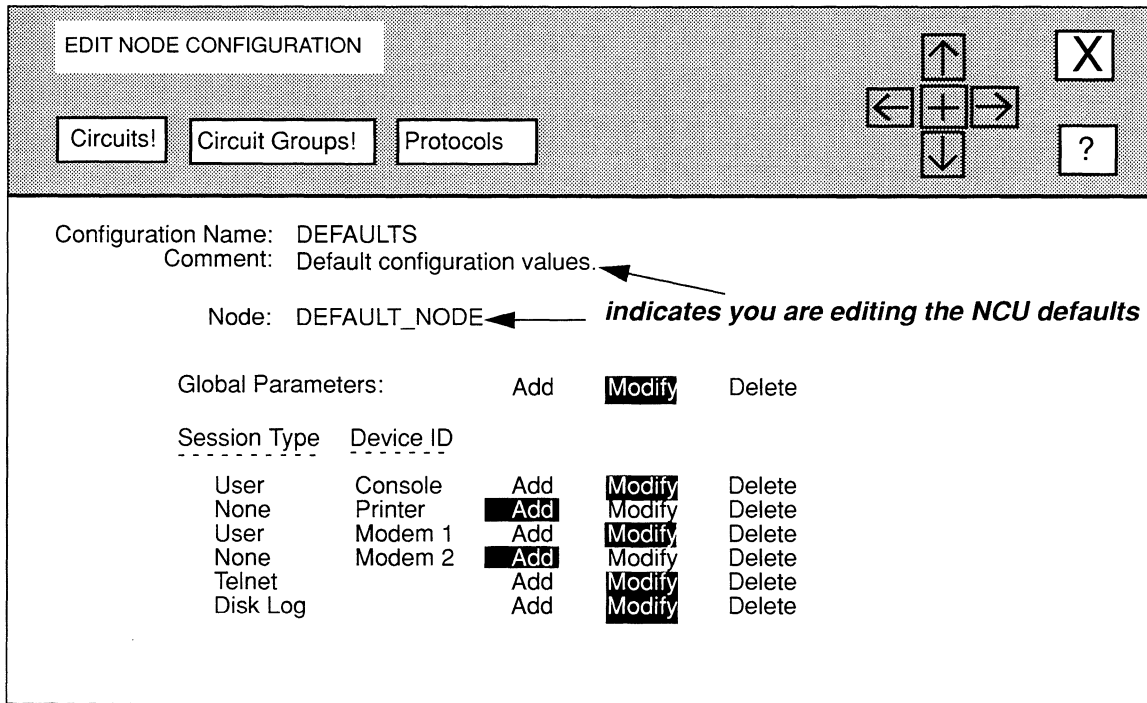


Figure 1-2. EDIT NODE CONFIGURATION Window for Default Settings

In Figure 1-3, the network operator displayed the **CONFIGURATION NODE LIST** for **NEUSCurrent**.

2. **Select the node that you wish to reconfigure.**

In Figure 1-3, the network operator selected **BOS**.

3. Select **Node Configuration** to display this menu:

Add Node to Config.
 Modify Selected Node Config.
 Remove Node From Config.
 Edit Node Configuration Detail
 Configuration File

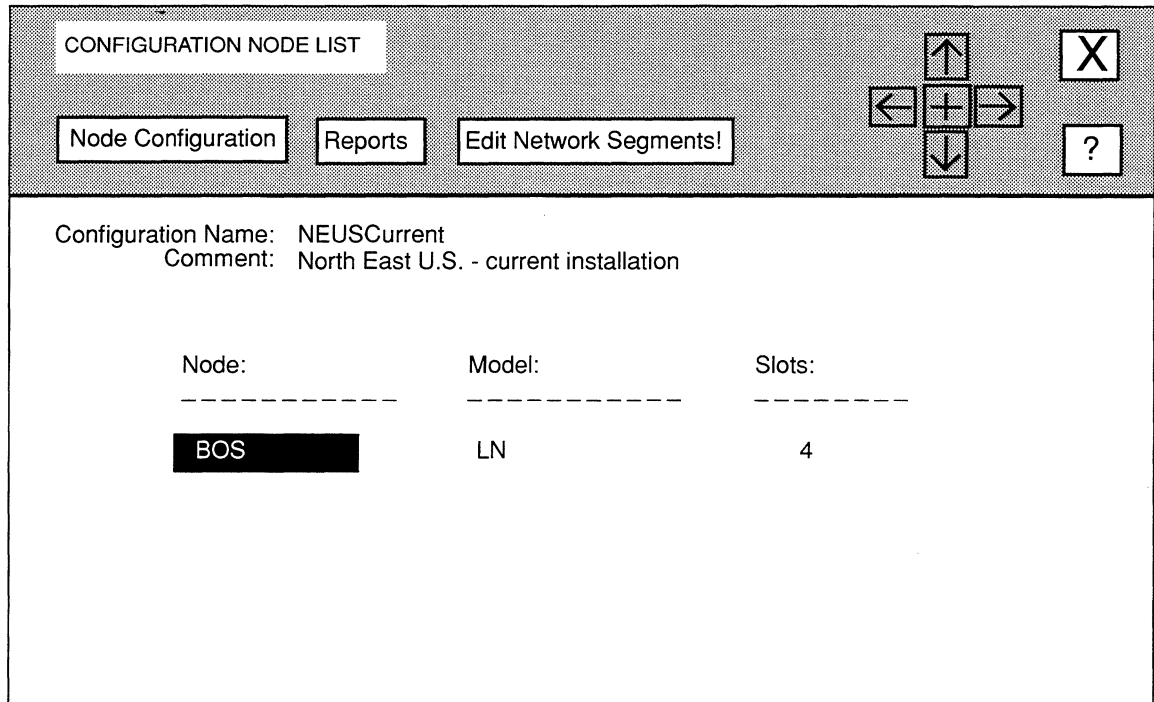


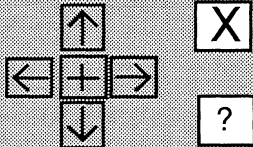
Figure 1-3. CONFIGURATION NODE LIST Window for NEUSCurrent

4. Select **Edit Node Configuration Detail** to display the **EDIT NODE CONFIGURATION** window for the node.

Figure 1-4 depicts the **EDIT NODE CONFIGURATION** window for **BOS**.

You may now proceed to the appropriate sections of this guide (refer to the Table of Contents for section numbers) for instructions on how to reset configuration parameters. Any changes that you make will affect only the node you selected in step 2.

EDIT NODE CONFIGURATION



Circuits!

Circuit Groups!

Protocols

Configuration Name: NEUSCurrent
 Comment: North East U.S. - current installation

Node: BOS ← *indicates you are editing the system parameters for BOS in the NEUSCurrent configuration*

Global Parameters: Add **Modify** Delete

Session Type	Device ID			
User	Console	Add	Modify	Delete
None	Printer	Add	Modify	Delete
User	Modem 1	Add	Modify	Delete
None	Modem 2	Add	Modify	Delete
Telnet		Add	Modify	Delete
Disk Log		Add	Modify	Delete

Figure 1-4. EDIT NODE CONFIGURATION Window for BOS

2 Editing System Parameters

System parameters consist of:

- ❑ Global Parameters

Global parameters specify which software version the node runs, how the node initializes software services, and how it reacts to failure.

- ❑ Session Parameters

Session parameters define the interfaces (or session modes) between the node and the I/O device or devices.

This chapter describes how to access and edit these parameters.

2.1 Accessing System Parameters

In order to access system parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

NOTE

Use the proper access mechanism to edit either the configuration-default global parameters or the global parameters of a single node. See Chapter 1.

Figure 2-1 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default system parameters in NCU; any changes the network operator makes will affect every node configured thence on.

2.2 Editing Global Parameters

NCU allows you to add, change, and delete global parameters.

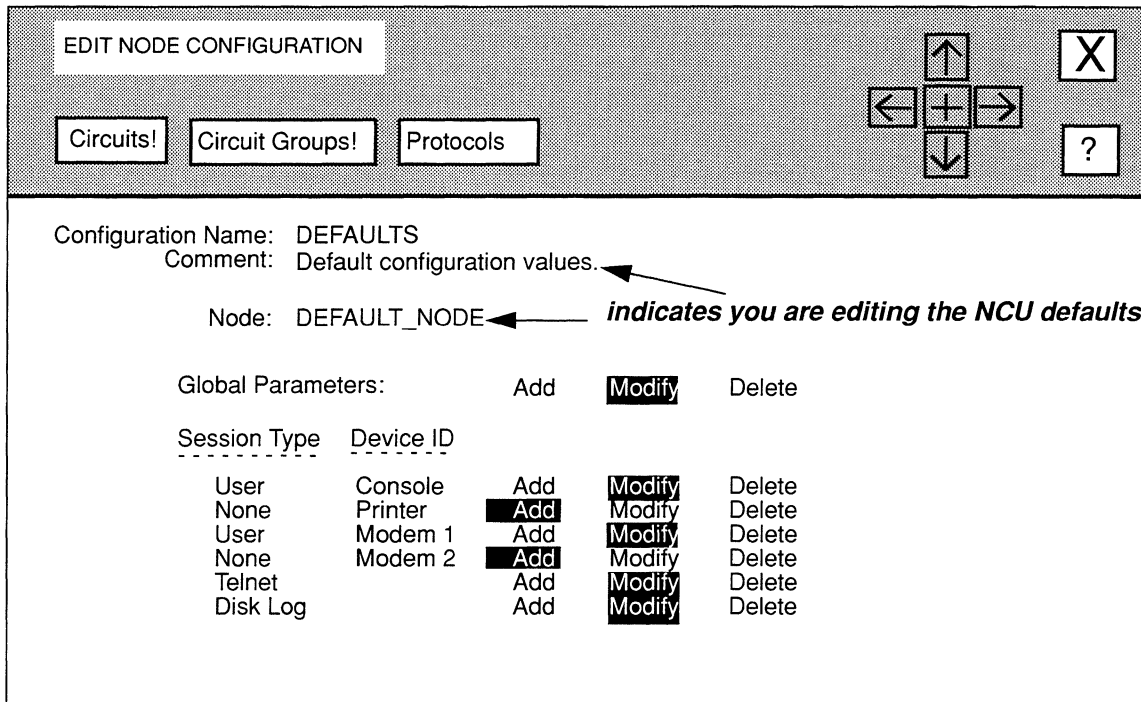


Figure 2-1. EDIT NODE CONFIGURATION Window for Default Settings

2.2.1 Adding Global Parameters

You add global parameters, as follows:

1. In the EDIT NODE CONFIGURATION window, select **Add** next to Global Parameters to display the GLOBAL PARAMETERS window.

The GLOBAL PARAMETERS window (see Figure 2-2) displays the global parameters.

2. Set the parameters as you wish (see Section 2.2.2, Changing Global Parameters).

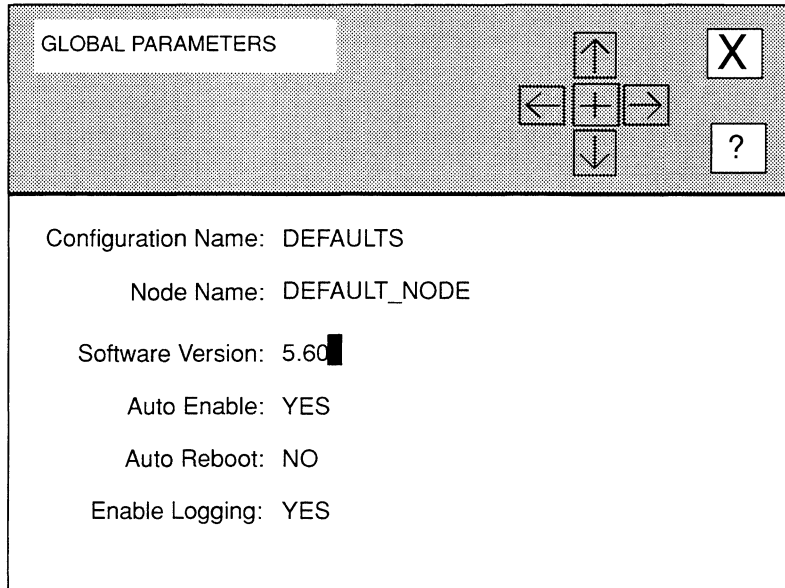


Figure 2-2. GLOBAL PARAMETERS Window for DEFAULT_NODE

2.2.2 Changing Global Parameters

You use the **GLOBAL PARAMETERS** window to change global parameters, as follows:

1. In the **EDIT NODE CONFIGURATION** window, select **Modify** next to **Global Parameters**.

NCU displays the **GLOBAL PARAMETERS** window. Figure 2-2 depicts the **GLOBAL PARAMETERS** window for the default-configuration settings in NCU; if it was the **GLOBAL PARAMETERS** window for a specific node, the window would display the node's name after **Node Name**, and the network configuration associated with the node after **Configuration Name**.

2. At **Software Version**, select the software version the node runs.

You may also type **[CONTROL] V** at **Software Version** to display a Values List (see Chapter 3 in *Volume 1: Simplified Network Definition of the NCU User Guide*) of valid software versions (see Figure 2-3). In Figure 2-3, the network operated selected **5.60**.

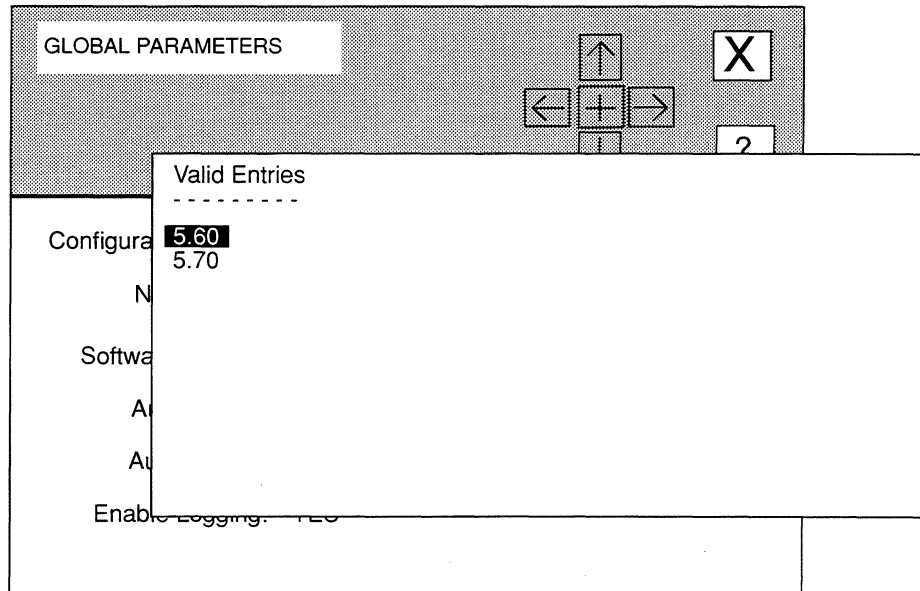


Figure 2-3. Software Version Values List

3. At Auto Enable, select the status of the application software and circuits when the node boots.

YES Specifies that the node conditionally enables the resident application software modules (the Bridge, the DoD IP Router, the DECnet Router, the XNS Router, the IPX Router, the AppleTalk Router) and all circuits when it boots.

NO Specifies that the node does not enable all resident application software and circuits when it boots.

If you select **NO**, you must subsequently enable application software and circuits manually with the Wellfleet NCL command **ENABLE** (see *Wellfleet Systems Operators Guide*).

4. **At Auto Reboot, select how the node responds to self failure.**

NO Specifies that the node does not reboot in the event of a failure (consequently, you would need to reboot the node manually).

YES Specifies that the node automatically tries to reboot in the event of failure.

5. **At Enable Logging, select whether you want the node to create a log file of sequential node-generated event messages on the system diskette.**

YES Specifies the node creates the log file.

The default file name is *log*. The default file contains a maximum of 50 entries, with each entry consisting of a single event message.

NO Specifies the node disables logging.

6. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the console.

```

      Press return when done.
    Global parameters set.
  
```

NCU returns to the **EDIT NODE CONFIGURATION** window and highlights **Modify** at **Global Parameters** to indicate that global parameters have been set.

2.2.3 Deleting Global Parameters

To delete global parameters, simply select **Delete** next to **Global Parameters** in the **EDIT NODE CONFIGURATION** window. NCU displays the following window:

```

    Are you sure you want to delete
    global parameters ? YES NO
  
```

Select **YES** to delete the global parameters.

2.3 Editing Session Parameters

Session parameters define the interfaces (or session modes) between the node and the I/O device or devices. There are four session modes, as follows:

Mode	Function
User	<p>Employs a directly-connected console or terminal device (or a similar device connected to the node with a modem) in an interactive fashion in order to exchange data with the node.</p> <p>NCU automatically configures two user sessions (one for a console and one for a modem) when you generate a <i>config</i> file for a node.</p>
Printer	<p>Employs a hard-copy device to collect node-generated event messages. Like disk-log mode, printer mode is unidirectional (data flow is from the node to the hard-copy device) and non-interactive.</p> <p>NCU does not configure a printer session when you generate a <i>config</i> file for a node.</p>
Telnet	<p>Employs the Internet virtual-terminal protocol in order to connect a remote IP host machine with the node. Except for the connection type, Telnet mode is identical to user mode.</p> <p>NCU automatically configures a telnet session when you generate a <i>config</i> file for a node.</p>
Disk Log	<p>Employs file space on the system disk to store node-generated event messages.</p> <p>NCU automatically configures a disk log session when you generate a <i>config</i> file for a node.</p>

NCU allows you to add, change, and delete each type of session parameter.

2.3.1 Editing User-Session Parameters

In order to edit user-session parameters, you must first select the port on the system I/O module for which you wish to add, change, or delete a user session. The **EDIT NODE CONFIGURATION** window identifies the four ports on the system I/O module as **Console**, **Printer**, **Modem 1**, and **Modem 2** under **Device ID** (see Figure 2-1). To the left of each port is the current session type (**User**, **Printer**, or **None**).

Once you select a port, you may add, change, or delete a user session.

2.3.1.1 Adding a User Session

You add a user session to a port, as follows:

- If the port is not configured (as indicated by **None** under **Session Type** next to the port):
Place the cursor over **None** and select **User** and go to step 1.
- If the port is configured for a **Printer** session (as indicated by **Printer** under **Session Type** next to the port):
You must delete the **Printer** session (see Section 2.3.2.3) before you can add a user session.
- If the port is configured for a **User** session (as indicated by **User** under **Session Type** next to the port):
You do not need to add a **User** session; you may change the session if you wish (see Section 2.3.1.2.)

1. Select **Add** next to the port to display the **USER SESSION** window.

The **USER SESSION** window (see Figure 2-4) displays the user-session parameters.

2. Set the parameters as you wish (see *Section 2.3.1.2, Changing a User Session*).

2.3.1.2 Changing a User Session

In order to change a **User** session, select **Modify** next to a port that is currently configured for a **User** session in order to display the **USER SESSION** window. Figure 2-5 depicts the **USER SESSION** window for **BOS**; if it was the **USER SESSION** window for the default-configuration settings in NCU, the window would display **DEFAULT_NODE** after **Node Name**, and **DEFAULTS** after **Configuration**.

The **USER SESSION** window allows you to edit user-session parameters. NCU automatically specifies the **Device ID** parameter to correspond with the port you selected.

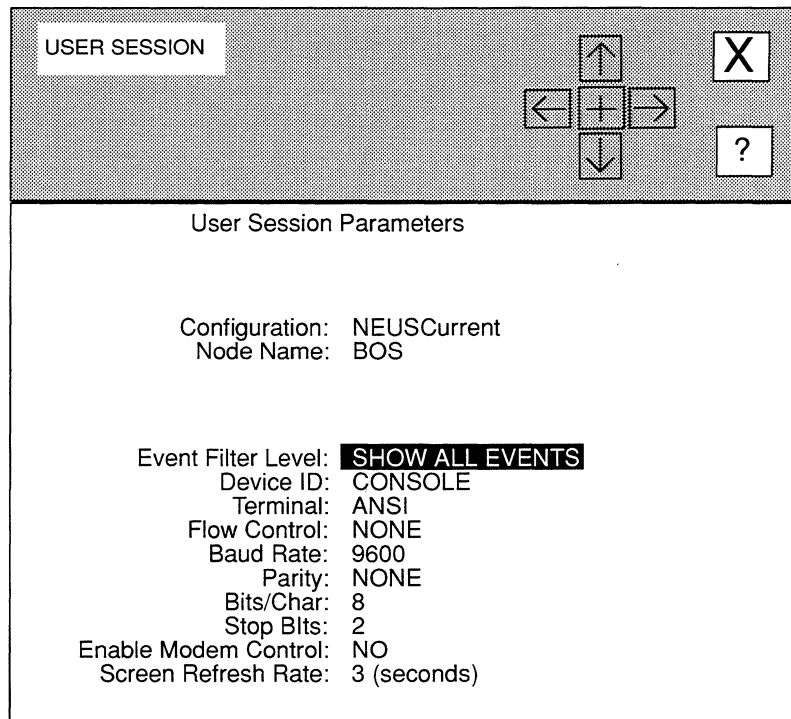


Figure 2-4. USER SESSION Window for BOS

You change these parameters, as follows:

1. **At Event Filter Level, select the severity level of event messages you want the node to display on the console screen.**

If you set **Enable Logging** to **YES** when you configured the global parameters, the node writes event messages (about network service and performance changes, and anomalous events) to a file on the system diskette. These event messages have five levels of severity, as follows:

Severity Level	Indicates
Major	A service has appeared or disappeared.
Warning	A service has behaved unexpectedly.
Performance	A service has upgraded/degraded.
Information	General system information.
Debug	Installation/diagnostic information.

NCU provides six responses to **Event Filter Level**:

- DEBUG** Specifies that the node displays messages on the console screen with these severity levels: Major, Warning, Performance, Information, and Debug.
- SHOW ALL EVENTS** Specifies that the node displays messages on the console screen with these severity levels: Major, Warning, Performance, and Information.
- NOT INFO**..... Specifies that the node displays messages on the console screen with these severity levels: Major, Warning, and Performance.
- PERF AND MAJOR** Specifies that the node displays messages on the console screen with these severity levels: Major and Performance.
- JUST MAJOR** Specifies that the node displays messages on the console screen with Major severity levels.
- DROP ALL**..... Specifies that the node displays no event messages.

2. At Terminal, select the type of console device.

- ANSI** Identifies ANSI-compatible devices.
- VT100** Identifies VT100-compatible devices.

3. At Flow Control, based on whether the console implements the XON/XOFF protocol, (which controls the data-transfer rate between the console and the node,) enable or disable the XON/XOFF protocol,

- NONE**..... Disables the protocol.
- XON/XOFF**..... Enables the protocol.

4. At Baud Rate, select the data-transfer rate for your console.

NCU provides five responses: **9600**, **4800**, **2400**, **1200**, and **300**.

5. At Parity, set the parity-scheme (Parity assigns a value to the eighth bit of each ASCII characters that the node transmits) that corresponds with the console's.

NCU provides three responses: **NONE**, **ODD**, and **EVEN**.

6. At Bits/Char, based on the requirements of the console, select the number of bits in each ASCII character the node receives or transmits.

NCU provides two responses: **8** and **7**.

7. **At Stop Bits, based on the requirements of the console, select the number of bits that follow each ASCII character the node receives or transmits.**

NCU provides three responses: 2, 1, and 1.5.

8. **At Enable Modem Control, select whether a modem connects the console to the node.**

YES Specifies a modem connects the console to the node.

NO Specifies no modem connects the console to the node.

9. **At Screen Refresh Rate, select the frequency rate (in seconds per cycle) at which the node updates the display of various reporting metrics.**

NCU provides eight responses: 3, 5, 10, 20, 30, 45, 60, and 1.

You can access these reporting metrics from the Statistics Screen Menu. The *Wellfleet Operator's Guide* describes how to access and interpret reporting metrics.

10. **Select and then .**

NCU displays the following window; press **[RETURN]** to clear it from the console.

```
Press return when done.
User session parameters set.
```

NCU returns to the **EDIT NODE CONFIGURATION** window and highlights **Modify** next to the port to indicate that user-session parameters have been set.

2.3.1.3 Deleting a User Session

To delete a **User** session from a port, select **Delete** next to the port. NCU displays the following window:

```
Are you sure you want to delete
console parameters ?YES NO
```

In the above window, the network operator is deleting a **User** Session from the **Console** port; if the **User** session had been configured for another port on the system I/O module, the window would reference that port.

Select **YES** to delete the user session.

2.3.2 Editing Printer-Session Parameters

In order to edit printer-session parameters, you must first select the port on the system I/O module for which you wish to add, change, or delete a printer session. The **EDIT NODE CONFIGURATION** window identifies the four ports on the system I/O module as **Console**, **Printer**, **Modem 1**, and **Modem 2** under **Device ID** (see Figure 2-1). To the left of each port is the current session type (**User**, **Printer**, or **None**).

Once you select a port, you may add, change, or delete a printer session.

2.3.2.1 Adding a Printer Session

You add a printer session to a port, as follows:

- If the port is not configured (as indicated by **None** under **Session Type** next to the port):
Place the cursor over **None** and select **Printer**; then go to step 1
- If the port is configured for a **User** session (as indicated by **User** under **Session Type** next to the port):
You must first delete the **User** session (see Section 2.3.1.3), before you can add a printer session.
- If the port is configured for a **Printer** session (as indicated by **Printer** under **Session Type** next to the port):
You do not need to add a **Printer** session; you may change the session if you wish (see Section 2.3.2.2)

1. Select **Add** next to the port to display the **PRINTER SESSION** window.

The **PRINTER SESSION** window (see Figure 2-5) displays the printer-session parameters.

2. Set the parameters as you wish (see *Section 2.3.2.2, Changing a Printer Session*).

2.3.2.2 Changing a Printer Session

In order to change a **Printer** session, select **Modify** next to a port that is currently configured for a **Printer** to display the **PRINTER SESSION** window. Figure 2-5 depicts the **PRINTER SESSION** window for **BOS**; if it was the **PRINTER SESSION** window for the default-configuration settings in NCU, the window would display **DEFAULT_NODE** after **Node Name**, and **DEFAULTS** after **Configuration**.

The **PRINTER SESSION** window allows you to edit these printer-session parameters. NCU automatically specifies the **Device ID** parameter to correspond with the port you selected, and the **Terminal** parameter to correspond with a printer.

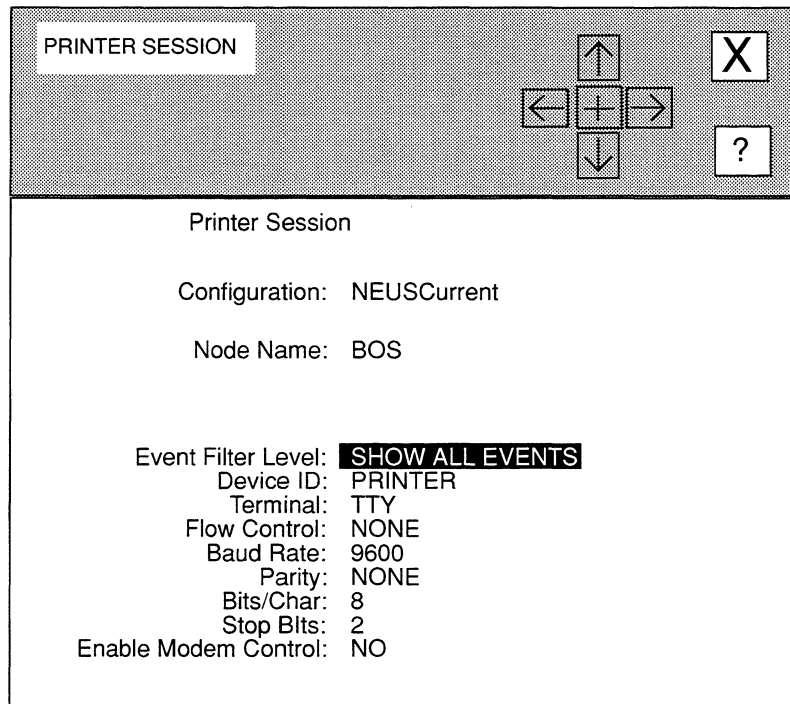


Figure 2-5. PRINTER SESSION Window

You change these parameters, as follows:

1. **At Event Filter Level, select the severity level of event messages you want the node to display on the console screen.**

If you set **Enable Logging** to **YES** when you configured the global parameters, the node writes event messages (about network service and performance changes, and anomalous events) to a file on the system diskette. These event messages have five levels of severity, as follows:

Severity Level	Indicates
Major	A service has appeared or disappeared.
Warning	A service has behaved unexpectedly.
Performance	A service has upgraded/degraded.
Information	General system information.
Debug	Installation/diagnostic information.

NCU provides six responses to **Event Filter Level**:

- DEBUG** Specifies that the node sends messages to the printer with these severity levels: Major, Warning, Performance, Information, and Debug.
- SHOW ALL EVENTS** Specifies that the node sends messages to the printer with these severity levels: Major, Warning, Performance, and Information.
- NOT INFO**..... Specifies that the node sends messages to the printer with these severity levels: Major, Warning, and Performance.
- PERF AND MAJOR** Specifies that the node sends messages to the printer with these severity levels: Major and Performance.
- JUST MAJOR** Specifies that the node sends messages to the printer with Major severity levels.
- DROP ALL**..... Specifies that the node sends no event messages to the printer.

2. **At Flow Control, based on whether the printer implements the XON/XOFF protocol (which controls the data-transfer rate between the printer and the node) enable or disable the XON/XOFF protocol.**

- NONE**..... Specifies the node disables the protocol.
- XON/XOFF**..... Specifies the node enables the protocol.

3. **At Baud Rate, select the data-transfer rate for your printer.**

NCU provides five responses: **9600, 4800, 2400, 1200, and 300.**

4. **At Parity, select the parity-scheme (Parity assigns a value to the eighth bit of each ASCII character that the node transmits) that corresponds with the printer's.**

NCU provides three responses: **NONE, ODD, and EVEN.**

5. **At Bits/Char, select the number of bits in each ASCII character the node receives or transmits.**

NCU provides two responses: **8 and 7.** Base your response on the printer requirements.

6. **At Stop Bits, based on printer requirements, select the number of bits that follow each ASCII character the node receives or transmits.**

NCU provides three responses: **2, 1, and 1.5.**

7. At **Enable Modem Control**, select whether a modem connects the printer to the console.

YES Specifies a modem connects the printer to the node.

NO Specifies no modem connects the printer to the node.

8. At **Screen Refresh Rate**, select the frequency rate (in seconds per cycle) at which the node updates the display of various reporting metrics.

NCU provides eight responses: **3, 5, 10, 20, 30, 45, 60**, and **1**.

You can access these reporting metrics from the Statistics Screen Menu. The *Wellfleet Operator's Guide* describes how to access and interpret reporting metrics.

9. Select and then **Save** .

NCU displays the following window; press **[RETURN]** to clear it from the console.

```
Press return when done.
Printer session parameters set.
```

NCU returns to the **EDIT NODE CONFIGURATION** window and highlights **Modify** next to the port to indicate that printer-session parameters have been set.

2.3.2.3 Deleting a Printer Session

To delete a **Printer** session from a port, select **Delete** next to the port. NCU displays the following window:

```
Are you sure you want to delete
console parameters ?YES NO
```

In the above window, the network operator is deleting a **Printer** Session from the **Console** port; if the **Printer** session had been configured for another port on the system I/O module, the window would reference that port.

Select **YES** to delete the printer session.

2.3.3 Editing Telnet-Session Parameters

NCU allows you to add, change, and delete Telnet sessions.

2.3.3.1 Adding a Telnet Session

You add a telnet session, as follows:

1. Select **Add** next to **Telnet** in the **EDIT NODE CONFIGURATION** window to display the **TELNET SESSION** window.

The **TELNET SESSION** window displays the telnet-session parameters. NCU automatically specifies the **Device ID** parameter to correspond with a telnet session.

Figure 2-6 depicts the **TELNET SESSION** window for **BOS**; if it was the **TELNET SESSION** window for the default-configuration settings in NCU, the window would display **DEFAULT_NODE** after **Node Name**, and **DEFAULTS** after **Configuration**.

2. Set the parameters as you wish (see *Section 2.3.3.2, Changing a Telnet Session*).

2.3.3.2 Changing a Telnet Session

You change a telnet session, as follows:

1. Select **Modify** next to **Telnet** in the **EDIT NODE CONFIGURATION** window to display the **TELNET SESSION** window (see Figure 2-6).
2. At **Event Filter Level**, specify the severity level of event messages you want the node to transmit to the remote device.

If you set **Enable Logging** to **YES** when you configured the global parameters, the node writes event messages (about network service and performance changes, and anomalous events) to a file on the system diskette. These event messages have five levels of severity, as follows:

Severity Level	Indicates
Major	A service has appeared or disappeared.
Warning	A service has behaved unexpectedly.
Performance	A service has upgraded/degraded.
Information	General system information.
Debug	Installation/diagnostic information.

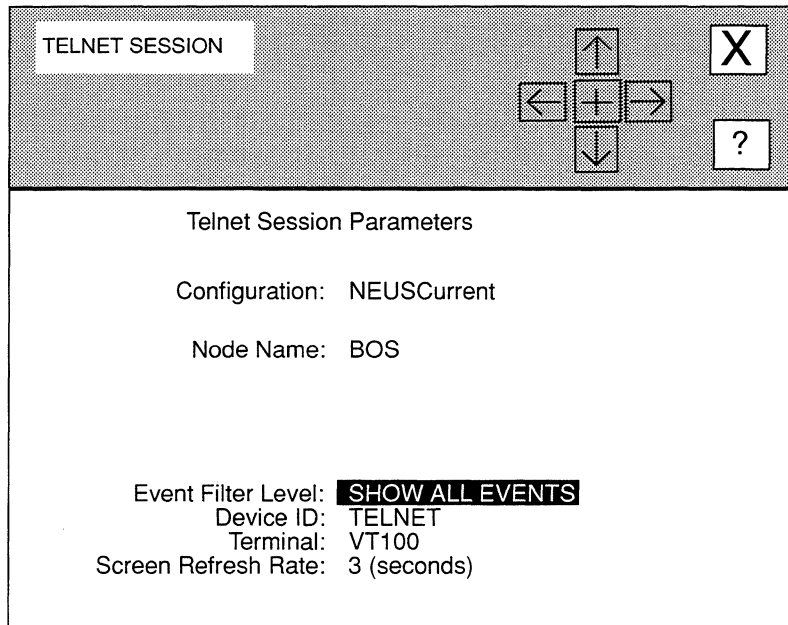


Figure 2-6. TELNET SESSION Window

NCU provides six responses to **Event Filter Level**:

- DEBUG** Specifies that the node transmits event messages to the remote device with these severity levels: Major, Warning, Performance, Information, and Debug.
- SHOW ALL EVENTS** Specifies that the node transmits event messages to the remote device with these severity levels: Major, Warning, Performance, and Information.
- NOT INFO**..... Specifies that the node transmits event messages to the remote device with these severity levels: Major, Warning, and Performance.
- PERF AND MAJOR** Specifies that the node transmits event messages to the remote device with these severity levels: Major and Performance.
- JUST MAJOR** Specifies that the node transmits event messages to the remote device with Major severity levels.
- DROP ALL**..... Specifies that the node transfers no event messages to the remote device.

3. **At Terminal, select the type of remote console device.**

ANSI Specifies an ANSI-compatible device.

VT100 Specifies a VT100-compatible device.

4. **At Screen Refresh Rate, select the rate (in seconds per cycle) at which the node updates the display of various reporting metrics.**

NCU provides eight responses: **3, 5, 10, 20, 30, 45, 60,** and **1.**

You can access these metrics from the Statistics Screen Menu. The *Wellfleet Operator's Guide* describes how to access and interpret reporting metrics.

NOTE

Frequent updates of reporting metrics may cause network congestion and may use available bandwidth inefficiently.

5. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the console.

Press return when done.

Telnet session parameters set.

NCU returns to the **EDIT NODE CONFIGURATION** window and highlights **Modify** next to **Telnet** to indicate that telnet-session parameters have been set.

2.3.3.3 Deleting a Telnet Session

To delete a **Telnet** session, select **Delete** next to **Telnet**. NCU displays the following window:

Are you sure you want to delete
the telnet session ? YES **NO**

Select **YES** to delete the telnet session.

2.3.4 Editing Disk-Log Session Parameters

During a disk-log session, the node directs output to a disk file. The node supports a single disk-logging session. Once created, the session can be disabled with the NCL **DISABLE** command, and re-enabled with the NCL **ENABLE** command.

NOTE

The NCL **REMOVE** command automatically disables disk logging. The NCL **INSERT** command restores disk logging. Disk logging is also disabled during the execution of the NCL **COPY** command. Refer to the *Wellfleet Systems Operator's Guide* for information on NCL commands.

NCU allows you to add, modify, and delete a disk-log session.

2.3.4.1 Adding a Disk-Log Session

You add a disk-log session, as follows:

1. Select **Add** next to Disk Log in the EDIT NODE CONFIGURATION window to display the LOG SESSION window.

The LOG SESSION window displays the disk-log session parameters. Figure 2-7 depicts the LOG SESSION window for BOS; if it was the LOG SESSION window for the default-configuration settings in NCU, the window would display **DEFAULT_NODE** after **Node Name**, and **DEFAULTS** after **Configuration**.

2. Set the parameters as you wish (see Section 2.3.4.2, *Changing a Disk-Log Session*).

2.3.4.2 Changing a Disk-Log Session

You change a disk-log session, as follows:

1. Select **Modify** next to Disk Log in the EDIT NODE CONFIGURATION window to display the LOG SESSION window.

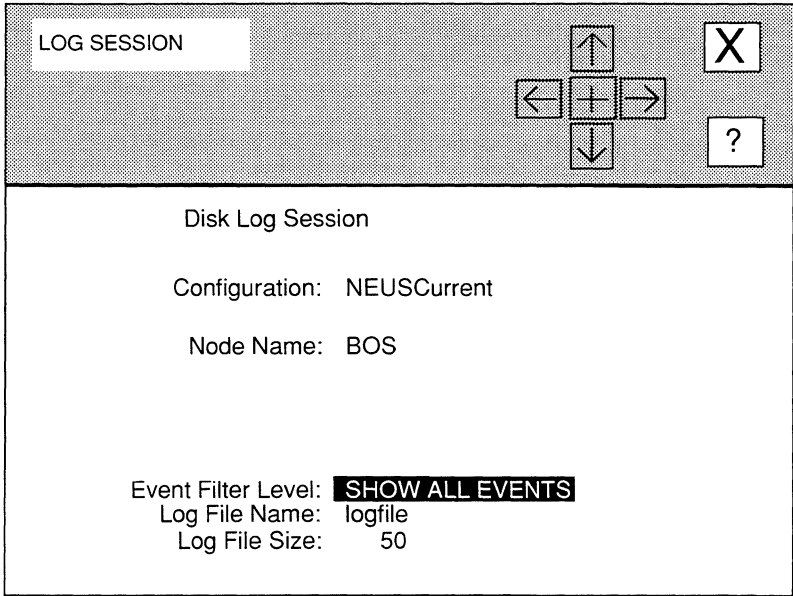


Figure 2-7. LOG SESSION Window

- 1. At Event Filter Level, specify the severity level of event messages you want the node to store in the log file.

Event messages have five severity levels, as follows:

Severity Level	Indicates
Major	A service has appeared or disappeared.
Warning	A service has behaved unexpectedly.
Performance	A service has upgraded/degraded.
Information	General system information.
Debug	Installation/diagnostic information.

NCU provides six responses to **Event Filter Level**:

- DEBUG** Specifies the node stores event messages in the disk-log file with these severity levels: Major, Warning, Performance, Information, and Debug.
- SHOW ALL EVENTS** Specifies the node stores event messages in the disk-log file with these severity levels: Major, Warning, Performance, and Information.

- NOT INFO**..... Specifies the node stores event messages in the disk-log file with these severity levels: Major, Warning, and Performance.
- PERF AND MAJOR** Specifies the node stores event messages in the disk-log file with these severity levels: Major and Performance.
- JUST MAJOR** Specifies the node store event messages in the disk-log file with Major serverity levels.
- DROP ALL**..... Specifies the node stores no event messages in the disk-log file.

NOTE

If you configure a disk-log session when the global **Enable Logging** parameter is set to **NO**, the node creates the disk-logging session, but stores no events in the disk file until the session is enabled with the NCL **ENABLE** command. See the *Wellfleet Systems Operators Guide* for information on NCL commands.

2. At Log File Name, enter the name of the disk file you wish to create.

If you wish, you can implement a log-file numbering feature that creates up to 10 log files (identified by a sequential numeric suffix, 0 through 9). With this feature enabled, the node creates a new log file each time it is rebooted either with the **RESET** button or with the NCL **BOOT** command. Prior to creating the new file, it closes the previous log file, appends a numeric suffix to the file name, and saves the file on the system diskette.

NOTE

The numeric suffix appended to the file name corresponds to the crash file number. Consequently, the nth log file is logically accompanied by the nth crash file that describes the reason for the reboot.

To implement sequential log files, at **Log File Name**, enter a file name in the following format:

filename.*

for example, **log.***

3. At Log File Size, enter the maximum number of event messages (from 1 to 999) that the log file can contain.

Once this number is exceeded, the node writes over the oldest file item as new items are added.

4. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the console.

```
Press return when done.
Log session parameters set.
```

NCU returns to the **EDIT NODE CONFIGURATION** window and highlights **Modify** next to **Disk Log** to indicate that disk-log session parameters have been set.

2.3.4.3 Deleting a Disk-Log Session

To delete a **Disk-Log** session, select **Delete** next to **Disk Log**. NCU displays the following window:

```
Are you sure you want to delete
the log session    ? YES NO
```

Select **YES** to delete the disk-log session.

CIRCUITS			
Modify Circuit!		Delete Circuit!	
Configuration: NEUSCurrent Node Name: BOS		<div style="display: flex; justify-content: space-around;"> ← ↑ + → × </div> <div style="display: flex; justify-content: space-around;"> ↓ ? </div>	
Slot:	Connector:	Circuit:	Network:
2 ■	XCVR1	E21	CHIR&D1
	COM1	S21	NYTOCHICAGO
	COM2	S22	NYTOCHICAGO

Figure 3-2. CIRCUITS Window with Ethernet Connector Selected

- If you selected **TOKEN**:
NCU displays the **TOKEN CIRCUIT CONFIGURATION** window, which allows you to edit token-ring/LAN-circuit parameters; go to *Section 3.3, Editing Token-Ring/LAN-Circuit Parameters* for instructions.
- If you selected **FDDI**:
NCU displays the **FDDI CIRCUIT CONFIGURATION** window, which allows you to edit FDDI/LAN-circuit parameters; go to *Section 3.4, Editing FDDI/LAN-Circuit Parameters* for instructions.
- If you selected **COM1**, **COM2**, **COM3**, or **COM4**:
NCU displays the **SYNC LINE** window, which allows you to edit synchronous line parameters. Based on the defined node/network-segment connection, the window also provides access either to point-to-point circuit parameters or X.25 circuit parameters; go to *Section 3.5, Editing Synchronous-Line Parameters* for instructions.

- If you selected **DS1-1** or **DS1-2**:
 NCU displays the **T1 LINE** window, which allows you to edit T1 line parameters and provides access to point-to-point circuit parameters; go to *Section 3.6, Editing T1-Line Parameters* for instructions.
- If you selected **E1-1** or **E1-2**:
 NCU displays the **E1 LINE** window, which allows you to edit E1 line parameters and provides access to point-to-point circuit parameters; go to *Section 3.7, Editing E1-Line Parameters* for instructions.

3.2 Editing Ethernet/LAN-Circuit Parameters

You edit Ethernet/LAN-circuit parameters from the **LAN CIRCUIT CONFIGURATION** window (see Figure 3-3). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.
Slot	Slot containing the Ethernet module.
Connector	Specific transceiver on the Ethernet module.
Physical Access Method	CSMA/CD (Carrier Sense Multiple Access/with Collision Detection) — the physical-access method of Ethernet lines.
Circuit Type	LAN circuit (Ethernet lines require LAN circuits).
Quality of Service	LLC1 (connectionless datagram service — the best-effort delivery model). LLC1 is the quality-of-service level for LAN circuits.
XCVR Signal Polling	XCVR signal polling is ACTIVE for LAN circuits. XCVR signal polling is when node software transmits periodic self-address messages to verify proper transceiver operation.

You may set three parameters (**Circuit Name**, **Auto Enable**, and **LAN Address**) in the **LAN CIRCUIT CONFIGURATION** window, as follows:

1. **At Circuit Name, enter a new circuit name.**

A circuit name may contain up-to-12 printable characters *except for the period (.)*.

LAN CIRCUIT CONFIGURATION	
Configuration: NEUSCurrent	Slot: 2
Node: BOS	Connector: XCVR1
	Physical Access Method: CSMA/CD
Circuit Name: E2-1	Circuit Type: LAN
Auto Enable: YES	
LAN Address:	Quality of Service: LLC1
	XCVR Signal Polling: ACTIVE

Figure 3-3. LAN CIRCUIT CONFIGURATION Window

2. At Auto Enable, specify the state of this LAN circuit when the node boots.

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this LAN circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.
- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the LAN circuit.
- Select **NO** to disable the LAN circuit (you will subsequently need to enable the LAN circuit manually with the NCL Interpreter after the node boots).

3. **At LAN Address, enter the node's 48-bit Ethernet address (in 12-character hexadecimal format).**

Wellfleet ships all nodes with a unique universally-administered address written in read-only memory (ROM). This address has a high-order (most significant 24-bits) value of 0000A2, hexadecimal, and a low-order (least significant 24-bits) value unique to the node. If you do not specify a value at **LAN Address**, the node takes the default address.

Note

The node ignores the value of **LAN Address** on circuits which support AppleTalk, DECnet, IPX, XNS, or the Bridge with spanning tree *enabled*. In such instances, the bridging/routing software asserts an internally-generated LAN address.

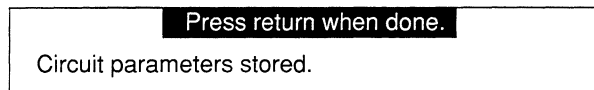
If the node uses only the IP Router, or if it uses the IP Router in conjunction with the Bridge with spanning tree *disabled*, you may assign a unique Ethernet address of your choosing. Each LAN device within your network requires a unique 48-bit address, so do not duplicate addresses.

Note

When you assign a value to **LAN Address**, ensure that the least significant bit of the most significant byte is clear (equal to zero). When LAN addresses are "sent across the wire", their bit order is reversed. Consequently, the least-significant bit of the most-significant byte is transmitted first. A logical one in the first bit position of a destination address designates a broadcast or multicast address.

4. Select and then Save .

NCU displays this window; press **[RETURN]** to clear it from the console.



3.3 Editing Token-Ring/LAN-Circuit Parameters

You edit Token-Ring/LAN-circuit parameters from the **TOKEN CIRCUIT CONFIGURATION** window (see Figure 3-4). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.

TOKEN CIRCUIT CONFIGURATION	
Configuration: NEUSCurrent	Slot: 3
Node: BOS	Connector: TOKEN
	Physical Access Method: TOKEN RING
	Ring Interface: 4MBPS
Circuit Name: T31	Circuit Type: LAN
Auto Enable: YES	Quality of Service: LLC1
LAN Address: [REDACTED]	XCVR Signal Polling: ACTIVE

Figure 3-4. TOKEN CIRCUIT CONFIGURATION Window

This Field (continued)	Displays
Slot	Slot containing the Token Ring module.
Connector	Specific connector on the module.
Physical Access Method	TOKEN RING — the physical-access method of Token Ring lines.
Circuit Type	LAN — Token Ring lines require LAN circuits.
Quality of Service	LLC1 (connectionless datagram service — the best-effort delivery model). LLC1 is the quality-of-service level for LAN circuits.
XCVR Signal Polling	XCVR signal polling is ACTIVE for LAN circuits. XCVR signal polling is when node software transmits periodic self-address messages to verify proper transceiver operation.

You may set four parameters (**Ring Interface**, **Circuit Name**, **Auto Enable**, and **LAN Address**) in the **TOKEN CIRCUIT CONFIGURATION** window, as follows:

1. **At Ring Interface, select the Token-Ring service offered by the attached network.**

4MBPS..... Specifies 5Mb/s service.

16MBPS..... Specifies 16Mb/s service.

16MBPSETR..... Specifies 16 Mb/s service with the Early Token Release option.

2. **At Circuit Name, enter a new circuit name.**

A circuit name may contain up-to-12 printable characters *except for the period* (.).

3. **At Auto Enable, specify the state of this LAN circuit when the node boots.**

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this LAN circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.

You will subsequently need to enable this circuit manually with the NCL Interpreter after the node boots.

- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the LAN circuit.
- Select **NO** to disable the LAN circuit (you will subsequently need to enable the LAN circuit manually with the NCL Interpreter after the node boots).

4. **At LAN Address, enter the node's 48-bit Ethernet address (in 12-character hexadecimal format).**

Wellfleet ships all nodes with a unique universally-administered address written in read-only memory (ROM). This address has a high-order (most significant 24-bits) value of 0000A2, hexadecimal, and a low-order (least significant 24-bits) value unique to the node. If you do not specify a value at **LAN Address**, the node takes the default address.

Note

The node ignores the value of **LAN Address** on circuits which support AppleTalk, DECnet, IPX, XNS, or the Bridge with spanning tree *enabled*. In such instances, the bridging/routing software asserts an internally-generated LAN address.

If the node uses only the IP Router, or if it uses the IP Router in conjunction with the Bridge with spanning tree *disabled*, you may assign a unique Ethernet address of your choosing. Each LAN device within your network requires a unique 48-bit address, so do not duplicate addresses.

Note

When you assign a value to **LAN Address**, ensure that the least significant bit of the most significant byte is clear (equal to zero). When LAN addresses are “sent across the wire”, their bit order is reversed. Consequently, the least-significant bit of the most-significant byte is transmitted first. A logical one in the first bit position of a destination address designates a broadcast or multicast address.

5. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

Circuit parameters stored.

3.4 Editing FDDI/LAN-Circuit Parameters

You edit FDDI/LAN-circuit parameters from the **FDDI CIRCUIT CONFIGURATION** window (see Figure 3-5). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.
Slot	Slot containing the FDDI module.
Connector	Specific connector on the module.
Physical Access Method	FDDI — the physical-access method of FDDI lines.
Circuit Type	LAN — FDDI lines require LAN circuits.

FDDI CIRCUIT CONFIGURATION

Configuration: NEUSCurrent	Slot: 8
Node: BOS	Connector: FDDI
	Physical Access Method: FDDI
Board Number:	
Circuit Name: F81	Circuit Type: LAN
Auto Enable: YES	Quality of Service: LLC1
LAN Address:	XCVR Signal Polling: ACTIVE

Figure 3-5. FDDI CIRCUIT CONFIGURATION Window

This Field	Displays
Quality of Service	LLC1 (connectionless datagram service — the best-effort delivery model). LLC1 is the quality-of-service level for LAN circuits.
XCVR Signal Polling	XCVR signal polling is ACTIVE for LAN circuits. XCVR signal polling is when node software transmits periodic self-address messages to verify proper transceiver operation.

You may set four parameters (**Board Number**, **Circuit Name**, **Auto Enable**, and **LAN Address**) in the **FDDI CIRCUIT CONFIGURATION** window, as follows:

1. **At Board Number, select the hardware (VMEbus) address of the FDDI controller.**

1 Specifies the short address **0000**; if the node contains a single factory-installed FDDI board set, select **1**.

-
- 2 Specifies the short address **0200**.
- 3 Specifies the short address **0400**.

Note

If the node contains more than one FDDI board set, or if you have manually changed the factory-set address jumpers, you must specify the actual FDDI hardware address. Refer to the *Wellfleet Installation Guide* for information on setting and determining FDDI hardware addresses.

2. At Circuit Name, enter a new circuit name.

A circuit name may contain up-to-12 printable characters *except for the period (.)*.

3. At Auto Enable, specify the state of this LAN circuit when the node boots.

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this LAN circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.

You will subsequently need to enable this circuit manually with the NCL Interpreter after the node boots.

- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the LAN circuit.
- Select **NO** to disable the LAN circuit (you will subsequently need to enable the LAN circuit manually with the NCL Interpreter after the node boots).

4. At LAN Address, enter the node's 48-bit Ethernet address (in 12-character hexadecimal format).

Wellfleet ships all nodes with a unique universally-administered address written in read-only memory (ROM). This address has a high-order (most significant 24-bits) value of 0000A2, hexadecimal, and a low-order (least significant 24-bits) value unique to the node. If you do not specify a value at **LAN Address**, the node takes the default address.

Note

The node ignores the value of **LAN Address** on circuits which support AppleTalk, DECnet, IPX, XNS, or the Bridge with spanning tree *enabled*. In such instances, the bridging/routing software asserts an internally-generated LAN address.

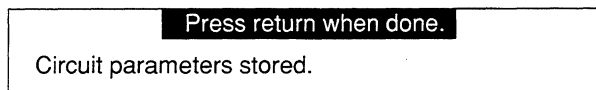
If the node uses only the IP Router, or if it uses the IP Router in conjunction with the Bridge with spanning tree *disabled*, you may assign a unique Ethernet address of your choosing. Each LAN device within your network requires a unique 48-bit address, so do not duplicate addresses.

Note

When you assign a value to **LAN Address**, ensure that the least significant bit of the most significant byte is clear (equal to zero). When LAN addresses are “sent across the wire”, their bit order is reversed. Consequently, the least-significant bit of the most-significant byte is transmitted first. A logical one in the first bit position of a destination address designates a broadcast or multicast address.

5. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.



3.5 Editing Synchronous-Line Parameters

You edit synchronous-line parameters from the **SYNC LINE** window (see Figure 3-6). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.
Slot	Slot containing synchronous link module.
Connector	Specific module communications port.
Physical Access Method	SYNC — physical access method of synchronous lines.
Circuit Name	Current circuit name.
Circuit Type	Current circuit type.

SYNC LINE

Circuit!

Configuration: NEUSCurrent Slot: 2
Node: BOS Connector: COM1

Physical Access Method: SYNC

Signal Mode: **V.35/RS-422 BALANCED**
Clock Source: EXTERNAL

RTS/CTS Control: NO

Circuit Name: S21
Circuit Type: POINT TO POINT

Figure 3-6. SYNC LINE Window

You may set three parameters (**Signal Mode**, **Clock Source**, and **RTS/CTS Control**) in the **SYNC LINE** window, as follows:

1. At Signal Mode, select the signalling mode of the connected device.

V.35/RS-422 BALANCED..... Specifies that the connected device uses two conductors to carry signals.

RS-423 UNBALANCED..... Specifies that the connected device uses a single conductor to carry a signal, with a ground providing the return path.

2. At Clock Source, select the origin of the synchronous timing signals.

EXTERNAL..... Specifies that an external network device supplies the required timing signals. In most cases, **EXTERNAL** is the appropriate response. The upper range for external clocking is 6.144 Mb/s.

INTERNAL..... Specifies that the node supplies the required timing signals. If you select **INTERNAL**, NCU displays the **Clock Speed** parameter.

At **Clock Speed**, set the line data-transmission rate (1.25M, 833K, 625K, 420K, 230.4K, 125K, 64K, 56K, 38.4K, 32K, 19.2K, 9.6K, 7.2K, 4.8K, 2.4K, or 1.2K).

3. At **RTS/CTS Control**, enable or disable **RTS/CTS** flow control.

NO Disables **RTS/CTS** flow control.

YES Enables **RTS/CTS** flow control. Select **YES** if the connected device uses **RTS/CTS** flow control (for example, a connected modem).

You have edited the synchronous line parameters. At this point, you may perform either of the following procedures:

- Save the changes by selecting and then .

NCU displays the following window; press **[RETURN]** to clear it from the console:

Press return when done.

Sync line parameters stored.

- Edit the current circuit parameters by selecting .

Depending on the current circuit type, NCU displays windows, as follows:

- If the current circuit is a point-to-point circuit, NCU displays the **POINT TO POINT CIRCUIT** window. Go to *Section 3.8, Editing Point-to-Point Circuit Parameters* for instructions.
- If the current circuit is an X.25 point-to-point circuit, NCU displays the **X.25 POINT TO POINT** window. Go to *Section 3.9, Editing X.25 Circuit Parameters* for instructions
- If the current circuit is an X.25 PDN circuit, NCU displays the **X.25 PDN** window. Go to *Section 3.9, Editing X.25 Circuit Parameters* for instructions.
- If the current circuit is an X.25 DDN circuit, NCU displays the **X.25 DDN** window. Go to *Section 3.9, Editing X.25 Circuit Parameters* for instructions.

3.6 Editing T1-Line Parameters

You edit T1-line parameters from the **T1 LINE** window (see Figure 3-7). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.
Slot	Slot containing the T1 link module.
Connector	Specific connector on the T1 link module.
Physical Access Method	DS1 — physical access method of T1 lines.
Circuit1 Name	Name of first circuit associated with the T1 line.
Circuit2 Name	Name of the second circuit associated with the T1 line.

You may set four parameters (**Frame Type**, **Line Buildout**, **B8ZS Supported**, and **Clock Mode**) in the **T1 LINE** window, as follows:

1. At Frame Type, select the frame form that the attached T1 equipment requires.

D4 Specifies superframe transmission consisting of 12 individual D4 frames and providing two bits for rudimentary signalling function. Channel banks commonly use the **D4** framing format.

ESF Specifies superframe transmission consisting of 24 individual D4 frames and providing four bits of signalling functions, synchronization, and error checking. Some PBXs use **ESF** framing format.

2. At Line Buildout, enter the approximate length of the cable (from 1 to 655 feet) that connects the T1 device.

Signal attenuation correlates with line length. By setting **Line Buildout** to equal the approximate length of the cable connected to the T1 device, you condition generated signals (from 6 Volts to 3 Volts peak-to-peak) in order to overcome signal attenuation.

3. At B8ZS Supported, based on whether the connected device implements binary 8 zeros suppression, select the status of binary 8 zeros suppression.

YES Enables binary 8 zeros suppression.

NO Disables binary 8 zeros suppression.

The received signal synchronizes connected T1 devices; thus, an insufficient number of logical ones within the received signal may cause carrier loss. Binary 8 zeros suppression is a scheme to maintain sufficient ones-density within the T1 data stream.

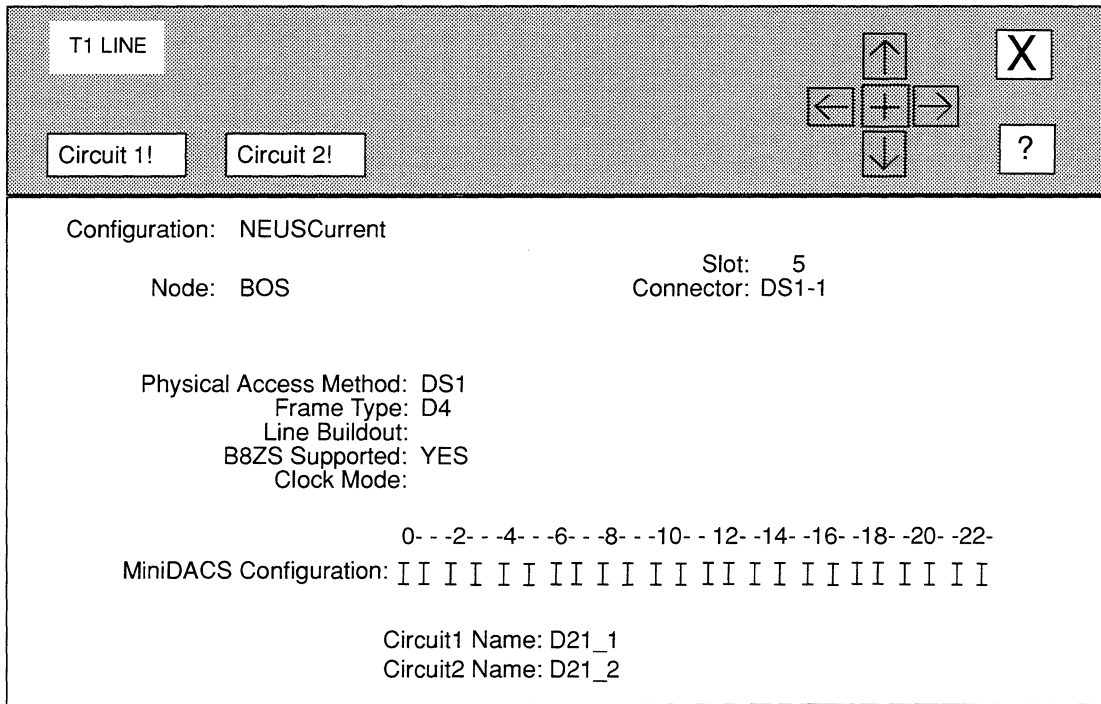


Figure 3-7. T1 LINE Window

4. At Clock Mode, select the source of the T1 transmit clock.

- MANUAL** Sets the clock source based on hardware jumpers on the T1 link module (refer to the *Wellfleet Installation Guide* for information on jumper settings).
- SLAVE** Overrides the jumper settings and places the T1 connection in *slave mode*.
- MASTER** Overrides the jumper settings and places the T1 connection in *master mode* so that the transmit clock is internally generated. Refer to the *Wellfleet Installation Guide* for additional information on clocking.

5. At MiniDACS Configuration, assign each T1 channel to a specific function, as follows:

- I..... Idles the channel.
- D..... Assigns the channel to data-pass through.
- Y..... Assigns the channel to voice-pass through.
- 1..... Assigns the channel to Circuit 1.
- 2..... Assigns the channel to Circuit 2.

For example, the network operator has assigned channel 3 below to data-pass through (D):

```

                                0- -2- -4- -6- -8- -10- -12- -14- -16- -18- -20- -22-
MiniDACS Configuration: I I  D I I I I I I I I I I I I I I I I I I
    
```

You have edited the T1 line parameters. At this point, you may perform either of the following procedures:

- Save the changes by selecting and then .
 NCU displays the following window; press **[RETURN]** to clear it from the console:

Press return when done.

T1 line parameters stored.

- Edit the current circuit parameters by selecting or .
 Depending on the command button you selected, NCU displays the **POINT TO POINT CIRCUIT** windows for either circuit 1 or circuit 2. Go to *Section 3.8, Editing Point-to-Point Circuit Parameters* for instructions.

3.7 Editing E1-Line Parameters

You edit E1-line parameters from the **E1 LINE** window (see Figure 3-8). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.

E1 LINE

Circuit 1!

Circuit 2!

Configuration: NEUSCurrent

Slot: 4

Node: BOS Connector: E1-1

Physical Access Method: CEPT

Clock Mode: MANUAL

HDB3 Supported: **YES**

Slot Configuration:

0 - - 3 - - 5 - - 7 - - 9 - - 11 - - 13 - - 15 - -

I I I I I I I I I I I I I I I I

17 - - 19 - - 21 - - 23 - - 25 - - 27 - - 29 - - 31

I I I I I I I I I I I I I I I I

Circuit1 Name: C31_1

Circuit2 Name: C31_2

Figure 3-8. E1 LINE Window

This Field (continued)	Displays
Slot	Slot containing the E1 link module.
Connector	Specific connector on the E1 link module.
Physical Access Method	CEPT — physical access method of E1 lines.
Circuit1 Name	Name of first circuit associated with the E1 line.
Circuit2 Name	Name of the second circuit associated with the E1 line.

You may set four parameters (**Clock Mode**, **HDB3 Supported**, and **Slot Configuration**) in the **E1 LINE** window, as follows:

1. At Clock Mode, select the source of the E1 transmit clock.

- MANUAL** Sets the clock source based on hardware jumpers on the E1 link module (refer to the *Wellfleet Installation Guide* for information on jumper settings).
- SLAVE** Overrides the jumper settings and places the E1 connection in *slave mode*.
- MASTER** Overrides the jumper settings and places the E1 connection in *master mode* so that the transmit clock is internally generated. Refer to the *Wellfleet Installation Guide* for additional information on clocking.

2. At HDB3 Supported, select the status of High Density Bipolar coding (a scheme to maintain sufficient ones-density within the E1 data stream) based on whether the connected device implements the scheme.

- YES** Enables High Density Bipolar coding.
- NO** Disables High Density Bipolar coding.

Synchronization between connected E1 devices is accomplished by means of the received signal; the signal edges provide the timing information. The presence of an extended string of logical zeros within the received signal can lose synchronization. To guard against such loss, HDB3 substitutes a known bit pattern for every occurrence of a string of four consecutive logical zeros within the CEPT data stream.

3. At Slot Configuration, assign each E1 channel to a specific function, as follows:

- I** Idles the channel.
- D** Assigns the channel to data-pass through.
- Y** Assigns the channel to voice-pass through.
- 1** Assigns the channel to Circuit 1.
- 2** Assigns the channel to Circuit 2.

Note

Channel 1 is reserved for signalling.

For example, the network operator has assigned channel **3** below to data-pass through (**D**):

```

0- - -3- - -5- - -7- - -9- - -11- - -13- -15- -
Slot Configuration: I I D I I I I I I I I I I I I I
17- - 19- - 21- - 23- - 25- - 27- - 29- - 31
I I I I I I I I I I I I I I I I I
    
```

You have edited the E1 line parameters. At this point, you may perform either of the following procedures:

- ❑ Save the changes by selecting and then .
 NCU displays the following window; press **[RETURN]** to clear it from the console:

Press return when done.

E1 line parameters stored.

- ❑ Edit the current circuit parameters by selecting or .
 Depending on the command button you selected, NCU displays the **POINT TO POINT CIRCUIT** windows for either circuit 1 or circuit 2. Go to *Section 3.8, Editing Point-to-Point Circuit Parameters* for instructions.

3.8 Editing Point-to-Point Circuit Parameters

You edit point-to-point circuit parameters from the **POINT TO POINT CIRCUIT** window (see Figure 3-9). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.
Slot	Number of the slot containing the line for which you are configuring the point-to-point circuit.
Connector	Specific physical connector that interfaces to the line for which you are configuring the point-to-point circuit.
Circuit Type	POINT TO POINT — NCU allows you to configure point-to-point circuits only in this window.
Remote Signal & Sense	ACTIVE - all point-to-point circuits are ACTIVE ; this parameter enables a proprietary protocol that detects any end-to-end connectivity failure.

POINT TO POINT CIRCUIT			
Configuration:	NEUSCurrent	Slot:	2
Node:	BOS	Connector:	COM1
Circuit Name:	S2-1	Remote Signal & Sense:	ACTIVE
Circuit Type:	POINT TO POINT	Quality of Service:	LLC1
Auto Enable:	YES		
Min. Frame Spacing:	2		
Data Link Protocol:	STANDARD		
Address:	DCE		

Figure 3-9. POINT TO POINT CIRCUIT Window

You may set six parameters (**Circuit Name**, **Auto Enable**, **Min. Frame Spacing**, **Data Link Protocol**, **Address** and **Quality of Service**) in the **POINT TO POINT** window, as follows:

1. **At Circuit Name**, enter the new circuit name.
2. **At Auto Enable**, specify the state of this point-to-point circuit when the node boots.

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this point-to-point circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.
You will subsequently need to enable this circuit manually with the NCL Interpreter after the node boots.
- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

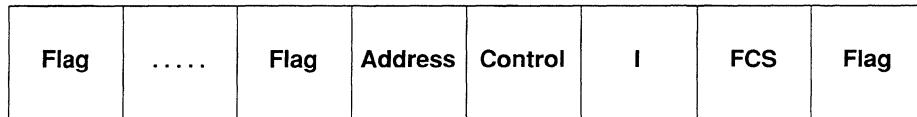
If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the point-to-point circuit.
- Select **NO** to disable the point-to-point circuit (you will subsequently need to enable the point-to-point circuit manually with the NCL Interpreter after the node boots).

3. At Min. Frame Spacing, enter the minimum number of flag sequences prefixed to an HDLC packet that the node transmits.

Figure 3-10 depicts the HDLC frame format. A variable number of 8-bit flag sequences prefix the HDLC packet and a single instance of the same flag terminates the packet. Therefore, the number of flags transmitted between sequential packets is equal to the sum of the constant 1 (the trailing flag) and the variable number of leading flags.

To calculate the minimum-frame spacing value, determine the minimum number of flags to transmit between each packet, and subtract one (to account for the terminating flag) from the minimum.



Key:

Flag Frame — 8-bit sequence (01111110)

Address Frame — 8/16 bits in length

Control Frame — 16 bits if Modulus is 128, else 8 bits

I (Information) Frame — Contains n bytes of data

FCS Frame — 32-bit frame check sequence

Flag Frame — 8-bit sequence (01111110)

Figure 3-10. HDLC Frame Format

4. At Data Link Protocol, enable or disable the proprietary Wellfleet protocol that provides a transparent point-to-point *pass-through* service for synchronous traffic across a Wellfleet node.

STANDARD Disables the proprietary Wellfleet protocol.

PROPRIETARY Enables the proprietary Wellfleet protocol (when you enable the protocol, the node restricts traffic across the circuit to explicitly configured addresses that terminate the point-to-point link; the node filters all other Bridge traffic from the *pass-through* interface).

Note

The pass-through protocol uses the Bridge as its transport mechanism; you must configure the Bridge on slots that have the protocol enabled.

Note

With the pass-through protocol enabled, the node ignores the **Quality of Service** and **Address** parameters, and disables the **Remote Signal & Sense** parameter. Consequently, the pass-through interface provides no end-to-end flow control or activity detection.

5. At Address, select the point-to-point address.

EXPLICIT Allows you to assign a unique address to each end of a point-to-point circuit. When you select **EXPLICIT**, NCU displays two parameters (**Local** and **Remote**):

— At **Local**, enter a unique decimal value from 00 through 99 (avoid the conventional address values of 01 or 03).

— At **Remote**, enter a unique value from 00 to 99.

DCE Assigns the circuit as the **DCE** end of the point-to-point circuit.

DTE Assigns the circuit as the **DTE** end of the point-to-point circuit.

The point-to-point address is a 1-byte value used in the address field of the HDLC packet (see Figure 3-10). Conventionally, one end of a point-to-point circuit is designated DCE and assigned an address of 01; while the other end of the point-to-point circuit is designated DTE and assigned a value of 03.

Conventional addressing, however, is inadequate when a common satellite link enables multiple communication channels. For example, Figure 3-11 depicts a common satellite relay-link that provides a point-to-point link between nodes **A** and **X**, **B** and **Y**, and **C** and **Z**.

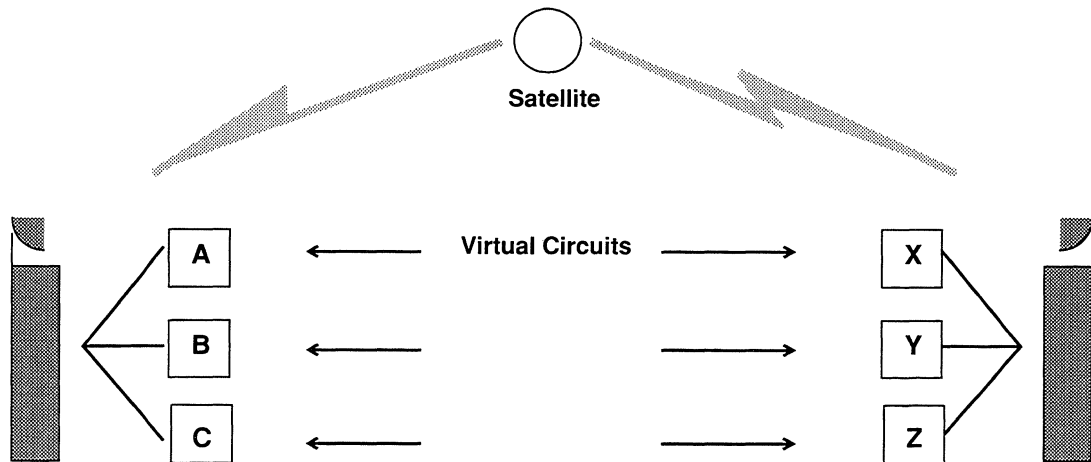


Figure 3-11. Satellite Broadcast (Sample Topology)

6. In the worst-case scenario, nodes **A**, **B**, and **C** are designated as DCE (address = 01), and nodes **X**, **Y**, and **Z** are designated as DTE (address = 03). Thus, if **A** transmits a frame across the virtual point-to-point circuit to **X**, the satellite broadcast is monitored not only by **X** (the intended recipient) but also by **Y** and **Z**. Because **X**, **Y**, and **Z** all perceive a properly-addressed frame, all three accept delivery and attempt to process frame contents with unpredictable results.

7. **At Quality of Service, select the link-level control level.**

LLC1 Specifies connectionless datagram service (the best-effort delivery model).

LLC2 Specifies connection-oriented service, which provides error detection and error recovery by retransmission. When you select **LLC2**, NCU displays six new parameters:

- At **Retry Counter**, enter the number of times the node will retransmit a frame after the **Retry Timer** expires.
- At **Retry Timer**, enter the number of seconds the node will wait for an acknowledgment after it issues a command.
- At **Connect Retries**, enter the number of times the node will transmit control messages to the remote

end of the circuit after **Retry Counter** and **Retry Timer** have expired.

Connect Retries operates with **Retry Counter** and **Retry Timer** to govern the number of retransmission attempts in the event of an unacknowledged packet.

After the **Retry Timer** expires, the node transmits up to N2 control messages in an attempt to get a response from the remote end of the circuit. If an acknowledgment is still outstanding, the node repeats the loop the number of times designated by **Connect Retries**. A value of 0 specifies infinite retries.

- At **Link Idle Timer**, enter the idle time (in seconds) after which the point-to-point circuit is disconnected.
- At **Modulus**, select the length (in bits) of the HDLC packet-control field.

8 Specifies an 8-bit control field.

128 Specifies a 16-bit control field.

The size of the control field (Figure 3-10 depicts the HDLC frame format) determines the maximum number of unacknowledged packets that may be pending at any one time. For example, an 8-bit control field provides three bits for message sequencing, and thus allows a maximum of seven outstanding packets. A 16-bit control field provides seven bits for sequencing, and thus allows a maximum of 127 unacknowledge packets.

- At **Window Size**, select the exact number (**Modulus** specified a maximum number) of packets that may be unacknowledge at any one time.

If **Modulus** is set to **8**, NCU provides the following responses: **7**, **1**, and **3**.

If **Modulus** is set to **128**, NCU provides the following responses: **7**, **15**, **31**, **63**, and **127**.

8. Select **X** and then **Save** .

NCU displays the following window; press **[RETURN]** to clear it from the console:

```
Press return when done.
Circuit parameters stored.
```

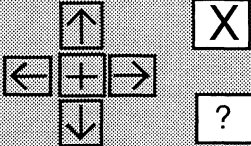
Depending on the type of line for which you configured the point-to-point circuit, NCU returns to either the **SYNC LINE**, **T1 LINE**, or **E1 LINE** window. If you made changes in these windows, select **X** and then **Save** ; otherwise, select **X** and then **Quit** .

3.9 Editing X.25 Circuit Parameters

The Wellfleet node provides three levels of X.25 services, as follows:

- X.25 Point-to-Point Service
Provides end-to-end connectivity between local and remote peer nodes. All bridging and routing software application modules can use point-to-point service. The **X.25 POINT TO POINT** window allows you to set X.25 point-to-point service parameters; go to *Section 3.9.1, Editing X.25 Point-to-Point Service Parameters* for instructions.
- X.25 Defense Data Network (DDN) Service
Provides end-to-end connectivity between the local node and a remote host or gateway equipped to support DDN “Standard Service”. Only the IP router uses DDN service; the router uses the service to transmit IP datagrams over the DDN. The **X.25 DDN** window allows you to set X.25 DDN service parameters; go to *Section 3.9.2, Editing X.25 DDN Service Parameters* for instructions.
- X.25 Public Data Network (PDN) Service
Provides end-to-end connectivity between the local node and a remote host or device equipped to support Internet RFC 877 X.25 service. (RFC 877 specifies a standard for IP datagram transmissions over public data networks). Only the IP Routers uses PDN service; the router uses the service to transmit IP datagrams over public data networks. The **X.25 PDN** window allows you to set X.25 PDN service parameters; go to *Editing 3.9.3, Setting X.25 PDN Service Parameters* for instructions.

X.25 POINT TO POINT



Virtual Circuits!

Construct Bitmap!

Configuration Name: NEUSCurrent
Node: BOS

Circuit Name: S42

Auto Enable: YES

PDN: TELENET

Local DTE Addr: 01

T1: 30
N2: 20

Low LCN: 1
High LCN: 32
Min Frame Spacing: 2
Max Queue Size: 10

Default Pkt Window: 2
Default Pkt Size: 128

Figure 3-12. X.25 POINT TO POINT Window

3.9.1 Editing X.25 Point-to-Point Service Parameters

An X.25 circuit enables you to access public data networks. You can then use the network's packet-switching facilities to transfer packets to a remote peer node. You must configure X.25 point-to-point service for any circuit providing end-to-end connectivity to a remote node through an intermediate public data network. NCU automatically configures X.25 point-to-point service for circuits requiring such service when you generate the node *config* file. This section describes how to edit X.25 point-to-point service parameters.

You edit X.25 point-to-point service parameters from the **X.25 POINT TO POINT** window (see Figure 3-12). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.

You may set the remaining parameters in the **X.25 POINT TO POINT** window, as follows:

1. **At Circuit Name, enter the new name of the circuit.**
2. **At Auto Enable, specify the state of this X.25 circuit when the node boots.**

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this X.25 circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.
You will subsequently need to enable this circuit manually with the NCL Interpreter after the node boots.
- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the X.25 circuit.
- Select **NO** to disable the X.25 circuit (you will subsequently need to enable the point-to-point circuit manually with the NCL Interpreter after the node boots).

3. **At PDN, select the X.25 services supplier.**

TELENET Specifies that TELENET is the X.25 services supplier.

DDN Specifies that DDN is the X.25 services provider.

UK-PSS Specifies that UK-PSS is the X.25 services provider.

OTHER Specifies a non-specific public data network.

USE BITMAP Specifies that you wish to configure certain low-level attributes that specify the interface between the node and a PDN. If you select **USE BITMAP**, Construct Bitmap! becomes active.

4. **At Local DTE Address, enter a network-supplied decimal number (X.121 address) that identifies the interface between the node and the X.25 network.**

Note

The network operator defined this parameter when the node was originally connected to the network segment.

5. At T1, enter the number of seconds a frame can remain unacknowledged.

Note

Express the **T1** value in tenths of a second. For example, a value of 30 sets the timer to 3 seconds.

Typically, a **T1** value in excess of 3 seconds is required only if your network connection has a substantial path delay (for example, if the connection is accomplished with a satellite link). Under these conditions, **T1** must have a value greater than the round-trip frame-transmission time, plus the time required to process the frame at the receiving end.

If you set **T1** with too small a value, you reduce throughput with unnecessary frame retransmission. If you set **T1** with too great a value, X.25 takes an excessive length of time to detect lost frames.

6. At N2, enter the number of times (from 1 to 255) a frame is retransmitted before the circuit is reset.

If a frame remains unacknowledged at the expiration of the **T1** timer, the node retransmits the outstanding frame up to **N2** times, with each retransmission requesting an immediate acknowledgment. If the frame remains unacknowledged after **N2** retries, the node resets the X.25 point-to-point service.

7. At Low LCN, set the minimum logical channel number (from 0 to 999).

The logical channel number, also called the logical channel identifier, is a decimal number that identifies the dedicated switched two-way virtual circuit being used between the local and remote nodes. The actual number of dedicated virtual circuits available to you and their numeric identifiers is generally governed by your subscription agreement with the X.25 provider.

8. At High LCN, set the maximum logical channel number (from 0 to 999)

The node supports up to 32 dedicated switched two-way virtual circuits for each PDN connection. To calculate the **High LCN** value, add the number of dedicated switched two-way virtual circuits provided for by your X.25 subscription agreement to the value assigned to **Low LCN**; then, decrease the number by 1.

Note

Because the logical channel number range for the physical link determines the number of virtual connections that can be established, the value you assign to **Low LCN** and **High LCN** must be identical on both sides of the X.25 physical link.

9. **At Min Frame Spacing, enter the minimum number of flag sequences prefixed to an X.25 packet that the node transmits.**

A variable number of 8-bit flag sequences prefix an X.25 frame. A single instance of the same flag terminates the frame. Therefore, the number of flags transmitted between sequential frames is equal to the constant 1 (the trailing flag) plus the (variable) number of leading flags. Thus, to obtain the **Min Frame Spacing** value, determine the number of flags to prefix to each frame and subtract 1 from this number.

10. **At Max Queue Size, enter the maximum number of packets (from 1 to 999) allowed in the transmit queue of each individual dedicated virtual circuit.**

If the value specified by Max Queue Size is exceeded, the node “clips” (discards) the oldest packet(s) in the transmit queue.

Note

To avoid queue “clipping”, set **Max Queue Size** to 0.

11. **At Default Pkt Window, enter the maximum number (from 1 to 7) of outstanding (unacknowledged) packets.**

Note

You can override this default value on a virtual-circuit basis when you configure individual virtual circuits.

12. **At Default Pkt Size, enter the maximum number of bytes (from 16 to 2048) in the data field of an X.25 packet.**

Note

You can override this default value on a virtual-circuit basis when you configure individual virtual circuits.

At this point, you may:

- Configure point-to-point virtual circuits by selecting **Virtual Circuits!** to display the **X.25 VIRTUAL CIRCUITS** window; go to *Section 3.9.1.1, Configuring Point-to-Point Virtual Circuits* for instructions.
- Configure bitmap values by selecting **Construct Bitmap!** to display the **X.25 BITMAP VALUES** window (but only if you set **PDN to USE BITMAP**); go to *Section 3.9.4, Configuring Bitmap Values* for instructions.

- ❑ Exit the **X.25 POINT TO POINT** window by selecting **X** and **Save** to save any changes you made, or by selecting **X** and **Quit** to exit the window without saving changes.

3.9.1.1 Configuring Point-To-Point Virtual Circuits

X.25 point-to-point service operates by establishing dedicated switched virtual circuits between the local node and a remote peer. These circuits are established during the node initialization process and remain permanently available unless taken out of service with the NCL **DISABLE** command. Calls are established by the exchanges of a *call request* packet (issued by the node with the higher X.121 address) and a *call confirm* packet (issued by the node with the lower X.121 address).

Each X.25 point-to-point service circuits supports up to 32 dedicated virtual circuits. This section describes how to add, modify, and delete X.25 point-to-point virtual circuits.

3.9.1.1.1 Adding Point-To-Point Virtual Circuits

You add point-to-point virtual circuits from the **X.25 VIRTUAL CIRCUITS** window (see Figure 3-13), as follows:

1. Select **Virtual Circuits** to display the menu in Figure 3-13.
2. Select **Add** to display the **ADD X.25 VIRTUAL CIRCUIT** window (see Figure 3-14).

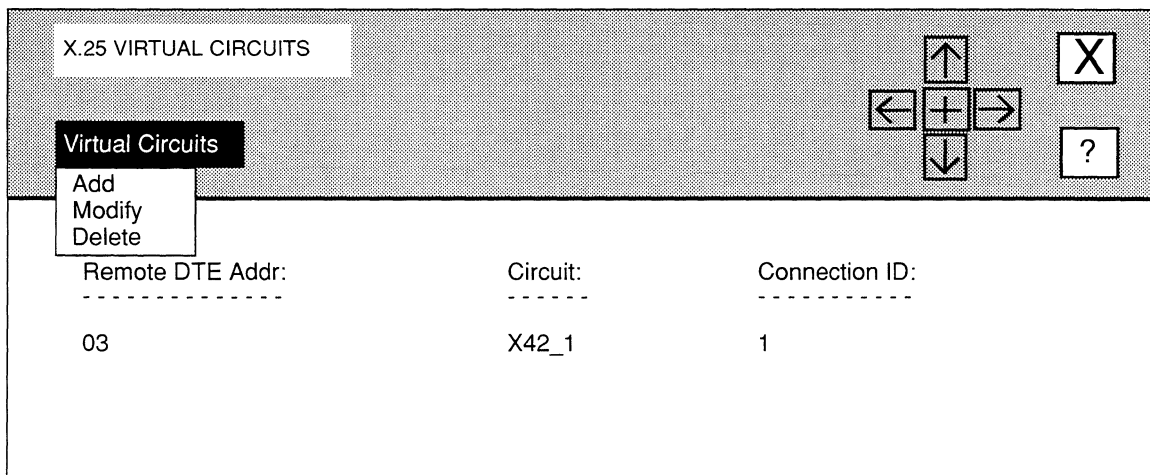


Figure 3-13. X.25 VIRTUAL CIRCUITS Window

ADD X.25 VIRTUAL CIRCUIT

X

?

Circuit Name:

Remote DTE Address:

Connection ID: 1

Flow Control: DEFAULT

Figure 3-14. ADD X.25 VIRTUAL CIRCUIT Window

- 3. At Circuit Name, enter the name of the dedicated virtual circuit.**

Because you use virtual circuit names to configure application software modules, it is very important that you maintain an accurate record of virtual circuit names.
- 4. At Remote DTE Address, enter the network-supplied decimal number (X.121 address) that identifies the interface between the remote node and the X.25 network.**
- 5. At Connection ID, enter a number (from 1 to 99) that identifies the connection.**

Assigning a value to **Connection ID** enables the establishment of multiple, parallel dedicated virtual circuits between two nodes. Such parallel circuits may result in higher throughput, because of the increased window size that multiple virtual circuits afford.

If you are establishing only one dedicated virtual circuit between the local node and the remote peer designated by **Remote DTE Address**, enter **1** at **Connection ID**. When you configure the remote node, you must assign it a **Connection ID** of 1 also.

If you are establishing multiple dedicated virtual circuits between the local and remote node, you must assign a unique **Connection ID** to each virtual circuit. When you configure the remote node, you must assign it identical **Connection ID** values.

6. **At Flow Control, enable or disable the Flow Control Parameter Negotiation (a subscription option available from most X.25 service providers).**

DEFAULT Disables negotiation in *call request* packets. If you select **DEFAULT**, the configured values for **Default Pkt Window** and **Default Pkt Size** serve as the defaults across the circuit. Also, you must ensure that the X.25 switching device (DCE) to which this circuit connects has also disabled flow control. Additionally, you must ensure that the values you select for **Default Pkt Window** and **Default Pkt Size** match those of the DCE. And you must ensure that the remote DTE has also disabled negotiation and that its assigned values for **Default Pkt Window** and **Default Pkt Size** match those of the local DTE.

NEGOT..... Enables negotiation so that the window and packet size are negotiated on a virtual-circuit basis. If you select **NEGOT**, NCU displays the following parameters which you must set, as follows:

- At **Negotiated Pkt Window**, enter a number (from 1 to 7) that specifies the window size that appears in the facilities field of *Call Request* packets originated on **Circuit Name**.
- At **Negotiated Pkt Size**, enter a number (from 128 to 2048) that specifies the packet size that appears in the facilities field of *Call Request* packets originated on **Circuit Name**.

Note

If you select **NEGOT**, you can maximize virtual circuit efficiency by ensuring that the remote DTE has also enabled negotiation and that its assigned values for **Negotiated Pkt Window** and **Negotiated Pkt Size** match those of the local DTE. In such an instance, the negotiation proceeds, as follows: (1) the node with the higher X.121 address issues a *Call Request* packet that specifies Flow Control Parameter Negotiation and includes the values for **Negotiated Pkt Window** and **Negotiated Pkt Size** in the facilities field of the packet; (2) the called node (the one with the lower X.121 address) performs simple boundary checking to verify that the negotiated parameters are within acceptable ranges and issues a *Call Confirm* packet.

Note

X.25 point-to-point services supports *Throughput Negotiation* (with an initial value of 48,000 bits per second) on incoming calls. *Throughput Negotiation* is not initiated on outgoing calls.

3.9.1.1.2 Modifying Point-To-Point Virtual Circuits

You modify point-to-point virtual circuits from the **X.25 VIRTUAL CIRCUITS** window (see Figure 3-13), as follows:

1. Select the virtual circuit you wish to modify under **Remote DTE Addr.**
2. Select to display the menu in Figure 3-13.
3. Select to display the **ADD X.25 VIRTUAL CIRCUIT** window (see Figure 3-14), which displays the current parameters for that virtual circuit you select in step 1.
4. Go to *Section 3.9.1.1.1, Adding Point-To-Point Virtual Circuits* for information on how to set the parameters.

3.9.1.1.3 Deleting Point-To-Point Virtual Circuits

You delete point-to-point virtual circuits from the **X.25 VIRTUAL CIRCUITS** window (see Figure 3-13), as follows:

1. Select the virtual circuit you wish to delete under **Remote DTE Addr.**
2. Select to display the menu in Figure 3-13.
3. Select .

NCU deletes the virtual circuit.

3.9.2 Editing X.25 DDN Service Parameters

You edit X.25 DDN service parameters from the **X.25 DDN** window (see Figure 3-15). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.

You may set the remaining parameters in the **X.25 DDN** window, as follows:

1. At **Circuit Name**, enter the new name of the circuit.
2. At **Auto Enable**, specify the state of this X.25 circuit when the node boots.

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this X.25 circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.

You will subsequently need to enable this circuit manually with the NCL Interpreter after the node boots.

X.25DDN			
Configuration Name:	NEUSCurrent		
Node:	BOS		
Circuit Name:	S44	Max Conns/Dest:	1
Auto Enable:	YES	Min Idle Time:	10
		Max Idle Time:	60
IP Circuit:	S44_DDN		
	T1: 30		
	N2: 20		
	Low LCN: 1	Flow Control:	NEGOT
	High LCN: 32	Pkt Window:	1
Min Frame Spacing:	2	Pkt Size:	16
Max Queue Size:	0	Precedence:	DEFAULT

Figure 3-15. X.25 DDN Window

- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

If global **Auto Enable** is set to **YES**, do one of the following:

 - Select **YES** to enable the X.25 circuit.
 - Select **NO** to disable the X.25 circuit (you will subsequently need to enable the point-to-point circuit manually with the NCL Interpreter after the node boots).
- 3. At IP Circuit, enter a new IP circuit name.**
- IP Circuit Name** further identifies the circuit providing X.25 DDN service. You will use this name (not the name previously specified at **Circuit Name**) when you configure circuit groups and when you configure the IP router for X.25 DDN service.

4. At T1, enter the number of seconds a frame can remain unacknowledged.

Note

Express the **T1** value in tenths of a second. For example, a value of 30 sets the timer to 3 seconds.

Typically, a **T1** value in excess of 3 seconds is required only if your network connection has a substantial path delay (for example, if the connection is accomplished with a satellite link). Under these conditions, **T1** must have a value greater than the round-trip frame-transmission time, plus the time required to process the frame at the receiving end.

If you set **T1** with too small a value, you reduce throughput with unnecessary frame retransmission. If you set **T1** with too great a value, X.25 takes an excessive length of time to detect lost frames.

5. At N2, enter the number of times (from 1 to 255) a frame is retransmitted before the circuit is reset.

If a frame remains unacknowledged at the expiration of the **T1** timer, the node retransmits the outstanding frame up to **N2** times, with each retransmission requesting an immediate acknowledgment. If the frame remains unacknowledged after **N2** retries, the node resets the X.25 point-to-point service.

6. At Low LCN, set the minimum logical channel number (from 0 to 999).

The logical channel number is a decimal number that identifies the switched virtual circuit being used to connect the local node and a remote host or gateway.

7. At High LCN, set the maximum logical channel number (from 0 to 999)

The node supports up to 254 logical channels per slot with each logical channel corresponding to a two-way switched virtual circuit. Upon initialization, the node first allocates permanent, dedicated, point-to-point, logical channels (up to the maximum of 32 for each PDN connection). After logical channels have been allocated to X.25 point-to-point service, the node makes remaining logical channels available to X.25 DDN or X.25 PDN services. Available channels are distributed equally among DDN/PDN circuits.

Determine the value of **High LCN**, as follows:

- If you are configuring only X.25 DDN and/or X.25 PDN service on the current slot, use the following formula to calculate **High LCN**:

$$\text{High LCN} = [254/N] + \text{Low_LCN} - 1$$

where:

- N** Is the number of X.25 DDN and X.25 PDN physical network connections on the slot.
- [254/N]** Is the integer quotient of 254 divided by **N**.
- Low_LCN** Is the value assigned to the **Low LCN** parameter.

- If you are configuring a combination of X.25 DDN and/or X.25 PDN service in conjunction with X.25 point-to-point service, use the following formula to calculate **High LCN**:

$$\text{High LCN} = [(254 - V) / N] + \text{Low_LCN} - 1$$

where:

- V** Is the number of dedicated point-to-point virtual circuits on the slot.
- N** Is the integer of X.25 DDN and X.25 PDN network connections on the slot.
- [254/N]** Is the integer quotient of 254 divided by **N**.
- Low_LCN** Is the value assigned to the **Low LCN** parameter.

Note

Both the **Low LCN** and the **High LCN** values must conform to the logical channel numbers allocated by the DDN.

8. At **Min Frame Spacing**, enter the **minimum number of flag sequences prefixed to an X.25 packet that the node transmits**.

A variable number of 8-bit flag sequences prefix an X.25 frame. A single instance of the same flag terminates the frame. Therefore, the number of flags transmitted between sequential frames is equal to the constant 1 (the trailing flag) plus the (variable) number of leading flags. Thus, to obtain the **Min Frame Spacing** value, determine the number of flags to prefix to each frame and subtract 1 from this number.

Note

X.25 DDN service supports *Throughput Negotiation* (with an initial value of 48,000 bits per second) on incoming calls. *Throughput Negotiation* is not initiated on outgoing calls.

9. **At Max Queue Size, enter the maximum number of packets (from 1 to 999) allowed in the transmit queue of each individual dedicated virtual circuit.**

If the value specified by Max Queue Size is exceeded, the node “clips” (discards) the oldest packet(s) in the transmit queue.

Note

To avoid queue “clipping”, set **Max Queue Size** to 0.

10. **At Max Conns/Dest, enter the maximum number (from 1 to 4) of connections that can be simultaneously established with a single destination host or gateway.**

The X.25 DDN service clears any incoming calls that would exceed this limit. Similarly, the X.25 DDN service makes no attempt to place outgoing calls that would exceed this limit. Establishing multiple connections with a single destination may improve throughput by increasing the window size.

11. **At Min Idle Time, enter the minimum number of seconds of circuit inactivity (no IP datagrams sent or received) before a circuit can be cleared and reused for a call to another destination.**

12. **At Max Idle Time, enter the maximum number of seconds that a circuit can remain idle.**

After the max idle timer expires, the node clears the circuit. This parameter is intended to minimize CPU and network overhead during periods of low datagram traffic. If **Min Idle Time** is set to 0, this parameter is ignored.

13. **At Flow Control, enable or disable Flow Control Parameter Negotiation (a facility available from the DDN that allows packet window and packet size to be set on a per-call basis.**

DEFAULT Disables negotiation in locally originated *Call Request* packets. When you select **DEFAULT**, the configured values for **Pkt Window** and **Pkt Size** serve as the defaults for each call. Also, you must ensure that the X.25 DDN switching device (DCE) to which this circuit connects has also disabled flow control. Additionally, you must ensure that the DCE’s **Pkt Size** and **Pkt Window** values are identical to yours.

NEGOT..... Enables negotiation so that all locally originated call request packets specify Flow Control Parameter Negotiation. If you select **NEGOT**, you must ensure that the X.25 DDN switching device (DCE) to which this circuit connects has also enabled flow control.

Additionally, you must ensure that the values you select for **Pkt Window** and **Pkt Size** match those of the DCE.

Note

If you set **Flow Control** to **DEFAULT**, you should ensure that the remote DTE has also disabled negotiation and that its assigned values for **Pkt Window** and **Pkt Size** match those of the local DTE.

14. **At Pkt Window, enter the maximum number (from 1 to 7) of allowable outstanding (unacknowledged) packets.**

If you set **Flow Control** to **NEGOT**, the value you assign to **Pkt Window** will appear in the facilities field of all *Call Request* packets that originate at **Circuit Name**. The X.25 end nodes may negotiate a lower or higher value on a per-call basis. If you set **Flow Control** to **DEFAULT**, the value you assign to **Pkt Window** specifies the maximum number of outstanding unacknowledged packets for all calls across **Circuit Name**.

15. **At Pkt Size, enter the maximum number of bytes (from 16 to 2048) in the data field of an X.25 packet.**

If you set **Flow Control** to **NEGOT**, the value you assign to **Pkt Size** will appear in the facilities field of all *Call Request* packets that originate at **Circuit Name**. The X.25 end nodes may negotiate a lower or higher value on a per-call basis. If you set **Flow Control** to **DEFAULT**, the value you assign to **Pkt Size** specifies the maximum packet length for all calls across **Circuit Name**.

16. **At Precedence, enable or disable a request for “Level 0” precedence.**

DEFAULT Disables precedence requests.

NEGOT..... Enables a request for “Level 0” precedence in all outgoing calls.

17. Select and then to save the X.25 DDN service parameters.

3.9.3 Editing X.25 PDN Service Parameters

X.25 PDN service provides X.25 service as specified in RFC 877. Such access enables the IP Router to use the 877-conforming network’s packet-switching facilities to transfer IP datagrams to a remote host or gateway (the remote host or gateway need not be a peer node). All outgoing calls specific “Internet Protocol”.

You edit X.25 PDN service parameters from the **X.25 PDN** window (see Figure 3-16). The window displays the following information which you may not change:

This Field	Displays
Configuration	Current network configuration.
Node	Current node.

You may set the remaining parameters in the **X.25 PDN** window, as follows:

1. **At Circuit Name, enter the new name of the circuit.**
2. **At Auto Enable, specify the state of this X.25 circuit when the node boots.**

This circuit-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable this X.25 circuit when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the node unconditionally disables all circuits (*including this circuit*) and all software modules.
You will subsequently need to enable this circuit manually with the NCL Interpreter after the node boots.

- When global **Auto Enable** is set to **YES**, the node conditionally enables all circuits (*including this circuit*) and all software modules.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the X.25 circuit.
- Select **NO** to disable the X.25 circuit (you will subsequently need to enable the point-to-point circuit manually with the NCL Interpreter after the node boots).

3. **At IP Circuit, enter a new IP circuit name.**

IP Circuit Name further identifies the circuit providing X.25 PDN service. You will use this name (not the name previously specified at **Circuit Name**) when you configure circuit groups and when you configure the IP router for X.25 PDN service.

4. **At Local DTE Address, enter a network-supplied decimal number (X.121 address) that identifies the interface between the node and the X.25 network.**

Note

The network operator defined this parameter when the node was originally connected to the network segment.

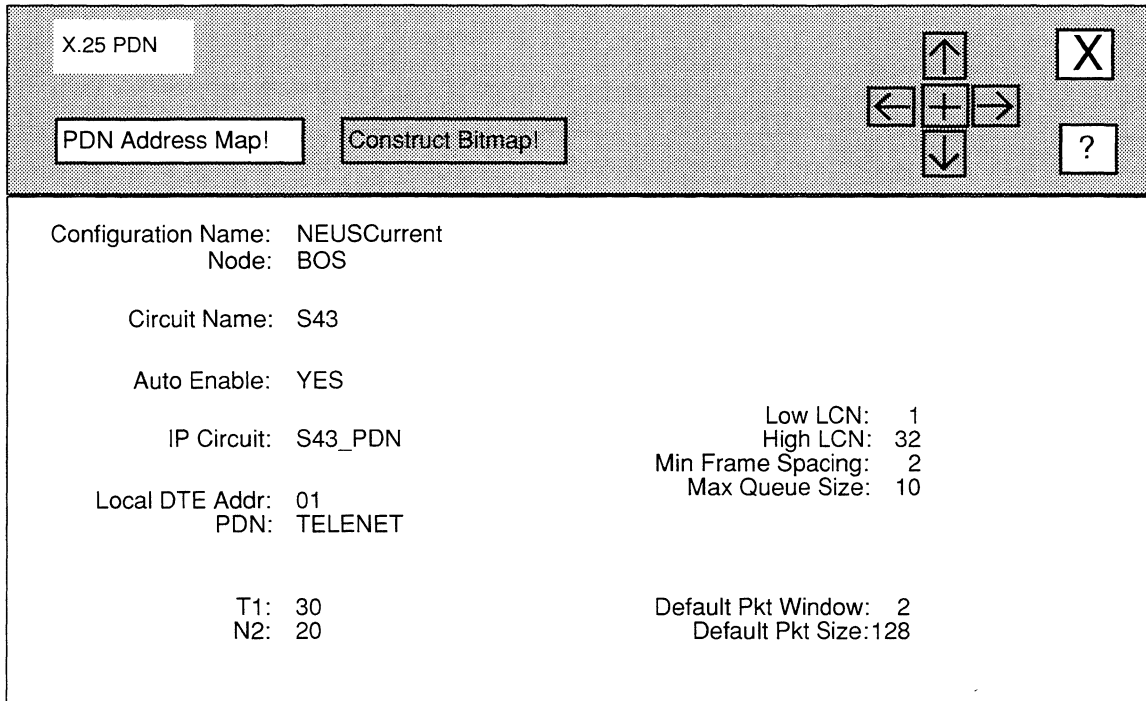


Figure 3-16. x.25 PDN Window

5. At PDN, select the X.25 services supplier.

TELENET Specifies that TELENET is the X.25 services supplier.

DDN Specifies that DDN is the X.25 services provider.

UK-PSS Specifies that UK-PSS is the X.25 services provider.

OTHER Specifies a non-specific public data network.

USE BITMAP Specifies that you wish to configure certain low-level attributes that specify the interface between the node and a PDN. If you select **USE BITMAP**, **Construct Bitmap!** becomes active.

6. At T1, enter the number of seconds a frame can remain unacknowledged.

Note

Express the **T1** value in tenths of a second. For example, a value of 30 sets the timer to 3 seconds.

Typically, a **T1** value in excess of 3 seconds is required only if your network connection has a substantial path delay (for example, if the connection is accomplished with a satellite link). Under these conditions, **T1** must have a value greater than the round-trip frame-transmission time, plus the time required to process the frame at the receiving end.

If you set **T1** with too small a value, you reduce throughput with unnecessary frame retransmission. If you set **T1** with too great a value, X.25 takes an excessive length of time to detect lost frames.

7. At N2, enter the number of times (from 1 to 255) a frame is retransmitted before the circuit is reset.

If a frame remains unacknowledged at the expiration of the **T1** timer, the node retransmits the outstanding frame up to **N2** times, with each retransmission requesting an immediate acknowledgment. If the frame remains unacknowledged after **N2** retries, the node resets the X.25 point-to-point service.

8. At Low LCN, set the minimum logical channel number (from 0 to 999).

The logical channel number is a decimal number that identifies the switched virtual circuit being used to connect the local node and a remote host or gateway.

9. At High LCN, set the maximum logical channel number (from 0 to 999)

The node supports up to 254 logical channels per slot with each logical channel corresponding to a two-way switched virtual circuit. Upon initialization, the node first allocates permanent, dedicated, point-to-point, logical channels (up to the maximum of 32 for each PDN connection). After logical channels have been allocated to X.25 point-to-point service, the node makes remaining logical channels available to X.25 DDN or X.25 PDN services. Available channels are distributed equally among DDN/PDN circuits.

Determine the value of **High LCN**, as follows:

- If you are configuring only X.25 DDN and/or X.25 PDN service on the current slot, use the following formula to calculate **High LCN**:

$$\text{High LCN} = [254/N] + \text{Low_LCN} - 1$$

where:

N Is the number of X.25 DDN and X.25 PDN physical network connections on the slot.

[254/N] Is the integer quotient of 254 divided by **N**.

Low_LCN Is the value assigned to the **Low LCN** parameter.

- ❑ If you are configuring a combination of X.25 DDN and/or X.25 PDN service in conjunction with X.25 point-to-point service, use the following formula to calculate **High LCN**:

$$\text{High LCN} = [(254 - V) / N] + \text{Low_LCN} - 1$$

where:

V Is the number of dedicated point-to-point virtual circuits on the slot.

N Is the integer of X.25 DDN and X.25 PDN network connections on the slot.

[254/N] Is the integer quotient of 254 divided by **N**.

Low_LCN Is the value assigned to the **Low LCN** parameter.

10. At Min Frame Spacing, enter the minimum number of flag sequences prefixed to an X.25 packet that the node transmits.

A variable number of 8-bit flag sequences prefix an X.25 frame. A single instance of the same flag terminates the frame. Therefore, the number of flags transmitted between sequential frames is equal to the constant 1 (the trailing flag) plus the (variable) number of leading flags. Thus, to obtain the **Min Frame Spacing** value, determine the number of flags to prefix to each frame and subtract 1 from this number.

11. At Max Queue Size, enter the maximum number of packets (from 1 to 99) allowed in the transmit queue of each individual dedicated virtual circuit.

If the value specified by Max Queue Size is exceeded, the node “clips” (discards) the oldest packet(s) in the transmit queue.

Note

To avoid queue “clipping”, set **Max Queue Size** to 0.

12. At Default Pkt Window, enter the maximum number (from 1 to 7) of outstanding (unacknowledged) packets.

Note

You can override this default value on a virtual-circuit basis when you configure individual virtual circuits.

- 13. At Default Pkt Size, enter the maximum number of bytes (from 16 to 2048) in the data field of an X.25 packet.**

Note

You can override this default value on a virtual-circuit basis when you configure individual virtual circuits.

At this point, you may:

- Build the X.25 PDN address map, by selecting to display the **X.25 PDN ADDRESS MAP** window; go to *Section 3.9.3.1, Building the X.25 PDN Address Map* for instructions.
- Configure bitmap values by selecting to display the **X.25 BITMAP VALUES** window (but only if you set **PDN** to **USE BITMAP**); go to *Section 3.9.4, Configuring Bitmap Values* for instructions.
- Exit the **X.25 PDN** window by selecting and to save any changes you made, or by selecting and to exit the window without saving changes.

3.9.3.1 Building the X.25 PDN Address Map

The X.25 PDN address map consists of a series of address pairs. Each pair associates a destination X.121 address with a 32-bit IP address. The address map can contain up to 256 address pairs for all PDN circuits.

You build the X.25 PDN address map from the **X.25 PDN ADDRESS MAP** window (see Figure 3-17. This window displays the current contents of the X.25 PDN address map. The following sections describe how to add, modify, and delete map entries.

3.9.3.1.1 Adding Map Entries

You add map entries from the **X.25 PDN ADDRESS MAP** window, as follows:

1. Select to display the menu depicted in Figure 3-17.
2. Select to display the **ADD PDN ADDRESS MAP** window (see Figure 3-18).
3. **At IP Address, enter the 32-bit IP address (in dotted decimal notation) of a recipient of IP datagrams that the X.25 PDN service transmits.**
4. **At X.121 Address, enter the X.121 address that corresponds to the IP address you just entered.**

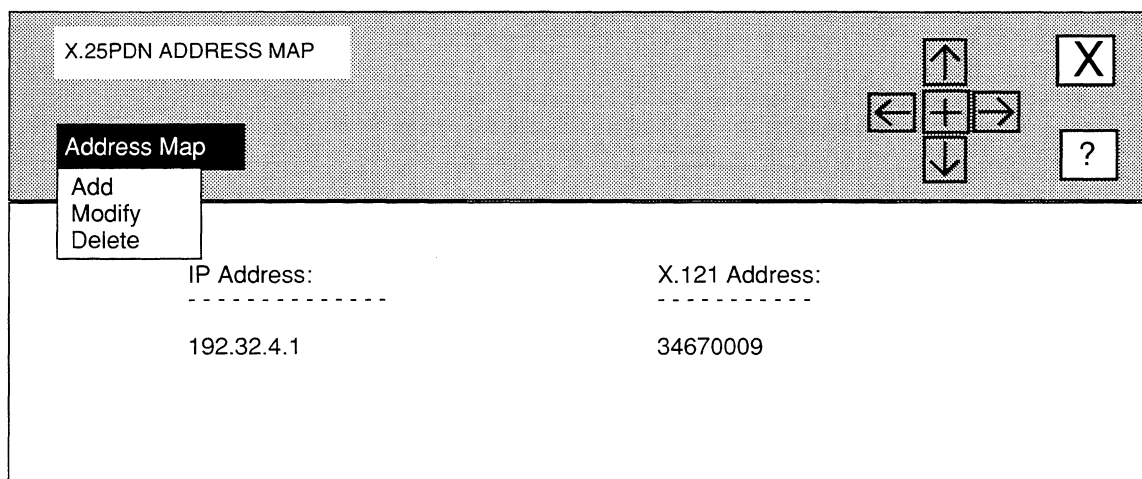


Figure 3-17. X.25 PDN ADDRESS MAP Window

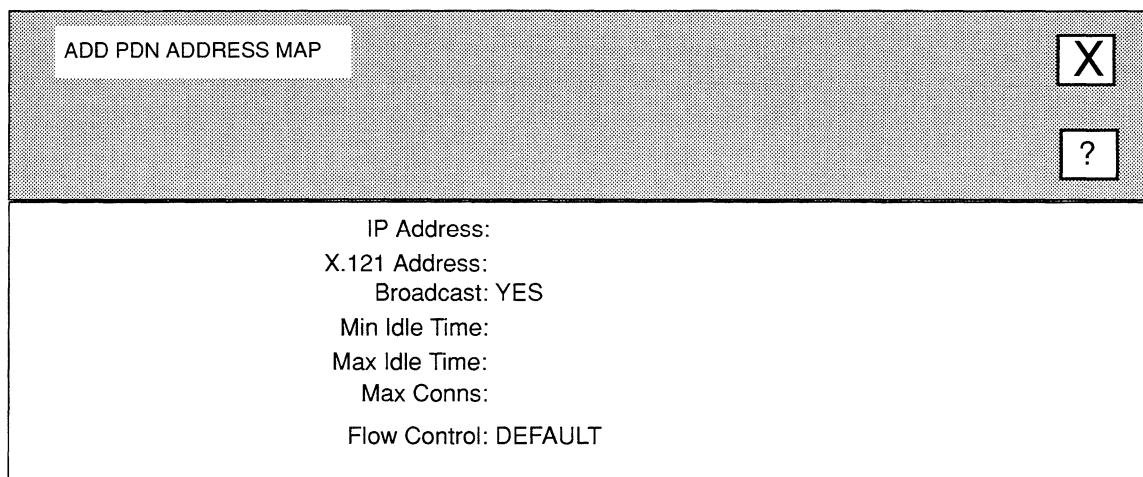


Figure 3-18. ADD PDN ADDRESS MAP Window

5. **At Broadcast, select whether the X.25 service sends broadcast messages to the IP address you entered in step 3.**

YES Specifies the X.25 service broadcasts messages to the IP address you entered in step 3.

NO Specifies the X.25 service does not broadcast messages to the IP address you entered in step 3.

6. **At Min Idle Time, enter the minimum period (in seconds) of circuit inactivity allowed before a circuit can be cleared and reused for a call to another destination; circuit inactivity is the time in which no IP datagrams are sent to or received from IP Address.**

Note

To prevent a connection to **IP Address** from ever being cleared once such a connection is established, enter a value of **0** (which implies an infinite idle time) at **Min Idle Time**.

7. **At Max Idle Time, enter the maximum period (in seconds) that a circuit may remain idle.**

Once **Max Idle Time** expires, the node clears the circuit. This parameter is intended to minimize CUP and network overhead during periods of low datagram traffic.

Note

If you set **Min Idle Time** to **0**, the node ignores **Max Idle Time**.

8. **At Max Conns, enter the maximum number (from 1 to 4) of connections that can be simultaneously established with a single destination host or gateway.**

The X.25 PDN service clears any incoming calls that would exceed this limit. Similarly, the X.25 PDN service makes no attempt to place outgoing calls that would exceed this limit. Establishing multiple connections with a single destination may improve throughput by increasing the window size.

9. **At Flow Control, enable or disable Flow Control Parameter Negotiation (a facility available from the PDN that allows packet window and packet size to be set on a per-call basis.**

DEFAULT Disables negotiation in *Call Request* packets. When you select **DEFAULT**, the configured values for **Pkt Window** and **Pkt Size** serve as the defaults for each call. Also, you must ensure that the X.25 DDN switching device (DCE) to which this circuit connects has also disabled flow control. Additionally, you must ensure that the DCE's **Pkt Size** and **Pkt Window** values are identical to yours.

NEGOT..... Enables negotiation so that all locally originated call request packets specify Flow Control Parameter Negotiation. If you select **NEGOT**, you must ensure that the X.25 DDN switching device (DCE) to which this circuit connects has also enabled flow control. Additionally, you must ensure that the values you select for **Pkt Window** and **Pkt Size** match those of the DCE. When you select **NEGOT**, NCU displays two parameters which you must set, as follows:

- At **Negotiated Pkt Window**, enter a value from 1 through 7 that specifies the window size that appears in the facilities field of *Call Request* packets originated on the PDN **Circuit Name**.
- At **Negotiated Pkt Size**, enter a value from 16 through 2048 that specifies the packet size that appears in the facilities field of *Call Request* packets originated on the PDN **Circuit Name**.

Note

If you set **Flow Control** to **DEFAULT**, you should ensure that the remote DTE has also disabled negotiation and that its assigned values for **Pkt Window** and **Pkt Size** match those of the local DTE.

Note

If you set **Flow Control** to **NEGOT**, you can maximize virtual circuit efficiency by ensuring that the remote DTE has also enabled negotiation and that its assigned values for **Negotiated Pkt Window** and **Negotiated Pkt Size** match those of the local DTE. In such an instance, the negotiation proceeds as follows: (1) the calling node issues a *Call Request* packet that specifies Flow Control Parameter Negotiation and includes the values for **Negotiated Pkt Window** and **Negotiated Pkt Size** in the facilities field of the packet; (2) the called node performs simple boundary checking to verify that the negotiated parameters are within acceptable ranges and issues a *Call Confirm* packet.

10. Select and then Save .

NCU returns to the **X.25 PDN ADDRESS MAP** window which now displays the map entry you just added.

3.9.3.1.2 Editing Map Entries

You edit map entries from the **X.25 PDN ADDRESS MAP** window, as follows:

1. Select the map entry you wish to modify under **IP Address**.
2. Select to display the menu depicted in Figure 3-17.
3. Select to display the **ADD PDN ADDRESS MAP** window (see Figure 3-18), which displays the current settings for this entry.
4. Go to *Section 3.9.3.1.1, Adding Map Entries* for instructions on how to set these parameters.

3.9.3.1.3 Deleting Map Entries

You delete map entries from the **X.25 PDN ADDRESS MAP** window, as follows:

1. Select the map entry you wish to delete under **IP Address**.
2. Select to display the menu depicted in Figure 3-17.
3. Select .

NCU deletes the entry from the X.25 PDN address map.

3.9.4 Configuring Bitmap Values

You set bitmap values from the **X.25 BITMAP VALUES** window (see Figure 3-19). This window allows you to set Bit numbers 0 through 19 to **ON** or **OFF**. NCU then constructs a 32-bit status word from these bitmap values that specifies certain low-level attributes of the interface between the node and the X.25 service provider. Refer to Table 3-1 to determine how to set Bit number 0 through 19.

Once you have set the bitmap values, select and . NCU automatically enters the status word you constructed in 8-digit hexadecimal. Complete the configuration process.

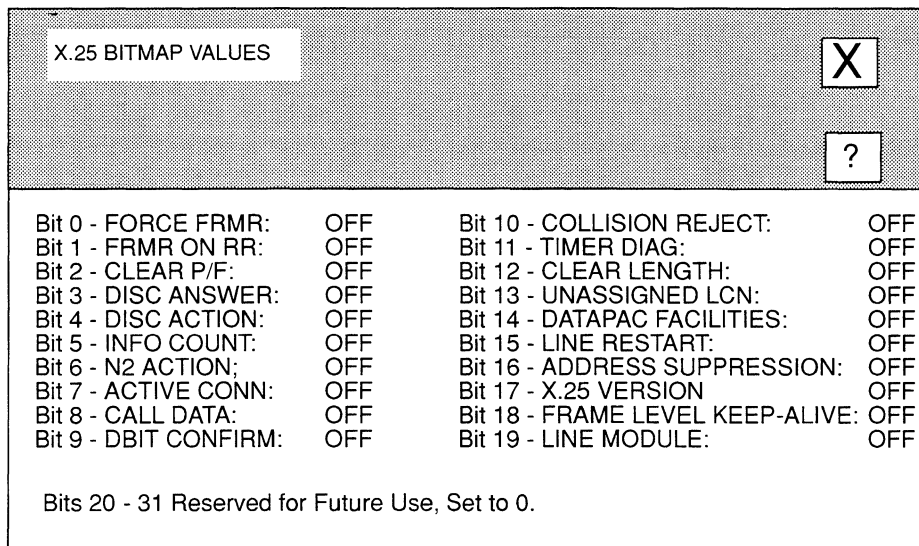


Figure 3-19. X.25 BITMAP VALUES Window

Table 3-1. X.25 Bitmap Values

Bit Number	Function	ON (logical 1)	OFF (logical 0)
0	FORCE FRMR	If X.25 sends an FRMR on the line, the reception of any other than an SABM, DISC, or FRMR causes another FRMR to be sent.	If X.25 sends an FRMR on the line, the reception of any frame other than an SABM, DISC, or FRMR is ignored.
1	FRMR ON RR	If X.25 sends an FRMR on the line, the reception of an RR frame causes another FRMR to be sent. All other frames (except SABM, DISC, and FRMR) are ignored).	If X.25 sends an FRMR on the line, the reception of any frame other than an SABM, DISC, or FRMR (to clear the condition is ignored).
2	CLEAR P/F	Receiving an unknown frame causes an FRMR frame to be sent with its PF bit set to zero (0), regardless of the PF setting in the received frame.	Receiving an unknown frame causes an FRMR frame to be sent with its P/F bit set to the same values as the P/F bit in the received frame

Table 3-1. (continued) X.25 Bitmap Values

Bit Number	Function	ON (logical 1)	OFF (logical 0)
3	DISC ANSWER	If X.25 sends an SABM (or is waiting for one), and receives a DISC, it responds with a UA.	If X.25 sends an SABM (or is waiting for one), and receives a DISC, it responds with a DM.
4	DISC ACTION	If X.25 sends an SABM (or is waiting for one), and receives a DISC, it sends an SABM immediately after responding to the previous flag.	If X.25 sends an SABM (or is waiting for one), and receives a DISC, it disconnects the link after responding to the previous flag.
5	INFO COUNT	If X.25 enters the T1 time-out state, sends an RR response, and then retransmits the unacknowledged INFO frame, the retry counter is not cleared until the retransmitted INFO frame is acknowledged. This procedure avoids an endless loop that would occur if the DCE were processing RR frames, but not INFO frames.	If X.25 enters the T1 time-out state, sends an RR, obtains an RR response, and then retransmits the unacknowledged INFO frame, the retry counter is cleared immediately per the CCIT definition. This procedure leaves open the possibility of an endless loop if the DCE were processing RR frames, but not INFO frames.
6	N2 ACTION	If X.25 is waiting for a UA, and receives either a T1 time-out or a DM, it retries the SABM up to N2(40) times. If (after N2 retries), it has still not received a UA, it goes to disconnect mode and ceases to transmit SABMs.	If X.25 is waiting for a UA, and receives either a T1 time-out or a DM, it retries the SABM up to N2 (40) times. If (after N2 retries), it has still not received a UA, it goes to disconnect mode and continues sending SABMs at intervals of T3 (20) seconds.
7	ACTIVE CONNECTION	X.25 begins sending SABMs as soon as the physical connection is established.	X.25 waits for an SABM from the remote end to initiate establishment of Frame Level.

Table 3-1. (continued) X.25 Bitmap Values

Bit Number	Function	ON (logical 1)	OFF (logical 0)
8	CALL DATA	X.25 will accept a CALL ACCEPT packet containing a User Data field, even if Fast Select was not specified in the call request.	X.25 will not accept a CALL ACCEPT packet containing a User Data field, unless Fast Select was specified in the call request.
9	D BIT CONFIRMATION	Disables the D bit in CALL CONFIRM packets.	Enables the D bit in CALL CONFIRM packets.
10	COLLISION REJECT	If a Clear Collision occurs, and the received CLEAR packet has a bad length, a new CLEAR packet is sent with a diagnostic code.	If a Clear Collision occurs, and the received CLEAR packet has a bad length, the CLEAR packet is dropped.
11	TIMER DIAG	If a T20 (3 minutes) time-out occurs, a RESTART packet is retransmitted with the original diagnostic code.	If a T20 time-out occurs, a RESTART packet is retransmitted with a "T20 Expired" diagnostic code.
12	CLEAR LENGTH	X.25 rejects CLEAR INDICATION and CLEAR CONFIRMATION packets if they contain facilities or user data.	X.25 accepts CLEAR INDICATION and CLEAR CONFIRMATION packets even if they contain facilities or user data.
13	UNASSIGNED LCN	X.25 clears calls received on an individual LCN.	X.25 ignores calls received on an invalid LCN.
14	DATAPAC FACILITIES	Enables special DATAPAC facilities checking.	Disables special DATAPAC facilities checking.
15	LINE RESTART	A RESTART packet is transmitted whenever Frame Level is established.	Frame Level establishment does not generate a RESTART packet.
16	ADDRESS SUPPRESSION	The local X.121 address is not included in any call packet sent from the device.	The local X.121 address is included in all locally originated call packets.
17	X.25 VERSION	The 1984 version of X.25 is supported.	The 1980 version of X.25 is supported.

Table 3-1. (continued) X.25 Bitmap Values

Bit Number	Function	ON (logical 1)	OFF (logical 0)
18	FRAME LEVEL KEEP_ALIVE	An RR with Poll Bit set is generated when the link has been idle for 2 seconds. In the absence of DCE response, normal retry and link recovery.	The link is not polled when it has been idle.
19	LINE_MODE	X.25 functions as DCE at Frame and Packet levels.	X.25 functions as DTE at Frame and Packet levels.
20 to 31	Reserved for Future Use	Not Applicable	Set to 0.

4 Editing Circuit Groups

When you generate the node *config* file, NCU automatically assigns each individual circuit to a circuit group. A circuit group is a collection of circuits (a collection may consist of only a single circuit) that the application modules use to bridge and route packets. All circuits in a circuit group are of the same type. In addition, all circuits in a circuit group originate at a common point and terminate at a common point.

Wellfleet nodes balance the traffic flow across all circuits in a circuit group in order to prevent one circuit from becoming overloaded (and possibly dropping packets) while other circuits providing unused bandwidth are available within the circuit group.

You can assign individual circuits (other than circuits that provide X.25 DDN or PDN service) to more than one circuit group. Figure 4-1 depicts an example in which individual circuits are assigned to multiple circuit groups. In the figure:

- ❑ Circuits **A**, **B**, **C**, and **D** belong to **Circuit Group 1**.
- ❑ Circuits **C** and **D** also belong to **Circuit Group 2**.
- ❑ Circuits **A** and **B** also belong to **Circuit Group 3**.

This chapter describes how to edit circuit groups. Specifically, it describes how to add, modify, and delete circuit groups. The first section describes how to access circuit group parameters.

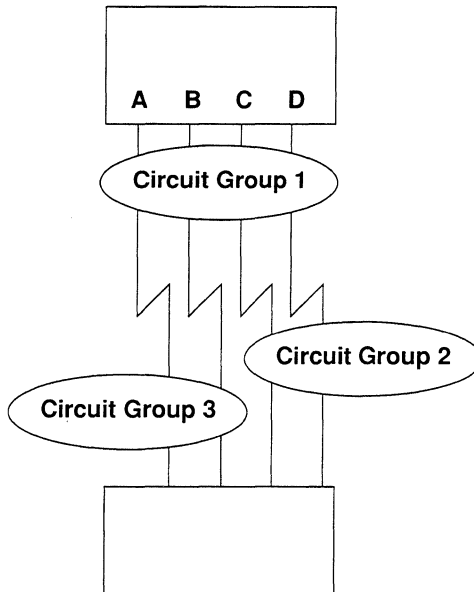


Figure 4-1. Multiple Circuit Group Assignment

4.1 Accessing Circuit Group Parameters

In order to access circuit group parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 4-2 displays the **EDIT NODE CONFIGURATION** window for a single node. In the figure, the network operator is changing the configuration-default parameters for **BOS**; any changes the network operator makes will affect **BOS** only.

To access the circuit group parameters, select . NCU displays the **CIRCUIT GROUPS** window which allows you to add, modify, and delete circuit groups (see Figure 4-3). Refer to the appropriate section of this chapter for the procedure you wish to perform.

EDIT NODE CONFIGURATION

X
?

Circuits!
Circuit Groups!
Protocols

Configuration Name: NEUSCurrent
 Comment:

Node: BOS

Global Parameters: Add **Modify** Delete

Session Type	Device ID			
User	Console	Add	Modify	Delete
None	Printer	Add	Modify	Delete
User	Modem 1	Add	Modify	Delete
None	Modem 2	Add	Modify	Delete
Telnet		Add	Modify	Delete
Disk Log		Add	Modify	Delete

← indicates you are editing the parameters of a single node

Figure 4-2. EDIT NODE CONFIGURATION Window for a Single Node

4.1.1 Adding Circuit Groups

You add circuit groups from the **CIRCUIT GROUPS** window (see Figure 4-3), as follows:

1. Select Add! in the **CIRCUIT GROUPS** window.
 NCU displays the **NAME CIRCUIT GROUP** window (see Figure 4-4).
2. Enter the name of the circuit group you wish to add at **Circuit Group Name**.
3. Select X and then Save.

NCU displays the **SELECT NETWORK** window (see Figure 4-5). This window displays the network segments in the current configuration.

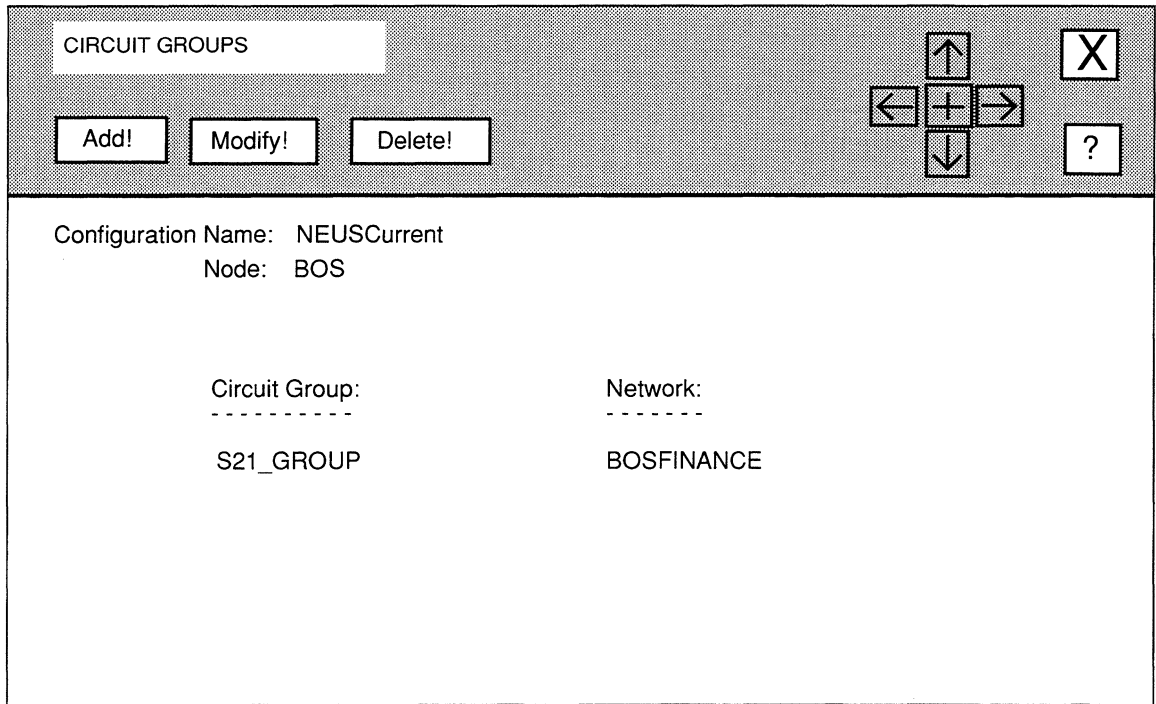


Figure 4-3. CIRCUIT GROUPS Window

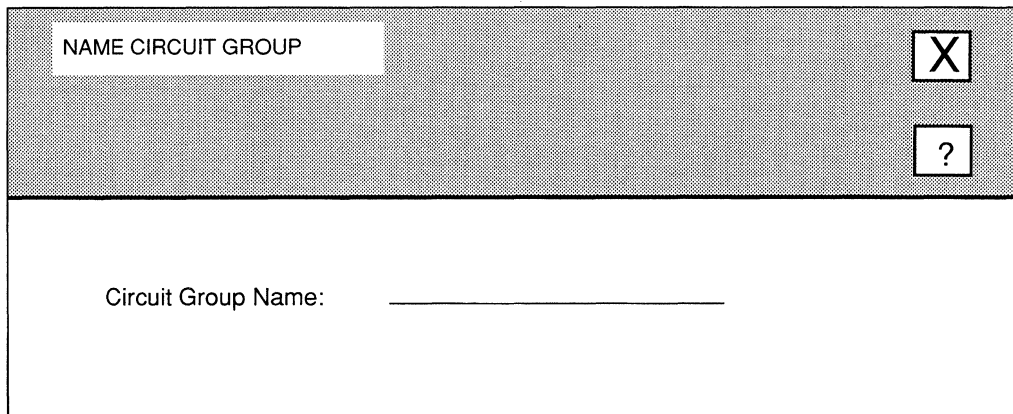


Figure 4-4. NAME CIRCUIT GROUP Window

SELECT NETWORK

Configuration Name: NEUSCurrent
Circuit Group: S21_GROUP

Network:	Network Type:	Circuit Type:
NYTOBOSTON	SYNC	POINT TO POINT
NYTOLONDON	DS1	POINT TO POINT
NYTOCHICAGO	SYNC	POINT TO POINT

Figure 4-5. SELECT NETWORK Window

4. Select the network segment to which you wish the new circuit group to connect.

Note

NCU does not automatically configure the circuit group (interface) for the protocols running on the connected network segment. Refer to the appropriate section of this guide for information on how to configure protocol-specific interface parameters.

5. Select and then .

NCU displays the **CIRCUIT GROUP** window for the circuit group you just added. Because you just added the circuit group, this window displays no circuits. You must now add circuits to the circuit group.

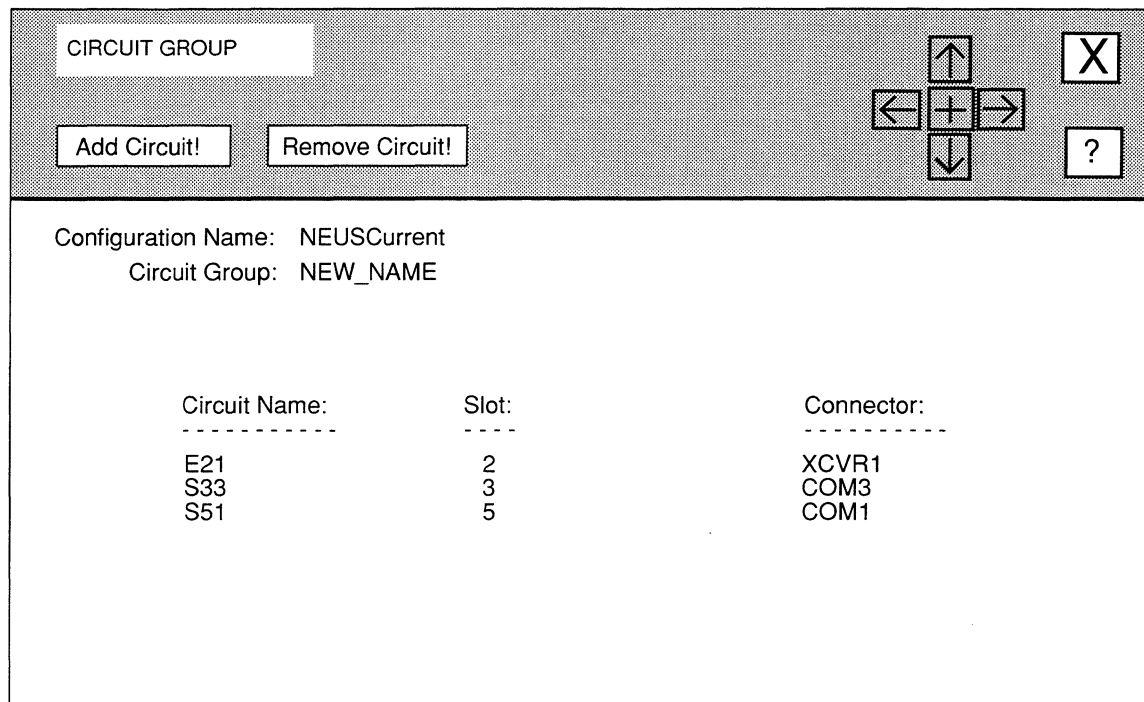


Figure 4-6. CIRCUIT GROUP Window

6. Select .
 NCU displays the **CIRCUITS** window (see Figure 4-7), which displays the available circuits on the current node.
7. Select the circuit you wish to add to the circuit group.
8. Select and then .
 NCU returns to the **CIRCUIT GROUP** window for the new circuit group, which now displays the circuit you just added.
9. Repeat steps 6 through 8 until you have added all desired circuits to the new circuit group.

CIRCUITS		
		<input type="button" value="↑"/> <input type="button" value="X"/>
		<input type="button" value="←"/> <input type="button" value="+"/> <input type="button" value="→"/> <input type="button" value="↓"/> <input type="button" value="?"/>
Circuit Name:	Slot:	Connector:
-----	-----	-----
E21	2	XCVR1
S21	2	COM1
T31	3	TOKEN

Figure 4-7. CIRCUITS Window

4.1.2 Modifying Circuit Groups

You modify circuits groups from the **CIRCUIT GROUPS** window (see Figure 4-3), as follows:

1. Select the circuit group you wish to modify in the **CIRCUIT GROUPS** window and then select .

NCU displays the **RENAME CIRCUIT GROUP** window (see Figure 4-8). If you wish to rename the circuit group, re-enter the name at **Circuit Group Name** and then select .

2. Select and then .

NCU displays the **CIRCUIT GROUP** window (see Figure 4-6) for the circuit group you wish to modify. If you changed the circuit group name, NCU displays the new name at **Circuit Group**.

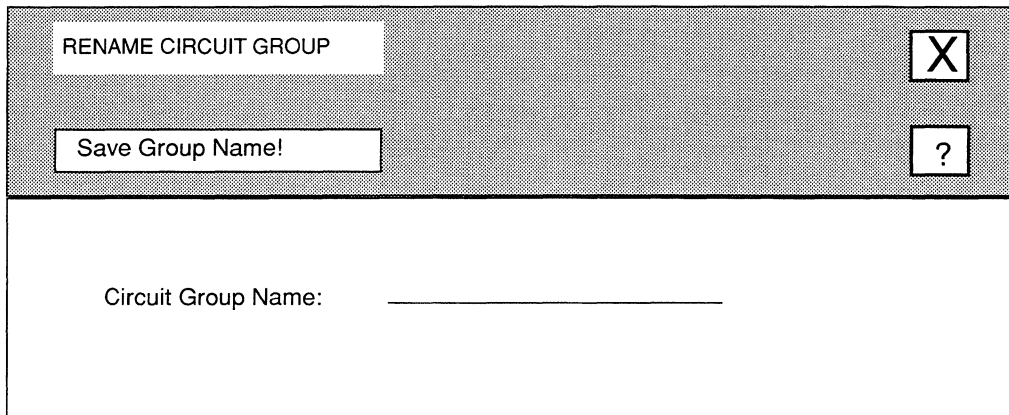


Figure 4-8. RENAME CIRCUIT GROUP Window

You may now modify the circuit group as follows:

- ❑ To add a circuit to the circuit group:
 - Select .
 - NCU displays the **CIRCUITS** window (see Figure 4-7), which displays the available circuits on the current node.
 - Select the circuit you wish to add.
 - Select and then .
 - NCU returns to the **CIRCUIT GROUP** window for the new circuit group, which now displays the circuit you just added.
- ❑ To remove a circuit from the circuit group:
 - Select the circuit you wish to remove.
 - Select .
 - NCU removes the circuit from the circuit group.

3. Select and then to exit the **CIRCUIT GROUP** Window.

4.1.3 Deleting Circuit Groups

You delete circuit groups from the **CIRCUIT GROUPS** window for a node (see Figure 4-3), as follows:

1. **Select the circuit group you wish to delete.**
2. Select .

NCU deletes the circuit group from the node.

5 Editing TCP/IP Parameters

TCP/IP parameters consist of the following:

- ❑ Basic parameters
Basic parameters apply to the entire TCP/IP router software module.
- ❑ Interface parameters
Interface parameters apply to individual TCP/IP interfaces.
- ❑ Routing parameters
Routing parameters consist of parameters for routing protocols (*Routing Information Protocol*, RIP, *Exterior Gateway Protocol*, EGP, and *Open Shortest Path First Protocol*, OSPF), as well as, static-route, default-route, and adjacent-host route parameters, and parameters for address filters, import-route filters, and export-route filters.
- ❑ Application parameters
Application parameters are *Trivial File Transfer Protocol* (TFTP), BOOTP, and *Simple Network Management Protocol* (SNMP) agent parameters.
- ❑ TCP parameters
Transmission Control Protocol (TCP) is the Internet-standard connection-mode, transport-protocol level.

This chapter describes how to access and edit the above parameters. The first section provides an overview of the TCP/IP protocol suite.

5.1 TCP/IP Overview

The TCP/IP (Transmission Control Protocol/Internet Protocol) protocol suite is the result of extensive experimentation (dating back to the late 1970s) performed by the Defense Advanced Research Projects Agency (DARPA). Such experimentation was given impetus by the exponential growth in the number of defense-related computers, many of which were manufactured by different vendors and which “spoke” different protocols. With no vendor implementations of international standards, and faced with a need to satisfy immediate operational requirements, the United States Department of Defense (DoD)

mandated that all DoD computer equipment conform to a series of military-standard protocols derived from DARPA-sponsored research. Table 5-1 lists the names and functions these protocols.

TCP/IP is now popular in the non-military market place. Numerous vendors have introduced TCP/IP-based products, and expenditures for such products have risen steadily. Many commercial users find TCP/IP a mature (although still evolving), robust, and reliable means of achieving multi-vendor interoperability. Some commercial users have adopted TCP/IP as an interim step while awaiting the widespread availability of OSI products.

Table 5-1. Military-Standard Protocols

Protocol	Function
<i>Internet Protocol (IP)</i>	Provides a connectionless, unreliable delivery service
<i>Transmission Control Protocol (TCP)</i>	Provides a reliable end-to-end delivery service
<i>File Transfer Protocol (FTP)</i>	Provides a simple mechanism for transferring text and binary files
<i>Simple Mail Transfer Protocol (SMTP)</i>	Provides an electronic mail facility
<i>Telnet Protocol</i>	Provides virtual scroll-mode terminal capability

Conceptually, TCP/IP consists of four layers (see Figure 5-1) with the following functions:

Layer	Function
Network interface	Manages the data exchange between a host computer and the network to which it is attached.
Internet	Provides a host-addressing function.
Transport	Provides a process-addressing function.
Application	Manages the functions (remote login, file transfer, etc.) that user programs require.

LANs that run TCP/IP can communicate with each other by means of IP routers. A router (which creates a network of networks, or an *internet*) is any device that chooses among available paths. Routing consists of sending an IP datagram from a source to a destination over one of several available paths. Unlike bridges, which must store routes to all *hosts* in an extended network, routers need only store routes to other *networks*, and to a small number of hosts (namely, those hosts on networks directly connected to the router).

IP routers perform three primary functions; the IP router:

- Acquires* knowledge of other routers and hosts on the network.
- Builds* the network's topology based on this information.
- Selects* the best route for each datagram.

In performing its basic functions, the IP router uses portions of the TCP/IP protocol suite. Table 5-2 lists the Internet Request for Comments (RFCs) that describe specific protocols and functionality that the IP router provides. This chapter assumes you are familiar with these RFCs.

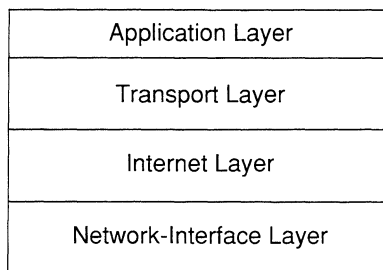


Figure 5-1. The Four Layers in the TCP/IP Model

5.2 Accessing IP Parameters

In order to access TCP/IP parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 5-2 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default parameters in NCU; any changes the network operator makes will affect every node configured thence on.

To access the IP parameters, select and then . NCU displays the **NODE IP CONFIGURATION** window which provides access to the IP Parameters (see Figure 5-2).

Table 5-2. Internet Request for Comments (RFCs) for the IP Router

RFC 768	<i>User Datagram Protocol (UDP)</i>
RFC 783	<i>Trivial File Transfer Protocol (TFTP)</i>
RFC 791	<i>Internet Protocol (IP)</i>
RFC 792	<i>Internet Control Message Protocol (ICMP)</i>
RFC 793	<i>Transmission Control Protocol (TCP)</i>
RFC 826	<i>Address Resolution Protocol (ARP)</i>
RFC 877	Transmission of IP datagrams over X.25 networks
RFC 904	<i>Exterior Gateway Protocol (EGP)</i>
RFC 950	Internet sub-netting procedures
RFC 951	<i>Bootstrap Protocol (BOOTP)</i>
RFC 1009	Requirements for Internet gateways
RFC 1042	Transmission of IP datagrams over IEEE 802 networks
RFC 1058	<i>Routing Information Protocol (RIP)</i>
RFC 1063	<i>IP Maximum Transmission Unit (MTU) discovery option</i>
RFC 1084	BOOTP Vendor Information Extensions
RFC 1131	<i>Open Shortest Path First Protocol (OSPF)</i>
RFC 1155	Structure and identification of management information
RFC 1156	<i>Internet Management Information Base (MIB)</i>
RFC 1157	<i>Simple Network Management Protocol (SNMP)</i>

5.3 Editing IP Basic Parameters

The **NODE IP CONFIGURATION** window displays IP basic parameters. This window allows you to add and delete IP basic parameters.

5.3.1 Adding IP Basic Parameters

You add IP basic parameters, as follows:

1. **At Auto Enable, select the state of the IP router software when the node boots.**

EDIT NODE CONFIGURATION

Circuits!
Circuit Groups!
Protocols

Configuration Name: DEFAULTS
 Comment: Default configuration values. ←

Node: DEFAULT_NODE ← *indicates you are editing the NCU defaults*

Global Parameters: Add **Modify** Delete

Session Type	Device ID			
User	Console	Add	Modify	Delete
None	Printer	Add	Modify	Delete
User	Modem 1	Add	Modify	Delete
None	Modem 2	Add	Modify	Delete
Telnet		Add	Modify	Delete
Disk Log		Add	Modify	Delete

Figure 5-2. EDIT NODE CONFIGURATION Window for Default Settings

This IP-router-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable the IP router application software module when the node boots, as follows:

- ❑ When global **Auto Enable** is **set to NO**, the IP router (and every other application software module) is unconditionally disabled.
 You will subsequently need to enable the IP router manually with the NCL Interpreter after the node boots.
- ❑ When global **Auto Enable** is set to **YES**, the IP router (and every other application software module) is conditionally enabled.
 If global **Auto Enable** is set to **YES**, do one of the following:
 - Select **YES** to enable the IP router.
 - Select **NO** to disable the IP router (you will subsequently need to enable the IP router manually with the NCL Interpreter after the node boots).

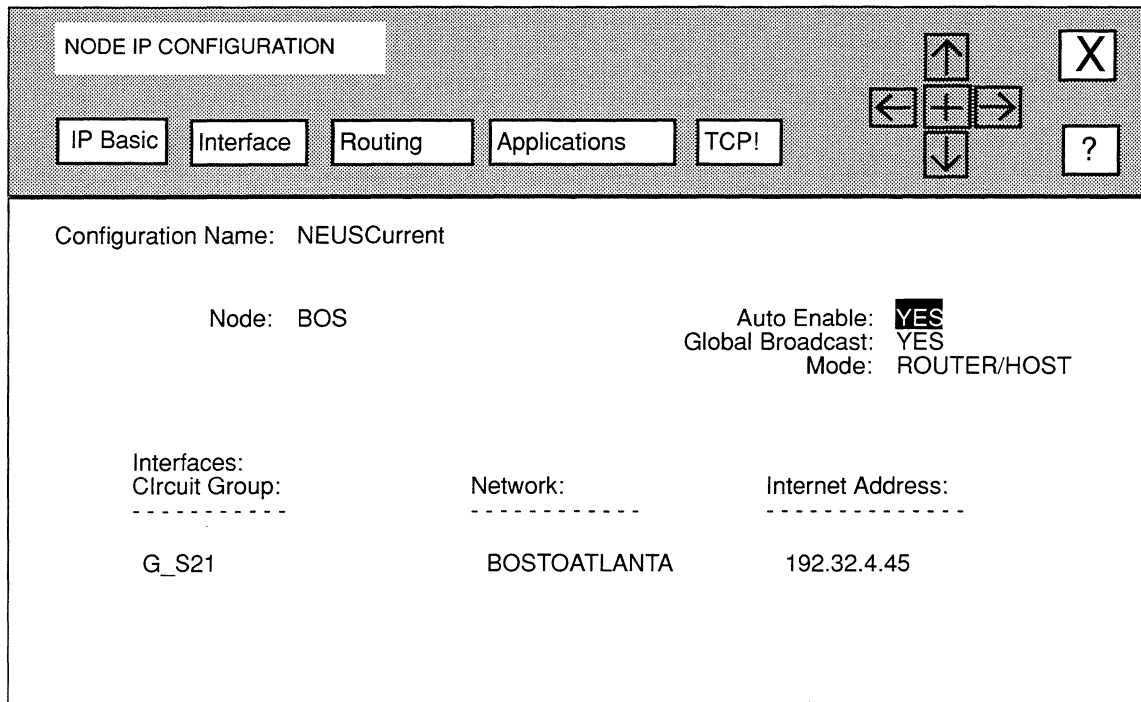


Figure 5-3. NODE IP CONFIGURATION Window

2. At **Global Broadcast**, select how the router responds when it receives a **global broadcast message (a message containing an all-1s IP destination address)**.

YES Specifies that the router accepts global broadcast messages.

NO Specifies that the router discards global broadcast messages.

Note

Because the RIP protocol uses global broadcast messages to propagate periodic routing updates, setting **Global Broadcast** to **NO**, effectively disables RIP.

3. At Mode, select the node's operational characteristics.

ROUTER/HOST..... Specifies that the IP router processes all packets explicitly addressed to it, and routes all other packets. If the node does not bridge IP datagrams, select **ROUTER/HOST**.

HOST ONLY Specifies that the node, acting as a bridge (or in "end-node" mode), receives IP datagrams addressed to it, while continuing to bridge all other IP and non-IP traffic. If the node bridges IP datagrams, and you wish to provide management access (through Telnet, TFTP, or SNMP) to the node, select **HOST ONLY**.

Because no IP routing can take place in end-node mode, you must configure the Bridge for each interface that conveys IP datagrams. The Bridge then forwards IP datagrams that are not addressed to the Wellfleet node.

In addition, you must configure a network interface for each interface over which management access is desired. You must specify an identical IP address and mask combination for each interface.

In end-node mode, the IP router functions as if the Wellfleet node were a virtual host on one of the bridged interfaces. For example, Figure 5-4 depicts a Wellfleet node (functioning in end-node mode, and with an IP address of 192.32.1.1) bridging traffic for three connected networks. NCU assigns the virtual host to the first circuit of the first circuit group to which you assigned an IP address.

Figure 5-5 depicts the virtual-host configuration, assuming that all circuit groups contain a single member and that the interface to **NETWORK A** was configured initially.

Traffic sent from the Wellfleet node to a host on **NETWORK A** appears only on **NETWORK A**; traffic sent from the Wellfleet node to a host on **NETWORK B**, however, appears on both **NETWORK B** and **NETWORK A**. Furthermore, if the interface to **NETWORK A** should become disabled for any reason, the IP router becomes inaccessible to hosts on any connected network.

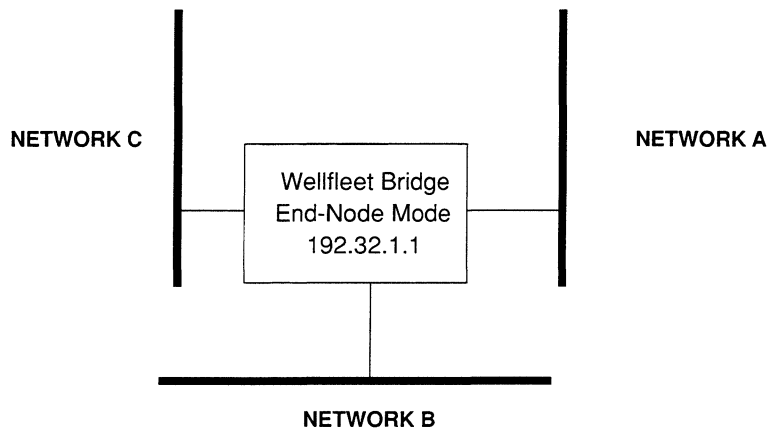


Figure 5-4. END-NODE Operation

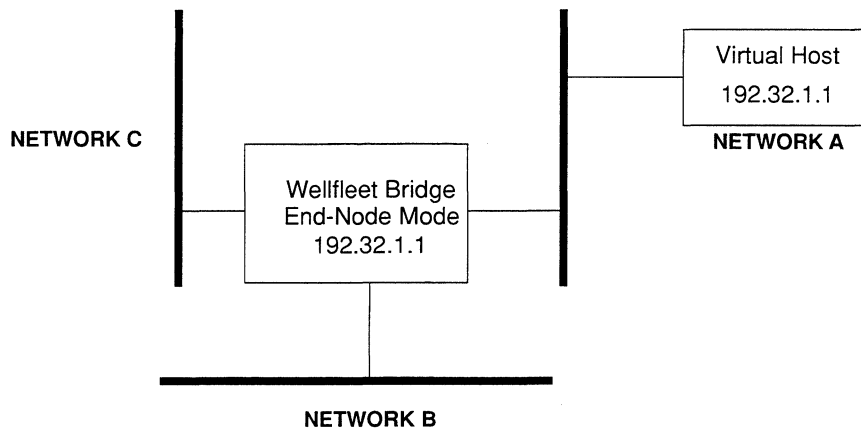


Figure 5-5. Virtual Host Configuration

4. To save the new IP basic parameters, select and then .
NCU displays the following window; press **[RETURN]** to clear it from the screen.

5.3.2 Deleting IP Basic Parameters

To delete IP basic parameters, in the **NODE IP CONFIGURATION** window, select

and then ,

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

IP Parameters deleted.

5.4 Editing IP Interfaces

You use the **NODE IP CONFIGURATION** window to edit IP interfaces. This window allows you to change and delete IP interface-specific parameters.

5.4.1 Changing Interface-Specific Parameters

The **IP INTERFACE DEFINITION** window (see Figure 5-6) displays interface-specific parameters. To access this window, select the interface you wish to reconfigure under **Interfaces** in the **NODE IP CONFIGURATION** window. Next, select and . NCU displays the **IP INTERFACE DEFINITION** window for that interface.

You may then reconfigure the interface, as follows:

1. **At Internet Address, enter the IP address in dotted decimal notation of the network interface.**

Every IP interface has a unique 32-bit address that contains two fields, as follows:

NET_ID LOCAL_ID

where:

- | | |
|-----------------|--|
| NET_ID | Is the network identification field which identifies a specific IP network. |
| LOCAL_ID | Is the host identification field which identifies a specific host on the IP network. |

The Internet Network Information Center (NIC) assigns the **NET_ID** portion of the IP address. Your network administrator assigns the **LOCAL_ID** portion. Depending on the network class (or size), **NET_ID** contains a specific number of bits, as follows:

If Network Contains:	Network Is:	NET_ID Contains:
More than 65,534 hosts	Class A	8 bits
Between 254 and 65,533 hosts	Class B	16 bits
Less than 254 hosts	Class C	24 bits

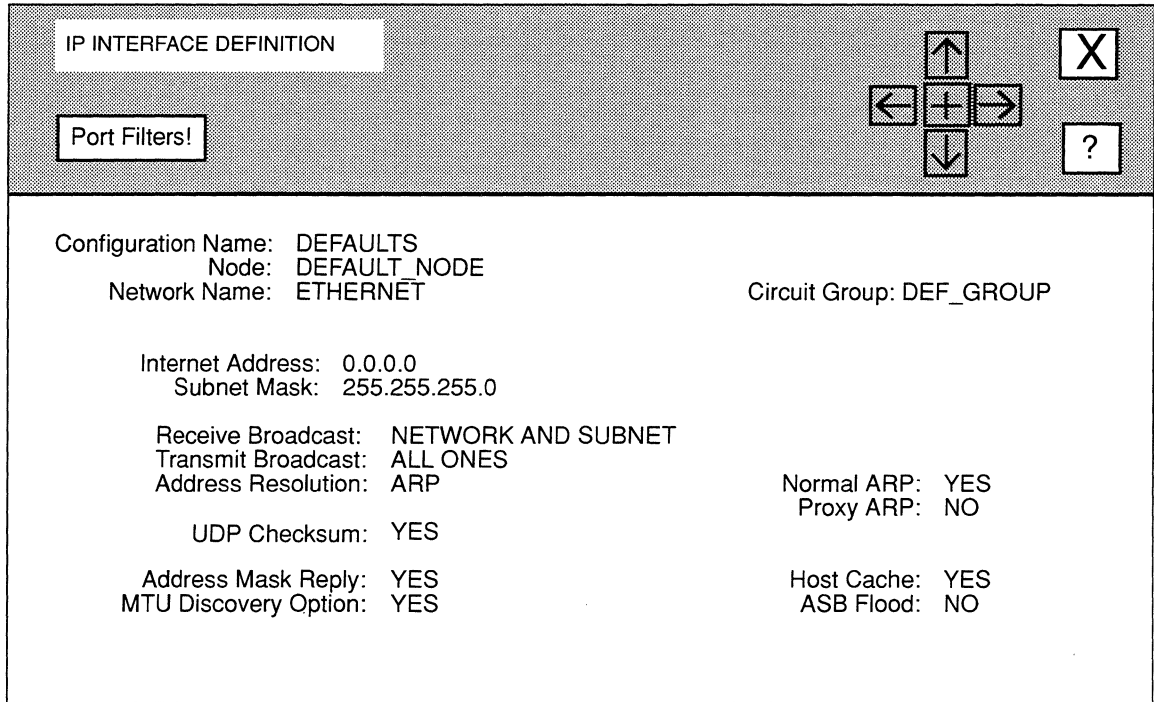


Figure 5-6. IP INTERFACE DEFINITION Window

The most-significant two bits of **NET_ID** indicate the network class. Depending on the network class, **LOCAL_ID** contains a specific number of bits, as follows:

Most-Significant 2 Bits:	Network Is:	LOCAL_ID Contains:
0X	Class A	24 bits
10	Class B	16 bits
11	Class C	8 bits

The most-significant two bits of **NET_ID** indicate the network class. Depending on the network class, **LOCAL_ID** contains a specific number of bits, as follows:

Most-Significant 2 Bits:	Network Is:	LOCAL_ID Contains:
0X	Class A	24 bits
10	Class B	16 bits
11	Class C	8 bits

For purposes of clarity, you specify IP addresses in dotted decimal notation. To specify an IP address in dotted decimal notation, convert each 8-bit octet of the IP address to a group of decimal digits, and separate the groups by decimal points.

For example, the 32-bit IP address below:

10000000 00100000 00001010 10100111

is specified as **128.32.10.167** in dotted decimal notation. The most-significant two bits (the first two bits, **10**) in the first octet indicate that the network is Class B; therefore, the first 16 bits compose the NIC-assigned **NET_ID** field. The third octet (**00001010**) and fourth octet (**10100111**) compose the **LOCAL_ID** or host-identification field.

2. At **Subnet Mask**, enter the dotted decimal subnet mask.

Subnets are two or more physical networks that share a common network-identification (**NET_ID**) field — the NIC-assigned portion of the 32-bit IP address. In networks that contain subnets, a portion of the host-identification (**LOCAL_ID**) field identifies the subnet designator. In such cases, **LOCAL_ID** contains the following:

SUBNET_ID HOST_ID

where:

SUBNET_ID Is the subnet designator.

HOST_ID Is the host designator.

SUBNET_ID and **HOST_ID** contain an arbitrary number of bits. Your network administrator allocates bits within **LOCAL_ID** to **SUBNET_ID** and **HOST_ID**, and then assigns values to **SUBNET_ID** and **HOST_ID**.

For example, below is the IP address of a network that contains subnets:

10000000 00100000 00001010 10100111

(The address is specified in dotted decimal notation as **128.32.10.167**.) The two most-significant bits in the first octet are **1** and **0** indicating that the network is a Class B network; therefore the NIC-assigned **NET_ID** field contains 16 bits, and the locally-assigned **LOCAL_ID** contains 16 bits.

The network administrator allocates the 16 bits in the **LOCAL_ID** field, as follows:

- Allocates the upper-eight bits (**00001010**) with a value of 10 to **SUBNET_ID**.
- Allocates the lower-eight bits (**10100111**) with a value of 167 to **HOST_ID**.

In other words, the 16-bit **LOCAL_ID** field, together with the 16-bit **NET_ID** field, specify host **167** on Subnet **10** of network **128.32**.

You need a subnet mask to identify those bits in the 32-bit IP address that specify **NET_ID** and those bits that specify **SUBNET_ID**. Like the IP address, you specify the subnet mask in dotted decimal notation.

You construct a subnet mask, as follows:

- Assign a value of 1 to each of the 8, 16, or 24 bits in **NET_ID**
- Assign a value of 1 to each bit in **SUBNET_ID**.
- Assign a value of 0 to each bit in **HOST_ID**.
- Convert the resulting 32-bit string to dotted decimal notation.

For example, the network administrator constructed a subnet mask for the IP address described earlier (**10000000 00100000 00001010 10100111**) as follows:

- Assign a value of **1** to each bit in **NET_ID** (the most-significant two bits of the IP address indicate that the network is Class B; therefore **NET_ID** contains 16 bits), as follows:

11111111 11111111

- Assign a value of **1** to each bit in **SUBNET_ID** (the network administrator allocated the upper-eight bits of **LOCAL_ID** to **SUBNET_ID**), as follows:

11111111

- Assign a value of **0** to each bit in **HOST_ID** (the network administrator allocated the lower-eight bits of **LOCAL_ID** to **HOST_ID**), as follows:

00000000

- Convert the resulting 32-bit string (**11111111 11111111 11111111 00000000**) to dotted decimal notation, as follows:

255.255.255.0000

3. At Receive Broadcast, select the types of broadcast messages that the IP router receives.

NETWORK AND SUBNET.... Specifies that the router accepts both network and sub-network broadcast messages. Selecting this option is preferable in most applications.

NETWORK ONLY Specifies that the router accepts only the network broadcasts. Select this option only if the router operates in a non-subnetted environment.

A network broadcast message takes one of the following forms:

- <Net_ID> <0's>
- <Net_ID> <1's>

where:

- <Net_ID> Is the NIC-assigned 8-bit, 16-bit, or 32-bit network address.
- <0's> Is a string of 8, 16, or 32 logical ones or zeroes.
- <1's> Is a string of 8, 16, or 32 logical ones or zeroes.

4. **At Transmit Broadcast, select the interface-specific (network and/or subnetwork) transmit broadcast address.**

ALL ONES Configures the interface to use the subnet mask (if one is specified at **SubnetMask**; otherwise, the interface uses the default mask) and places all ones in the host portion of the destination address

ALL ZEROS Configures the interface to use the subnet mask (if one is specified at **SubnetMask**; otherwise, the interface uses the default mask) and places all zeros in the host portion of the destination address

EXPLICIT BROADCAST Allows you to assign an explicit transmit-broadcast address (an address pattern other than all-1s or all-0s) for the interface. If you are configuring an interface that serves multiple networks, you must assign an explicit broadcast address. When you select **EXPLICIT BROADCAST**, the **IP INTERFACE** window displays **Broadcast Address**; specify the explicit address in dotted decimal notation.

5. **At Address Resolution, select how the node maps 32-bit IP addresses to 48-bit Ethernet addresses.**

ARP Configures the node to enable conditionally IP-to-Ethernet address mapping using the Address Resolution Protocol (ARP, as described in RFC 826) and the Proxy ARP protocol. When you select **ARP**, the **IP INTERFACE DEFINITION** window displays two parameters:

— **Normal ARP**

Select **YES** to enable ARP or **NO** to disable ARP

— **Proxy ARP**

Select **YES** to enable Proxy ARP or **NO** to disable Proxy ARP. Proxy ARP allows the IP router

to respond on a local interface to ARPs for a remote network; thus, the router assumes responsibility for IP datagrams destined for that network.

HP PROBE Enables the proprietary Hewlett Packard Probe protocol. Probe is an address-resolution mechanism that functions much like ARP. The node supports the following Probe messages:

- *Unsolicited Reply* (incoming and outgoing)
- *Name Request* (incoming)
- *Name Reply* (outgoing)
- *Virtual Address Request* (incoming and outgoing)
- *Virtual Address Reply* (incoming and outgoing)
- *Gateway Request* (incoming)
- *Gateway Reply* (outgoing)

ARP & HP PROBE Enables both HP PROBE and ARP. When you select **ARP & HP PROBE**, you configure the node to use the first-in resolved media address until the address is modified by subsequent updates. Also, when you select **ARP & HP PROBE**, the **IP INTERFACE DEFINITION** window displays two parameters:

- **Normal ARP**
Select **YES** to enable ARP or **NO** to disable ARP
- **Proxy ARP**
Select **YES** to enable Proxy ARP or **NO** to disable Proxy ARP. Proxy ARP allows the IP router to respond on a local interface to ARPs for a remote network; thus, the router assumes responsibility for IP datagrams destined for that network.

DDN Enables the DDN address-resolution algorithm. Select **DDN**, if the network interface provides X.25 DDN service.

PDN Enables a table-based RFC 877-compliant address-resolution mechanism. Select **PDN**, if the network interface provides X.25 PDN service.

NONE..... Disables address mapping. When you select **NONE**, you must configure all MAC address-to-IP address relationships statically.

6. At UDP Checksum, enable or disable UDP checksum processing for the network interface.

YES Enables checksum processing. In virtually all instances, you should enable checksum processing.

NO Disables checksum processing and provides backward compatibility with UNIX BSD 4.1.

7. At Address Mask Reply, select whether the node generates ICMP *address-mask-reply* messages at boot time and in response to valid *address mask request* messages.

YES Enables ICMP *address-mask-reply* message generation in compliance with the relevant sections of RFCs 950 and 1009.

NO Disables ICMP *address-mask-reply* message generation.

Note

Set **Address Mask Reply** to **NO** if the network interface provides X.25 DDN or PDN service. Neither X.25 DDN service nor X.25 PDN service support the *address-mask-reply* facility.

8. At MTU Discovery Option, enable or disable *Probe MTU* and *Reply MTU* options.

YES Enables *Probe MTU* and *Reply MTU* options (IP options numbers 11 and 12 in RFC 1063). These options enable the router to learn the minimum MTU of all networks traversed by an IP datagram from source to destination. The MTU option can significantly decrease network load by eliminating the need for transit fragmentation and destination reassembly.

NO Disables *Probe MTU* and *Reply MTU* options (IP options numbers 11 and 12 in RFC 1063).

Note

Set **MTU Discovery Option** to **NO** if the network interface provides X.25 DDN or PDN service. Neither X.25 DDN service nor X.25 PDN service support the MTU discovery facility.

9. At Host Cache, enable or disable the aging of physical-level addresses learned by any of the address resolution protocols.

YES Enables the aging features so that cache entries that have not been accessed within two minutes are aged out (removed from the cache). Once an entry has been aged out, the IP router must re-acquire the physical-level address (via an address-resolution protocol) should it be needed in the future.

NO Disables the aging feature so that entries in the address-resolution cache are not aged out.

10. At ASB Flood, enable or disable the all-subnet broadcast facility.

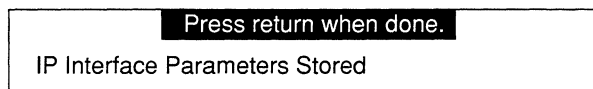
YES Enables the all-subnet broadcast facility which allows certain IP broadcast datagrams (specifically all-subnet broadcast datagrams) received on one interface to be flooded across other router interfaces.

An all-subnet broadcast datagram is a datagram whose destination address is equal to the broadcast address for an entire subnet. If, for example, a network interface serves the subnet 128.10.2.1 (with a subnet mask of 255.255.255.0), any datagram with a destination address of 128.10.255.255 is considered an all-subnet broadcast.

When you set **ASB Flood** to **YES**, the IP router floods all-subnet broadcasts received on this interface to other interfaces which service the same subnet. Similarly, all-subnet broadcasts received on other interfaces are flooded to this interface.

NO Disables the all-subnet broadcast facility.

If you wish to configure port filters for the interface, go to the next section. Otherwise, select and then . NCU displays the following window; press **[RETURN]** to clear it from the screen:



Note

To exit without saving the IP interface parameters, select and then .

5.4.1.1 Editing Port Filters

Port filters manage the flow of Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) datagrams across an interface. TCP and UDP are internet transport-level protocols: TCP provides a reliable, connection-mode, full-duplex data stream across an internet (Figure 5-7 depicts the format of a TCP segment); UDP provides connectionless datagram service (Figure 5-8 depicts the structure of a UDP datagram).

UDP datagrams and TCP segments are originated by and addressed to ports. Ports are logical abstractions that transport-level protocols use to distinguish between multiple sources and destinations at a single node.

In order to facilitate application-to-application data flow, the Internet authorities have assigned *well-known port numbers* to certain commonly-used application programs. Examples of well-known port numbers include port numbers assigned to remote-login (Telnet) programs, to file-transfer programs, and to remote-job-entry (RJE) programs. Table 5-3 lists well-known port numbers that UDP and TCP use.

TCP/UDP filters enable you to control the flow of TCP segments and UDP datagrams on an interface-by-interface basis. For example, you can construct two TCP/UDP port filters for the “Blue” Net interface: the first filter forwards (routes) traffic to well-known port number 25; the second filter drops all other application traffic (specified by port number 0). Together these two filters provide “Blue” Net users access to internet mail service while they deny users access to all other TCP and UDP functionality.

NCU allows you to add, delete, and update interface-specific port filters.

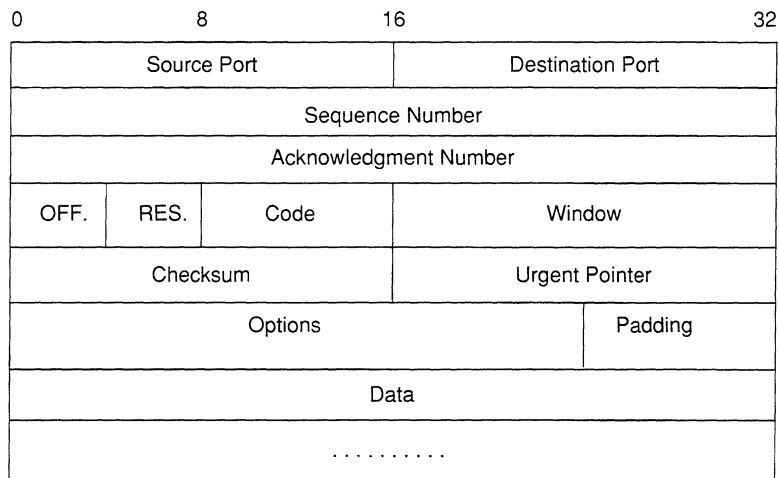


Figure 5-7. TCP Segment Format

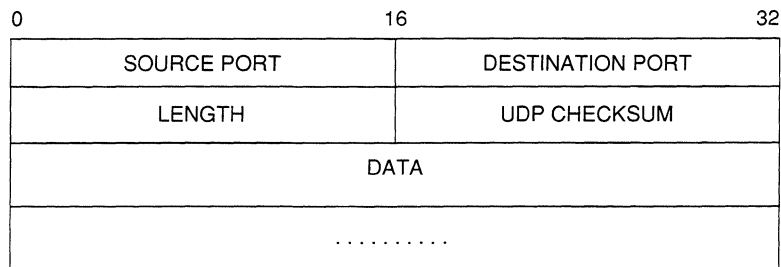


Figure 5-8. UDP Datagram Format

5.4.1.1.1 Adding Port Filters

You add TCP/UDP port filters to an interface, as follows:

1. **In the IP INTERFACE DEFINITION window for the interface, select .**
 NCU displays the **IP PORT FILTERS** window (see Figure 5-9).
2. **Select and then to display the ADD PORT FILTER window (see Figure 5-10).**
3. **At Port Number, enter the well-known port number associated with the application traffic you want to filter.**

Refer to Table 5-3 to obtain the well-known port number for the type of TCP or UDP traffic you want to filter.

Note

If you specify a value of **0**, the interface filters all port numbers (both well-known and dynamically-assigned).

4. **At Action, select how the interface disposes of the filtered TCP segment or filtered UDP datagram.**
 - FORWARD** Specifies that the interface sends the filtered TCP segment or UDP datagram to its destination.
 - DROP**..... Specifies that the interface drops the TCP segment or UDP datagram.
 - CONTINUE** Is not used to filter TCP segments or UDP datagrams. Do not select **CONTINUE**.

Table 5-3. TCP/UDP Well-Known Ports

Port Number	Application	Used By
0	Reserved	--
1	Unassigned	--
2	Unassigned	--
3	Unassigned	--
4	Unassigned	--
5	RJE	UDP and TCP
7	ECHO	UDP and TCP
9	DISCARD	UDP and TCP
11	USERS	UDP and TCP
13	DAYTIME	UDP and TCP
15	NETSTAT	UDP and TCP
17	QUOTE	UDP and TCP
19	CHARGEN	UDP and TCP
20	FTP-DATA	TCP
21	FTP	TCP
23	TELNET	TCP
25	SMTP	TCP
37	TIME	UDP and TCP
39	RLP	UDP and TCP
42	NAMESERVER	UDP and TCP
43	NICNAME	UDP and TCP
53	DOMAIN	UDP and TCP
67	BOOTPS	UDP and TCP
68	BOOTPC	UDP and TCP
69	TFTP	UDP and TCP
75	Any private dial-out service	UDP and TCP
77	Any private RJE service	UDP and TCP
79	FINGER	UDP and TCP
95	SUPDUP	TCP
101	HOSTNAME	TCP
102	ISO-TSAP	TCP
113	AUTH	TCP
117	UUCP-PATH	TCP
123	NTP	UDP and TCP
133-159	Unassigned	--
160-223	Reserved	--
224-241	Unassigned	--
247-255	Unassigned	--
247-255	Unassigned	-

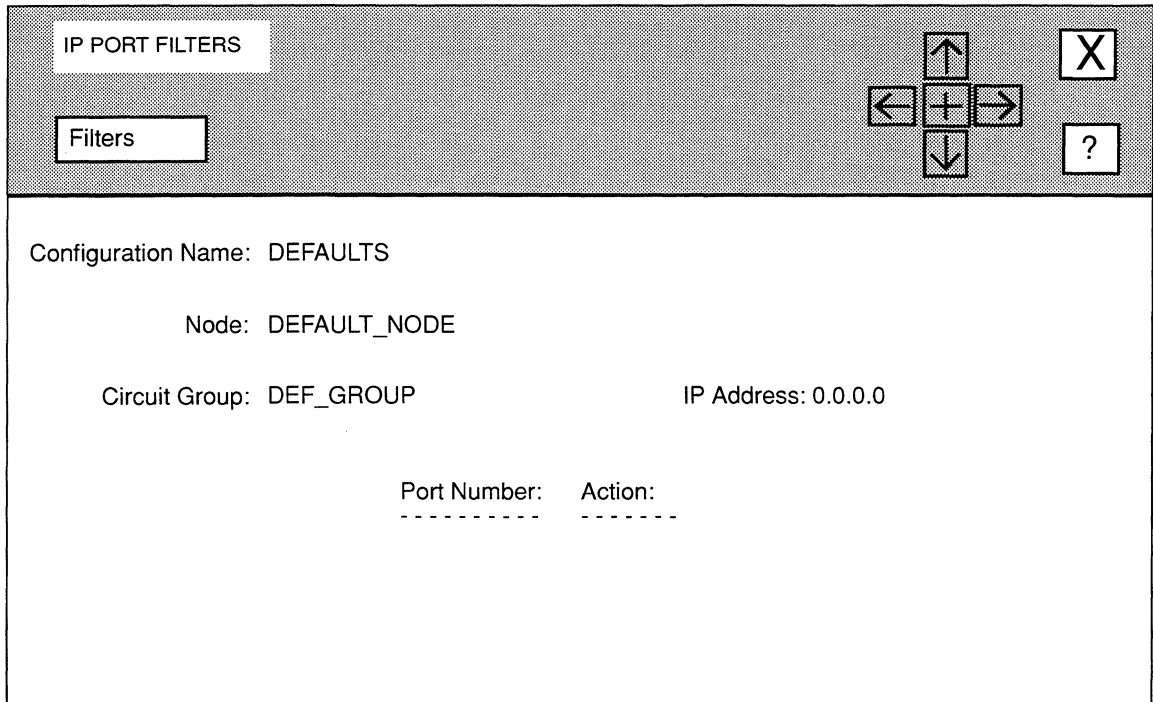


Figure 5-9. IP PORT FILTERS Window

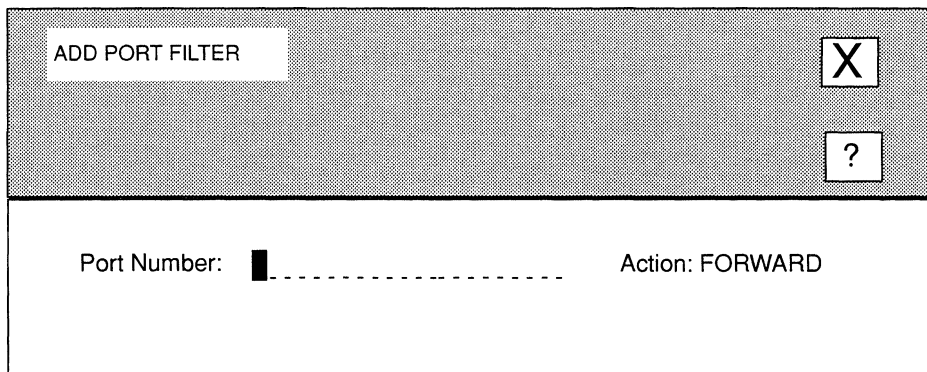


Figure 5-10. ADD PORT FILTER Window

5. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.
Port Filter stored.

NCU returns to the **IP PORT FILTERS** window which now lists the port filter you configured. Select and then Confirm to return to the **IP INTERFACE DEFINITION** window.

To save the displayed IP-interface parameters, select and then Save . NCU displays the following window; press **[RETURN]** to clear it from the screen:

Press return when done.
IP Interface Parameters Stored

Note

To exit without saving the IP interface parameters, select and then Quit .

5.4.1.1.2 Deleting Port Filters

You delete port filters from the **IP PORT FILTERS** window (see Figure 5-11). Select the port filter you wish to delete (in Figure 5-11, the network operator selected **Port Number 0**). Then, select Filters and Delete . NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.
Port Filter deleted

Select and then Confirm to return to the **IP INTERFACE DEFINITION** window:

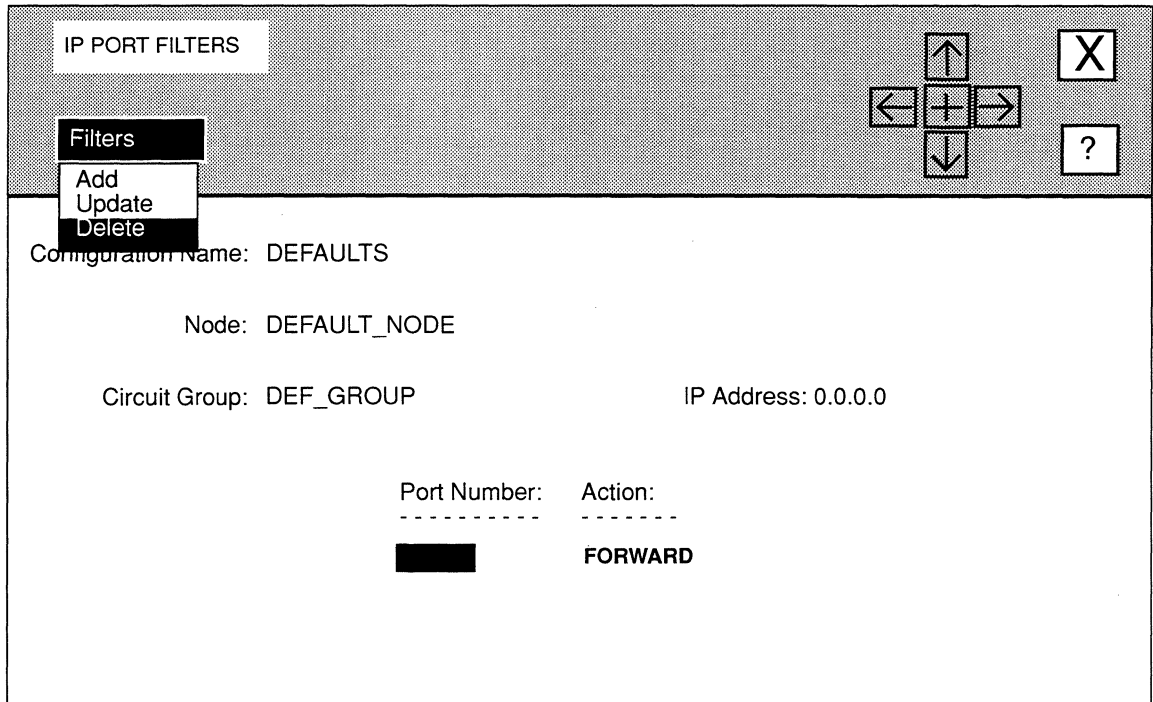
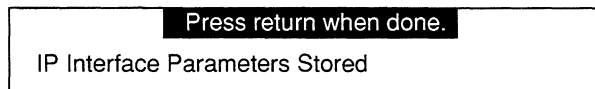


Figure 5-11. IP PORT FILTERS Window with Port Selected

To save the displayed IP-interface parameters, select and then . NCU displays the following window; press **[RETURN]** to clear it from the screen:



Note

To exit without saving the IP interface parameters, select and then .

NODE IP CONFIGURATION		
IP Basic	Interface	Routing Applications TCP!
	Modify	
	Delete	
Configuration Name: NEUSCurrent		
Node: BOS	Auto Enable: YES	Global Broadcast: YES
	Mode: ROUTER/HOST	
Interfaces:	Network:	Internet Address:
Circuit Group:		
-----	-----	-----
G_S21	BOSTTOATLANTA	192.32.4.45

Figure 5-12. NODE IP CONFIGURATION Window

5.4.1.1.3 Updating Port Filters

You update port filters from the **IP PORT FILTERS** window (see Figure 5-11). Select the port filter you wish to update. Then, select and . NCU displays the **ADD PORT FILTER** window which displays the current configuration information for the port filter you selected. Reconfigure the filter as you wish (see *Section 5.4.1.1.1, Adding Port Filters* for instructions).

5.4.2 Deleting IP Interface-Specific Parameters

You delete IP interfaces from the **NODE IP CONFIGURATION** window (see Figure 5-12). First select the interface you wish to delete (in Figure 5-12, the network operator selected the interface **G_S21**). Then, select and . NCU displays the following window; select **yes** to delete the IP interface and to clear the above window from the screen.

Are you sure you want to delete G_S21 ? yes no
--

5.5 Editing Routing Parameters

Routing parameters consist of:

- ❑ Routing-protocol parameters

The Wellfleet node supports the following routing protocols:

- *Routing Information Protocol, RIP*
- *Exterior Gateway Protocol, EGP*
- *Open Shortest Path First Protocol, OSPF*

- ❑ Route parameters

You can configure the following types of routes:

- Static routes
- Default routes
- Adjacent-Host routes

- ❑ Filter parameters

You can configure the following types of filters:

- Address filters
- Import-route filters
- Export-route filters

The following sections describe how to edit the above parameters.

5.5.1 Editing RIP Parameters

You configure RIP from the **RIP** window (see Figure 5-13). To display the **RIP** window, select and then in the **NODE IP CONFIGURATION** window. The **RIP** window allows you to add, update, and delete RIP basic parameters, as well as, to add, update, and delete RIP interface-specific parameters.

5.5.1.1 Adding RIP Basic Parameters

The **RIP Network Diameter** parameter is the only parameter you need to specify in order to add RIP basic parameters. You specify this parameter in the **RIP** window. At **RIP Network Diameter**, specify the value, or hop count, that RIP uses to denote infinity.

RIP

RIP Basic RIP Interface

Configuration Name: NEUSCurrent

Node: BOS RIP Network Diameter: 15

Interfaces: Circuit Group:	Network:	Internet Address:	RIP Cost:
G_S21	NYR&D2	192.32.4.45	

Figure 5-13. RIP Window

Note

Because internet convention uses a value of 15 to denote infinity (at which point RIP declares a network unreachable), it is strongly recommended that you accept the default value 15 when you set **RIP Network Diameter**. For RIP to operate properly, *every* router within the network must use the same network diameter value. Hosts also use **RIP Network Diameter** to determine reachability. However, if you can configure *every* router within the internet to accept the *identical* number of hops, you may set **RIP Network Diameter** to a maximum value of 127.

Once you set RIP Network Diameter, you add the RIP basic parameters, by selecting **RIP Basic** and then **Add**. NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

RIP Basic Parameters Stored.

After you have added the RIP basic parameters, you configure RIP on an interface-by-interface basis.

5.5.1.2 Updating RIP Basic Parameters

To update RIP basic parameters, you reset the **RIP Network Diameter** parameter in the **RIP** window.

Note

Because internet convention uses a value of 15 to denote infinity (at which point RIP declares a network unreachable), you should accept the default value 15 when you set **RIP Network Diameter**. For RIP to operate properly, *every* router within the network must use the same network diameter value. Hosts also use **RIP Network Diameter** to determine reachability. However, if you can configure *every* router within the internet to accept the *identical* number of hops, you may set **RIP Network Diameter** to a maximum value of 127.

You may then update the RIP basic parameters, by selecting and then . NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

RIP Basic Parameters Stored.

You may now configure RIP on an interface-by-interface basis.

5.5.1.3 Deleting RIP Basic Parameters

To delete RIP basic parameters, select and then in the **RIP** window. NCU deletes the RIP basic parameters.

5.5.1.4 Adding RIP Interfaces

Adding a RIP interface consists of configuring RIP interface-specific parameters. First, select the interface that you want to configure in the **RIP** window (in Figure 5-14, the network operator selected the interface **G_S21**). Then select and to display the **RIP INTERFACE DEFINITION** window for the interface (see Figure 5-15).

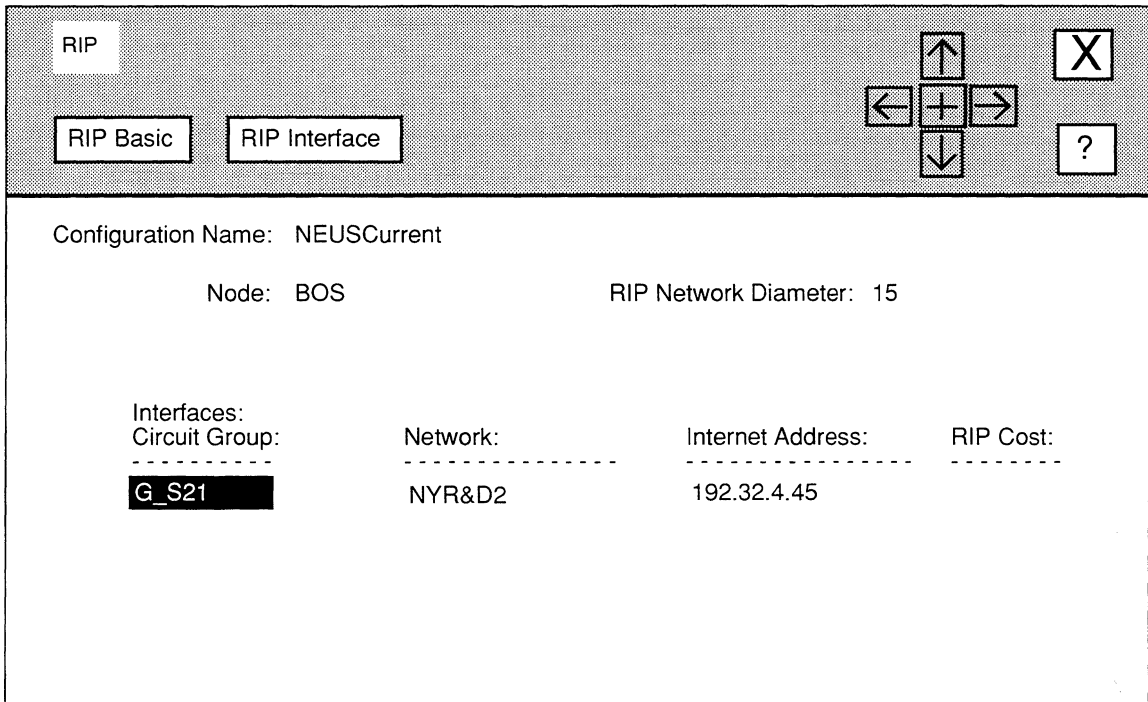


Figure 5-14. RIP Window with Interface Selected

The **RIP INTERFACE DEFINITION** window for an interface displays the interface's internet address and circuit group name, as well as the node and the network that the interface connects, and the associated network configuration. The **RIP INTERFACE DEFINITION** window allows you to reset RIP interface-specific parameters, as follows:

1. At RIP Supply, select how the IP router transmits periodic RIP updates to neighboring routers within the network.

YES Specifies that the router transmits RIP updates. If you wish to supply default route information, set RIP Supply to **YES**.

NO Specifies that the router does not transmit updates. If the interface provides X.25 DDN service, set **RIP Supply** to **NO** (X.25 DDN services does not support RIP).

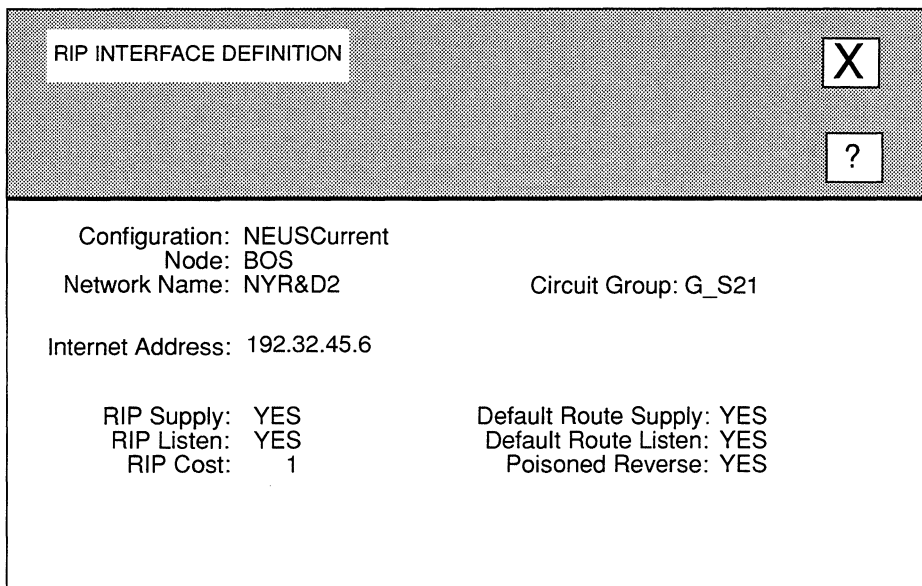


Figure 5-15. RIP INTERFACE DEFINITION Window

- At **RIP Listen**, select how the IP router adds routing information (that it receives in RIP updates from neighboring routers) to its internal routing table.

YES Specifies that the router adds received routing information to its internal routing table. Set **RIP Listen** to **YES**, if you wish to listen for default-route information.

NO Specifies that the router does not add received routing information to its internal routing table. If the interface provides X.25 DDN service, set **RIP Listen** to **NO** (X.25 DDN services does not support RIP).

- At **RIP Cost**, select the cost for each router hop.

Standard RIP implementations assign a cost of 1 to each hop. You can increase this cost by entering a new value. However, if you increase **RIP Cost**, the upper bound set by **RIP Network Diameter** (beyond which the router declares a network unreachable) is more rapidly attained.

4. **At Default Route Supply, select how the IP router advertises default routes in RIP updates it sends to neighboring routers.**
 - YES** Specifies that the router advertises default routes. If you set **Default Route Supply** to **YES**, you must set **RIP Supply** to **YES**.
 - NO** Specifies that the router does not advertise default routes. If the interface provides X.25 DDN service, set **Default Route Supply** to **NO** (X.25 DDN services does not support RIP).

5. **At Default Route Listen, select how the IP router adds network and sub-network default route information (that it receives in RIP updates from neighboring routers) to its internal routing table.**
 - YES** Specifies that the router adds received default-route information to this table. If you set **Default Route Listen** to **YES**, you must set **RIP Listen** to **YES**.
 - NO** Specifies that the router does not add received default-route information to this table. If the interface provides X.25 DDN service, set **Default Route Listen** to **NO** (X.25 DDN services does not support RIP).

6. **At Poisoned Reverse, select how the router advertises routes (that it has learned from a neighboring router) in the periodic updates it sends to that neighbor.**
 - YES** Specifies that the node implements poisoned reverse, which means that the router advertises the routes (that it learns from a neighboring router) in RIP updates (that it sends to that neighbor) with a hop count of **RIP Network Diameter** plus 1; thus declaring the network unreachable.
 - NO** Specifies that the router implements a split-horizon, which means that the router omits routes learned from a neighbor in the RIP updates it sends to that neighbor.

7. **Select and then .**

NCU displays the following window; press **[RETURN]** to clear it from the screen:

Press return when done.
RIP Interface Parameters Stored.

NCU returns to the **RIP** window which now lists the RIP cost associated with the interface you just configured. Repeat this procedure to configure RIP to run across additional interfaces.

5.5.1.5 Updating RIP Interfaces

Updating a RIP interface consists of resetting RIP interface-specific parameters. First, select the RIP interface that you wish to update under **Interfaces** in the **RIP** window, and then select and . NCU displays the **RIP INTERFACE DEFINITION** window, which displays the current parameters for that interface. See *Section 5.5.1, Editing RIP Parameters*, for information on how to reset these parameters.

5.5.1.6 Deleting RIP Interfaces

You delete RIP interfaces from the **RIP** window. First, select the RIP interface that you wish to delete, then select and .

5.5.2 Editing EGP Parameters

EGP, short for *Exterior Gateway Protocol*, facilitates the exchange of IP datagrams between autonomous systems.

You configure EGP from the **EGP** window (see Figure 5-16). To display the **EGP** window, select and then in the **NODE IP CONFIGURATION** window. The **EGP** window allows you to add, update, and delete EGP basic parameters, as well as, to add, update, and delete EGP neighbor parameters.

5.5.2.1 Adding EGP Basic Parameters

EGP basic parameters are two parameters: **Auto Enable** and **Local ASN**. You configure these parameters in the **EGP** window, as follows:

1. **At Auto Enable, select the initial state of EGP.**

This EGP-specific **Auto Enable** works with global **Auto Enable** to enable or disable EGP when the node boots. Depending on the global **Auto Enable** setting, you specify this EGP-specific **Auto Enable**, as follows:

- Global **Auto Enable** set to **NO**.

EGP is unconditionally disabled. It does not matter how you set the EGP-specific **Auto Enable**. You need to enable EGP manually with the NCL Interpreter after the node boots.

EGP

EGP Basic EGP Neighbors

Configuration Name: NEUSCurrent

Node: BOS

Auto Enable: **YES**

Local ASN:

Local Address: Remote ASN: Remote Address:

Figure 5-16. EGP Window

- ❑ Global **Auto Enable** set to **YES**.

EGP is conditionally enabled. Select **YES** to enable EGP, or select **NO** to disable EGP. If you select **NO**, you will subsequently need to enable EGP manually with the NCL Interpreter after the node boots.

2. At **Local ASN** enter the **NIC-assigned decimal number that identifies the local autonomous system**.
3. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the screen.

Press return when done.

EGP Basic Parameters Stored.

After setting EGP basic parameters, you identify EGP neighbors.

5.5.2.2 Updating EGP Basic Parameters

To update EGP basic parameters, you reset the **Auto Enable** and/or **Local ASN** in the **EGP** window, as follows:

1. At Auto Enable specify the initial state of EGP.

This EGP-specific **Auto Enable** works with global **Auto Enable** to enable or disable EGP when the node boots. Depending on the global **Auto Enable** setting, you specify this EGP-specific **Auto Enable**, as follows:

- Global **Auto Enable** set to **NO**.

EGP is unconditionally disabled. It does not matter how you set the EGP-specific **Auto Enable**. You need to enable EGP manually with the NCL Interpreter after the node boots.

- Global **Auto Enable** set to **YES**.

EGP is conditionally enabled. Select **YES** to enable EGP, or select **NO** to disable EGP. If you select **NO**, you will subsequently need to enable EGP manually with the NCL Interpreter after the node boots.

2. At Local ASN enter the NIC-assigned decimal number that identifies the local autonomous system.

3. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the screen.

Press return when done.

EGP Basic Parameters Stored.

5.5.2.3 Deleting EGP Basic Parameters

To delete EGP basic parameters, select and then in the **EGP** window.

5.5.2.4 Adding EGP Neighbors

An EGP neighbor is a router in a remote autonomous system with which the local router exchanges routing information. You add EGP neighbors, as follows:

1. Select and then in the EGP window.

NCU displays the **EGP NEIGHBORS** window (see Figure 5-17).

2. At Local ASN, enter the NIC-assigned decimal number that identifies the local autonomous system.

3. **At Local Address, enter the IP address (in dotted decimal notation) of the local interface that establishes the connection to the remote autonomous system.**
4. **At Remote ASN, enter the NIC-assigned decimal number that identifies the remote autonomous system.**
5. **At Remote Address, enter the IP address (in dotted decimal notation) of the remote interface that establishes the connection to the remote autonomous system.**
6. **At Acquisition Mode, select which of the two neighbors initiates EGP connections.**

ACTIVE Specifies that the local interface initiates EGP connections.

PASSIVE Specifies that the remote interface initiates EGP connections.

EGP connections are initiated when one neighbor issues an *acquisition request message*, and finalized when the recipient of the *acquisition request message* issues an *acquisition confirm response*. A neighbor who issues *acquisition request messages* is said to be active; a neighbor who responds to such messages is said to be passive. Although the EGP protocol allows both neighbors to be active, protocol efficiency is enhanced when one neighbor is active and the other neighbor is passive.

7. **At Polling Mode, select the neighbor-reachability algorithm.**

ACTIVE Specifies that the local router issues periodic *Hello* and *Poll* commands. Neighbor reachability is verified by receipt of corresponding I-H-U (I Hear You) and Update responses.

PASSIVE Specifies that the local router does not issue *Hello* commands; nor does it expect I-H-U responses. Neighbor reachability is verified by examining the Status field of received *Hello* or *Poll* commands, or of Update responses.

BOTH Specifies that the neighboring routes arbitrate a mutually agreeable neighbor-reachability algorithm.

Note

Although the EGP protocol allows both neighbors to be active, you enhance protocol efficiency when one neighbor is active and the other neighbor is passive.

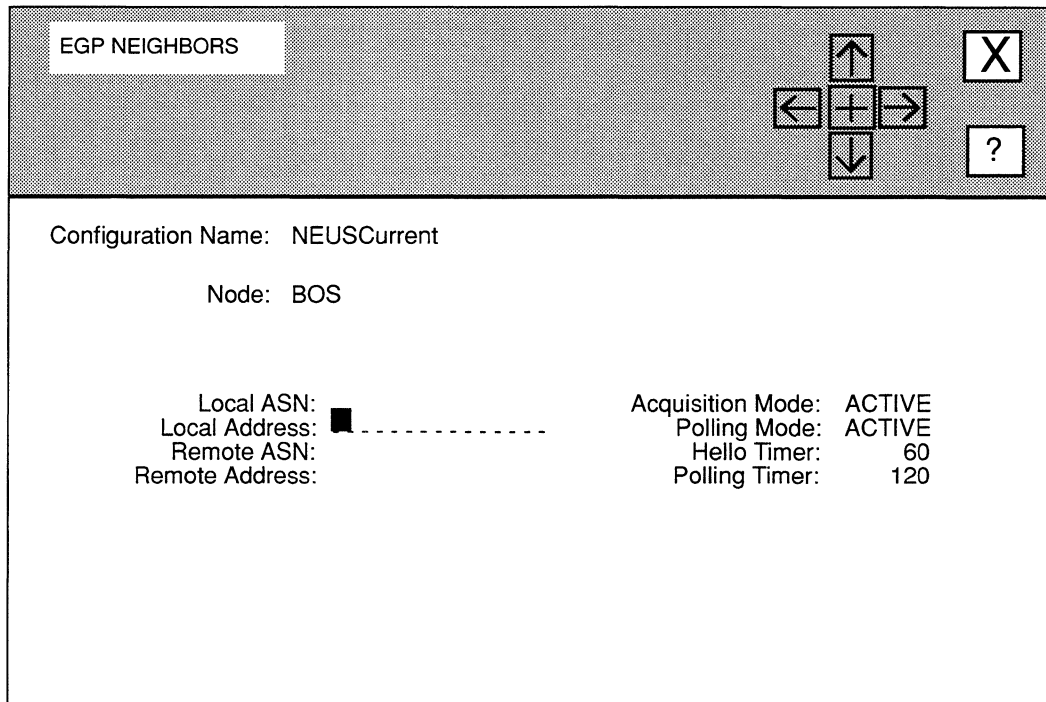
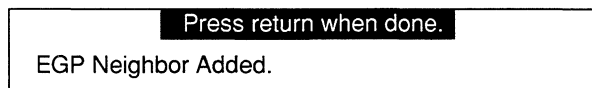


Figure 5-17. EGP NEIGHBORS window

8. At *Hello Timer*, enter the number of seconds between *Hello* commands.
9. At *Polling Timer*, enter the number of seconds between *Poll* Commands.
10. Select **X** and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen:



NCU returns to the **EGP** window which now lists the EGP neighbor. Repeat this procedure to configure additional EGP neighbors.

5.5.2.5 Updating EGP Neighbors

To update an EGP neighbor, first select the neighbor under **Local Address:** in the **EGP** window, and then select and . NCU displays the **EGP NEIGHBORS** window which displays the current parameter settings for that neighbor. See *Section 5.5.2.4, Adding EGP Neighbors* for information on how to reset the parameters.

5.5.2.6 Deleting EGP Neighbors

To delete an EGP neighbor, first select the neighbor under **Local Address:** in the **EGP** window, and then select and . NCU deletes the EGP neighbor.

5.5.3 Editing OSPF Parameters

OSPF, short for Open Shortest Path First, is, like RIP, an internal gateway routing protocol. Unlike RIP, however, OSPF uses a link-state algorithm to route datagrams through an internet.

For RIP (a distance-vector protocol), the “best” path between source and destination is the shortest path. RIP computes distance as a metric, usually the number of hops from the origin network to the target network. For RIP, the best path is the one with the fewest hops.

OSPF is more sophisticated in recognizing that a simplistic hop-count takes no account of available bandwidth. Passing through an extra hop to get to a 1.54 Mb T1 channel, for instance, may be more efficient than traversing a shorter, but congested route. For OSPF, the “best” path is the one that offers the least delay.

In order to reduce the level of protocol traffic, OSPF allows collections of contiguous networks and hosts to be grouped together. This grouping of contiguous networks and hosts along with the routers having an interface(s) to any of the included networks is called an *area*. The topology of an area is invisible to non-area residents; similarly routers that reside within a single area know nothing of the topology external to the area. Figure 5-18 illustrates OSPF area configuration.

Segmentation of an autonomous system in OSPF areas leads to two types of routing:

- ❑ *Intra-area*

Routes packets between sources and destinations that reside within the same area. In intra-area routing, packets are routed solely on the basis of information obtained within the area; external information need not (and can not) be used.

❑ *Inter-area*

Routes packets between sources and destinations that reside within different areas. In inter-area routing, packets are routed in three stages:

1. An internal router (a router whose directly connected networks all reside within the same area) directs the packet to an area border router (a router that services multiple areas).
2. The area border router directs the packet across the OSPF backbone to the destination network.
3. An internal router forwards the packet to the destination.

Table 5-4 briefly describes the types of routers within an OSPF domain. Router types are not mutually exclusive: area border routers are also backbone routers, while backbone routers can, depending upon the topology, be internal routers.

The OSPF *backbone* consists of networks not contained within any area (**Net 5** in Figure 5-18), their attached routers (**R_7** in Figure 5-18), and those routers that belong to multiple areas (**R_3** in Figure 5-18). Area border routers (**R_3**, **R_4**, and **R_8** in Figure 5-18) and routers that attach only to the OSPF backbone (**R_6** in Figure 5-18) must be configured to reflect their backbone connectivity.

Table 5-4. OSPF Routers

Router	Description
Internal Routers	A router with all directly connected networks belonging to the same area (R_1 and R_2 in Figure 5-18). An internal router maintains a single copy of the routing algorithm.
Area Border Routers	A router with directly connected networks belonging to more than one area. An area border router maintains a copy of the routing algorithm for each attached network and for the OSPF backbone (networks not contained within any area, their attached routers, and routers that service multiple areas).
Backbone Routers	A router that connects to the backbone. By definition, area border routers are backbone routers. Routers with all interfaces connected to the backbone are considered to be internal routers.
AS Boundary Routers	A router that exchanges routing information with other autonomous systems.

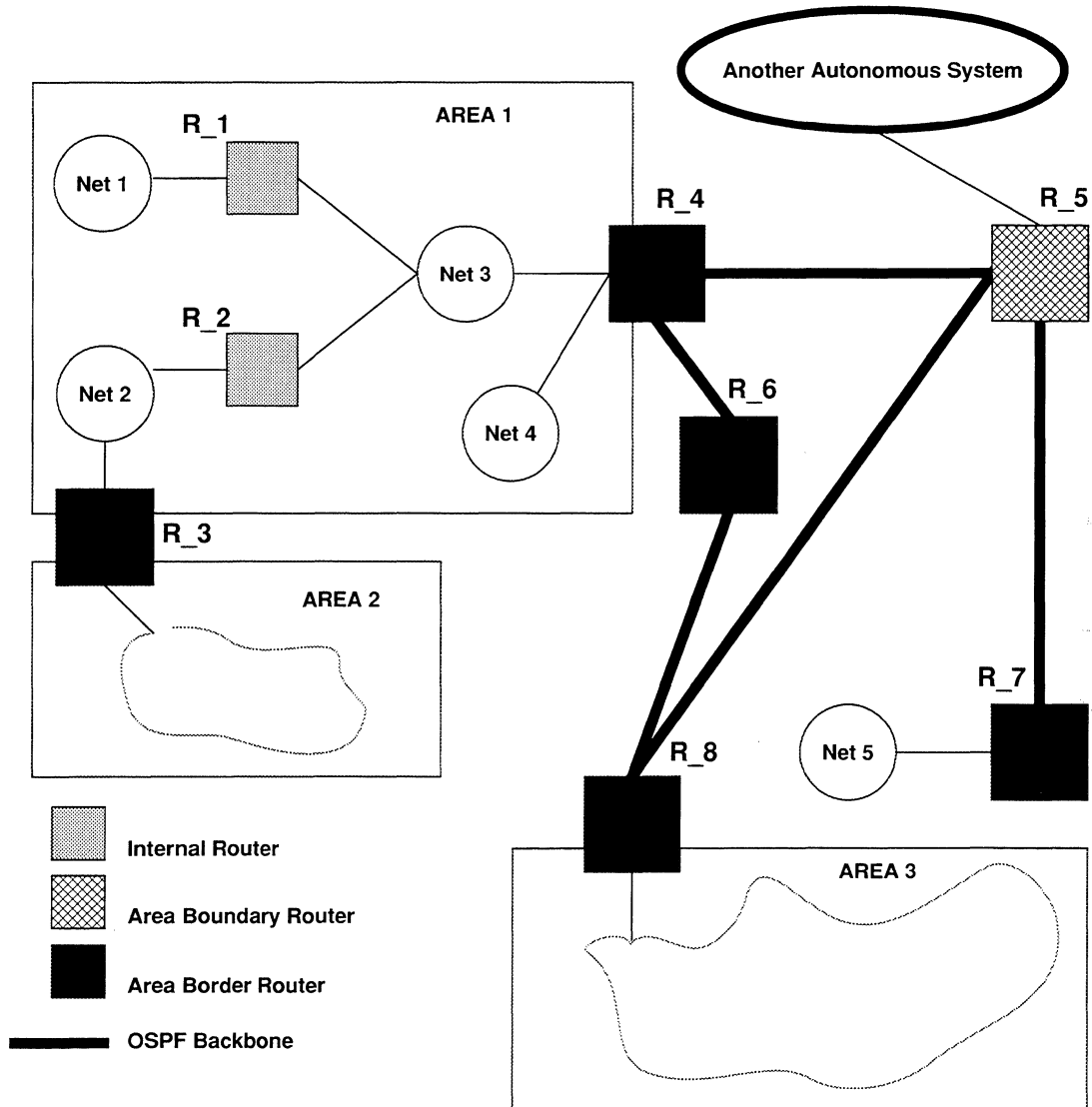


Figure 5-18. Sample OSPF Topology

The backbone distributes routing information between OSPF areas. The backbone has all the properties of an OSPF area (the topology of the backbone is hidden from other areas, while other areas know nothing of the backbone topology). The area id 0.0.0.0 is assigned to the backbone.

The OSPF backbone must be contiguous. Depending on network topology and area definition, it is possible to construct an OSPF topology in which the backbone is no longer contiguous. For example, in Figure 5-18, **R_3** is not contiguous to the OSPF backbone. In such cases, *virtual links* are required to restore contiguity.

Virtual links are statically configured backbone components that join two backbone routers that have an interface to a common non-backbone area (**R_3** and **R_4** in Figure 5-18). The OSPF protocol treats a virtual link as if it were a point-to-point network connection between the two backbone routers.

You configure OSPF from the **OSPF** window (see Figure 5-20). To display the **OSPF** window, select and then in the **NODE IP CONFIGURATION** window. The **OSPF** window allows you to add, update, and delete OSPF basic parameters, as well as, to add, update, and delete OSPF areas.

5.5.3.1 Adding OSPF Basic Parameters

You add OSPF basic parameters in the **OSPF** window, as follows:

1. At Auto Enable, select the initial state of OSPF.

This OSPF-specific **Auto Enable** works with global **Auto Enable** to enable or disable OSPF when the node boots. Depending on the global **Auto Enable** setting, you specify this OSPF-specific **Auto Enable**, as follows:

- Global **Auto Enable** set to **NO**.

OSPF is unconditionally disabled. It does not matter how you set the OSPF-specific **Auto Enable**. You need to enable OSPF manually with the NCL Interpreter after the node boots.

- Global **Auto Enable** set to **YES**.

OSPF is conditionally enabled. Select **YES** to enable OSPF, or select **NO** to disable OSPF. If you select **NO**, you will subsequently need to enable OSPF manually with the NCL Interpreter after the node boots.

2. At Router ID, enter the numeric identifier (in dotted decimal notation) that uniquely identifies the IP router within the OSPF domain.

Note

One algorithm for **Router ID** assignment is to choose the largest or smallest IP address assigned to the router.

3. At AS Boundary, select the access to the OSPF routing pool.

YES Specifies that the OSPF routing pool includes manually configured routes and routes obtained from RIP and EGP.

NO Specifies that the OSPF routing pool includes only those routes acquired by OSPF.

4. At External Route Preference, select a default weighted value (a decimal value in the range 1 to 16) to manually configured static routes and to routes acquired by OSPF from other routing protocols (RIP and EGP).**Note**

Leave External Route Preference blank to accept the default routing hierarchy.

Once enabled, the IP router maintains a routing pool consisting of routes learned from OSPF, from other enabled routing protocols (RIP and EGP) and, optionally, manually configured static and default routes. Consequently, the routing pool may contain multiple routes for a single destination.

If confronted with multiple routes, the IP router, by default, grants preference to (chooses) a manually-configured static or default route, as follows:

- In the absence of a manually configured route, the IP router chooses an OSPF-derived route.
- Lacking an OSPF route, it chooses an EGP route.
- In the absence of either an OSPF or EGP route, it chooses a RIP-derived route.

Figure 5-19 illustrates the routing pool hierarchy.

5. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

OSPF Basic Parameters Stored.

After adding the OSPF basic parameters, you configure OSPF areas.

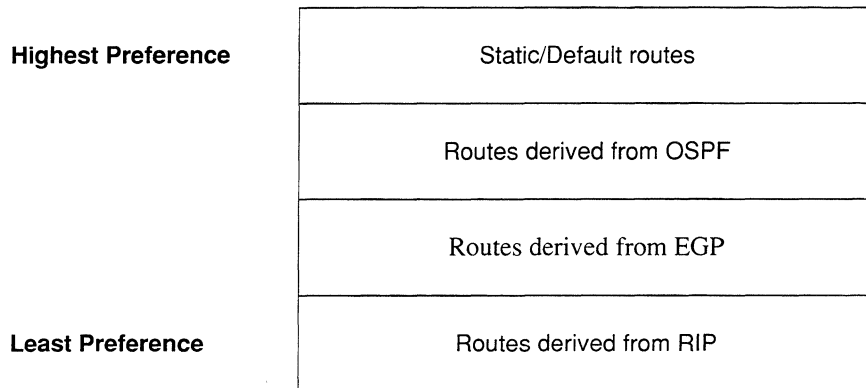


Figure 5-19. Routing Hierarchy

5.5.3.2 Updating OSPF Basic Parameters

You update OSPF basic parameters in the **OSPF** window. Simply reset the parameters (see *Section 5.5.3.1, Adding OSPF Basic Parameters*, and then select and .

NCU displays this window; press **[RETURN]** to clear it from the console.

```
Press return when done.  
OSPF Basic Parameters Stored.
```

5.5.3.3 Deleting OSPF Basic Parameters

You delete OSPF basic parameters in the OSPF window. Select and .

NCU displays this window; press **[RETURN]** to clear it from the console.

```
Press return when done.  
OSPF Basic Parameters deleted.
```

5.5.3.4 Configuring OSPF Areas and Backbone Connections

An OSPF *area* is a collection of contiguous networks, hosts, and routers. The topology of an area is invisible to non-area residents; similarly routers that reside within a single area know nothing of the topology outside the area.

The OSPF *backbone* consists of networks not contained within any area, their attached routers, and those routers that belong to multiple areas. You must configure the backbone connection for area-border routers, and routers that attach only to the OSPF backbone.

The following sections describe how to configure OSPF areas and backbone connections.

5.5.3.4.1 Adding Backbone Connections

If the IP router connects to the OSPF backbone, complete the following procedure to establish the backbone connection. If the IP router does not connect to the backbone, go directly to *Section 5.5.3.4.2, Adding OSPF Areas*.

You add a backbone connection, as follows:

1. Select and then in the OSPF window.

NCU displays the **OSPF AREA** window (see Figure 5-21).

2. At **Area Type**, select **BACKBONE**.

NCU automatically sets **Area ID** to **0.0.0.0** (which identifies the OSPF backbone) and automatically sets **Stub Area** to **NO**.

3. At **Authentication**, enable or disable password authentication.

SIMPLE PASSWORD Enables password authentication across the OSPF backbone. All OSPF packet exchanges can be authenticated by means of a password contained within the OSPF packet. Authentication is enabled on an area basis.

NO AUTHENTICATION Disables password authentication.

4. Select in the OSPF window.

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

OSPF Area stored.

After configuring the OSPF backbone connection, you configure OSPF backbone networks.

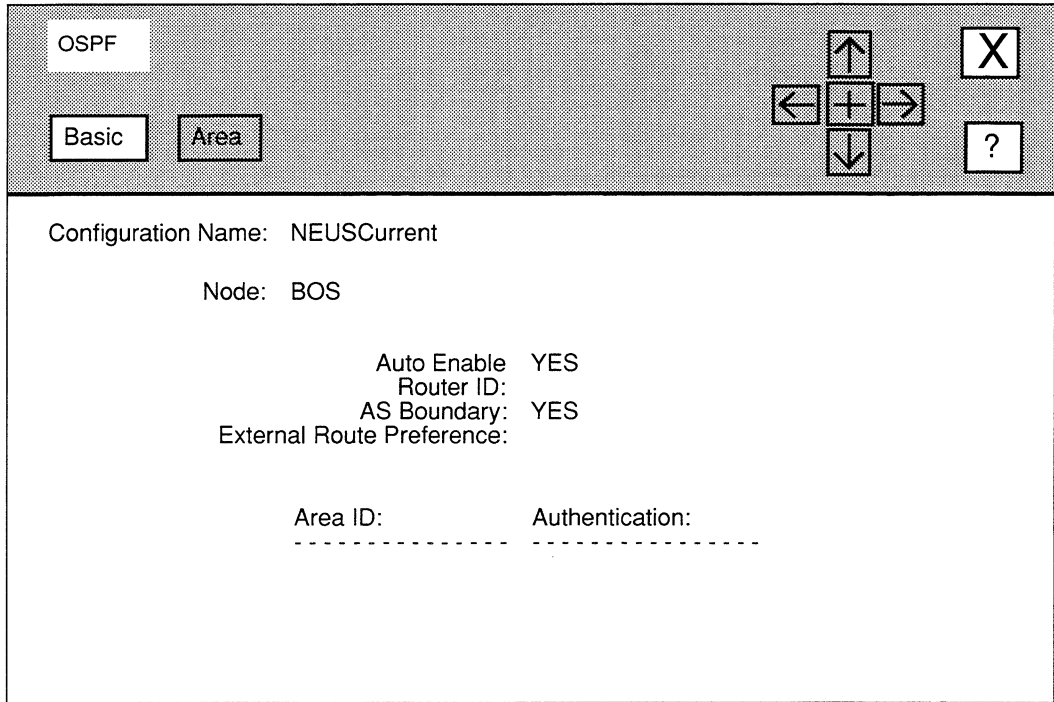


Figure 5-20. OSPF Window

5.5.3.4.1.1 Adding Backbone Networks

An OSPF backbone network is a network not contained within any area (for example, **Net 5** in Figure 5-18). If your network topology includes such a network(s), you first configure OSPF backbone network connections, and then configure the OSPF backbone network, as follows:

1. Select in the OSPF AREA window.
NCU displays the **OSPF NETWORKS** window (see Figure 5-22).
2. Select and then in the OSPF AREA Window.
NCU displays the **ADD OSPF NETWORK** window (see Figure 5-23).

OSPF AREA

Save Area! Network! Interface! Virtual Links!

Configuration Name: NEUSCurrent

Node: BOS

Area Type: **NON-BACKBONE** Stub Area: NO

Area ID:

Authentication: SIMPLE PASSWORD

Figure 5-21. OSPF AREA Window

3. At IP Address, enter the IP network address in dotted decimal notation of the backbone network.
4. At Network Mask, enter the network mask value in dotted decimal notation.
5. Select **X** and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

OSPF Network stored.

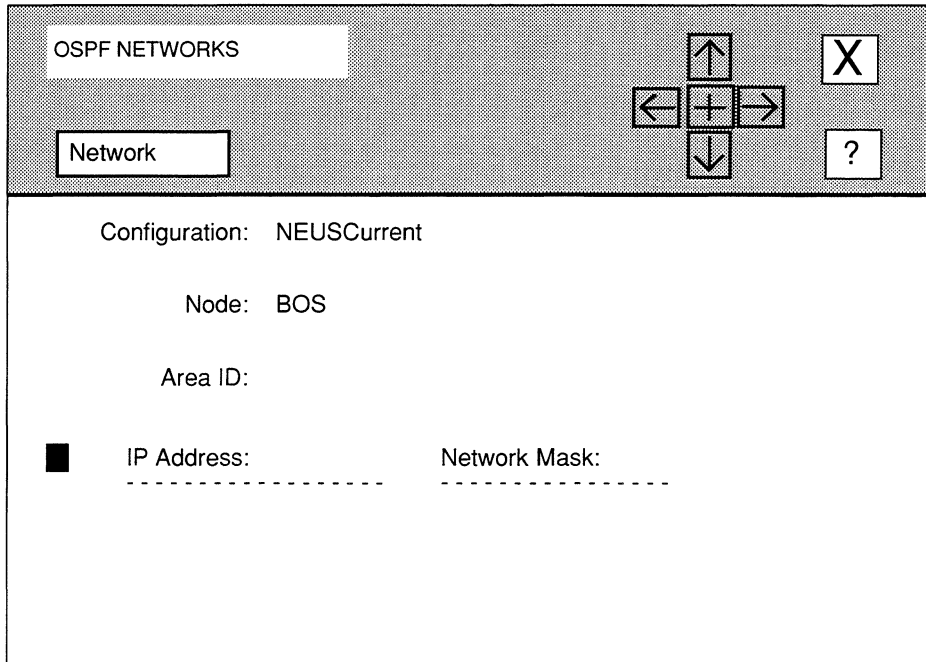


Figure 5-22. OSPF NETWORKS Window

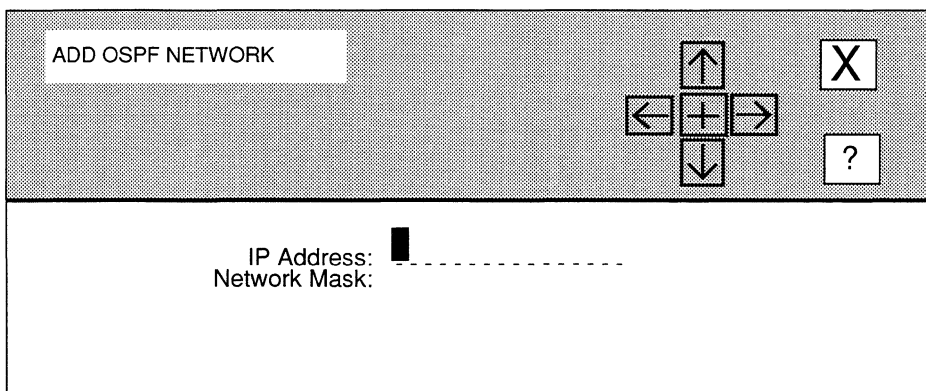


Figure 5-23. ADD OSPF NETWORK Window

NCU returns to the **OSPF NETWORKS** window which now lists the IP address and network mask associated with the backbone network you just configured. Repeat this procedure to configure additional backbone networks.

After configuring all OSPF backbone networks, select and then in the **OSPF NETWORKS** window to return to the **OSPF AREA** window. After configuring the OSPF backbone networks, you configure OSPF backbone interfaces.

5.5.3.4.1.2 Adding Backbone Virtual Links

After configuring the interfaces for the OSPF backbone, you configure any virtual links that are required to ensure backbone contiguity. For example, referring to Figure 5-18, and assuming that you were configuring **R_3**, you would configure a virtual link from **R_3** to **R_4**.

You add backbone virtual links from the **OSPF AREA** window, as follows:

1. Select .

NCU displays the **OSPF VIRTUAL LINK** window (see Figure 5-24).

2. Select and then .

NCU displays the **ADD VIRTUAL LINK** window (see Figure 5-25).

3. At **Neighbor ID**, enter the router ID (in dotted decimal notation) of the remote end of the virtual link.
4. At **Transit Area**, enter the area ID (in dotted decimal notation) of the transit area through which traffic to Neighbor ID is forwarded.
5. At **Hello Interval**, select the number of seconds between the IP router's transmission of OSPF *Hello* packets.

NCU provides the following responses: 5, 10, 15, 20, 30, or 60.

6. At **Dead Interval**, select the number of seconds before a "silent" router is declared down.

NCU provides responses ranging from 20 seconds to 360 seconds.

7. At **Retransmit Interval**, select the number of seconds between the IP router's retransmission of OSPF link-state advertisements.

NCU provides the following responses: 5, 10, 15, 20, and 30.

8. At **Virtual Interface**, enter the IP address (in dotted decimal notation) of the interface through which to conduct the virtual link.

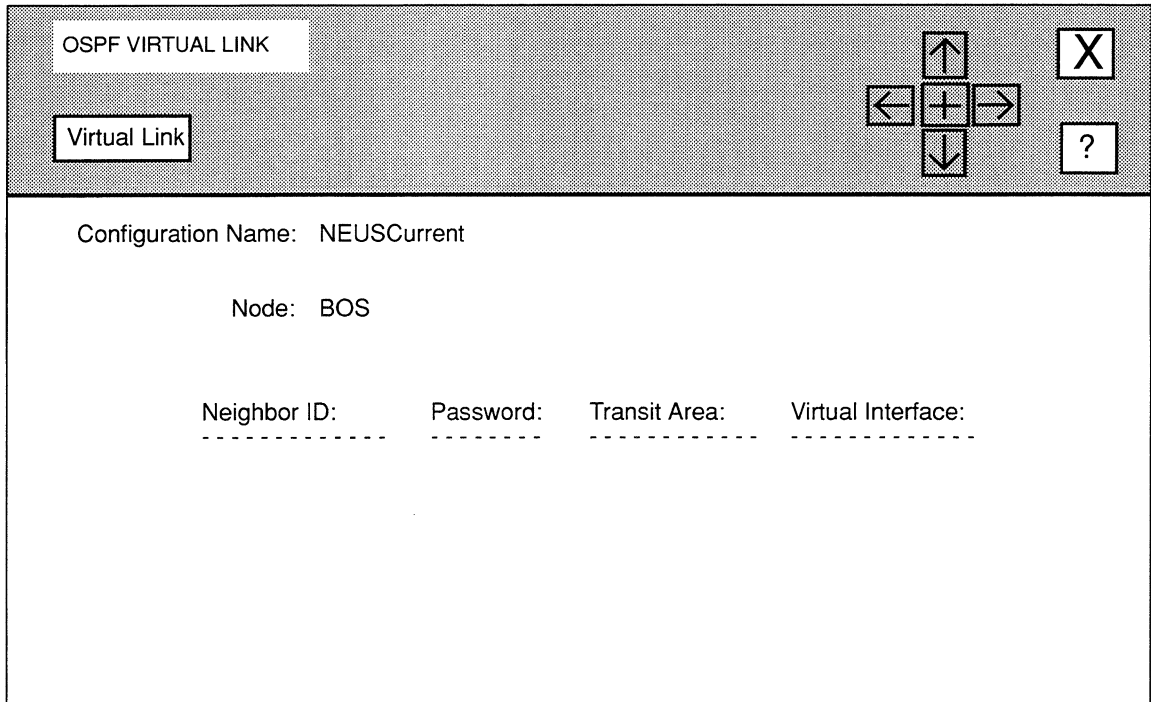


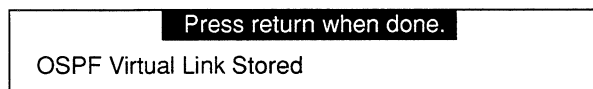
Figure 5-24. OSPF VIRTUAL LINK Window

9. At **Password**, enter the authentication key (a one-to-eight character ASCII string that appears in the authentication field of all OSPF packets across this interface) used across this virtual interface.

Leave **Password** empty, if you do not want authentication across the virtual link.

10. Select **X** and then **Save** .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **OSPF VIRTUAL LINK** window which now displays the virtual link you just configured. Repeat this procedure for each additional backbone virtual link you wish to add.

The screenshot shows a window titled "ADD VIRTUAL LINK". At the top right, there are navigation controls: a central cross with a plus sign, four arrows pointing up, down, left, and right, an "X" button, and a "?" button. The main content area displays the following configuration:

```

Configuration:  NEUSCurrent

                Node:  BOS

Neighbor ID:
Transit Area:
Hello Interval: 5
Dead Interval: 20
Retransmit Interval: 5
Virtual Interface:
Password:

```

Figure 5-25. ADD VIRTUAL LINK Window

5.5.3.4.1.3 Updating Backbone Virtual Links

You update backbone virtual links from the **OSPF VIRTUAL LINK** window. First, select the virtual link under **Neighbor ID**. Next, select and then . NCU displays the **ADD VIRTUAL LINK** window, which displays the current parameter settings for the virtual link. See *Section 5.5.3.4.1.2, Adding Backbone Virtual Links* for information on how to reset the parameters.

5.5.3.4.1.4 Deleting Backbone Virtual Links

You delete backbone virtual links from the **OSPF VIRTUAL LINK** window. First, select the virtual link under **Neighbor ID**. Next, select and then . NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

OSPF Virtual Link deleted

5.5.3.4.2 Adding OSPF Areas

Once you have added the OSPF basic parameters, you may add OSPF areas, as follows:

1. Select and then in the OSPF window.

NCU displays the **OSPF AREA** window (see Figure 5-21).

2. At **Area Type**, select **NON-BACKBONE**.

3. At **Stub Area**, select the area type.

YES Specifies *stub area* type (stub areas carry traffic that is either locally originated or destined). If you select **YES**, NCU displays **Metric** after **Authentication**.

NO Specifies *transit area* type (transit areas can carry/pass-through traffic that is originated by or destined for other OSPF areas). Select **NO** if this OSPF area will carry transit traffic.

4. At **Area ID**, enter a dotted decimal 32-bit number (for example, 1 . 1 . 1 . 1) that identifies the area. If you are assigning subnetted networks as different areas, you can use the 32-bit network address as the Area ID.

Note

The value 0.0.0.0 is reserved for the backbone.

5. At **Authentication**, enable or disable password authentication.

NO AUTHENTICATION Disables password authentication.

SIMPLE PASSWORD Enables password authentication across the OSPF backbone. All OSPF packet exchanges can be authenticated by means of a password contained within the OSPF packet. You enable authentication on an area basis.

6. At **Metric** (which only applies to stub areas), enter a cost to the transit hop from the router to the stub network.

7. Select .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

OSPF Area stored.

After you configure OSPF areas, you configure OSPF area networks.

5.5.3.4.2.1 Adding Area Networks

Previously defined as a collection of networks, OSPF area networks are more precisely a list of address ranges. Each address range is defined by an address/mask pair.

After you configure OSPF areas, you add OSPF area networks from the **OSPF AREA** window, as follows:

1. Select .

NCU displays the **OSPF NETWORKS** window (see Figure 5-22).

2. Select and then .

NCU displays the **ADD OSPF NETWORK** window (see Figure 5-23).

3. At **IP Address** specify an IP network resident within this OSPF area.
4. At **Network Mask** specify the network mask in dotted decimal notation.
5. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

OSPF Network stored.

NCU returns to the **OSPF NETWORKS** window which now lists the IP address and network mask associated with the OSPF area you just configured. Repeat this procedure to assign additional ranges to the OSPF area.

Once you have configured all OSPF area networks, select in the **OSPF NETWORKS** window to return to the **OSPF AREA** window. You may now configure OSPF interfaces.

5.5.3.4.3 Adding OSPF Interfaces

After configuring backbone networks (if any), and/or OSPF area networks, you configure the actual interface(s) between the OSPF backbone router and adjacent backbone routers or networks. For example, refer to Figure 5-18, assuming that you are configuring **R_7**, you would configure the interfaces to **Net 5** and **R_5**.

You add OSPF interfaces, as follows:

1. Select in the **OSPF AREA** window.

NCU displays the **OSPF INTERFACES** window (see Figure 5-26).

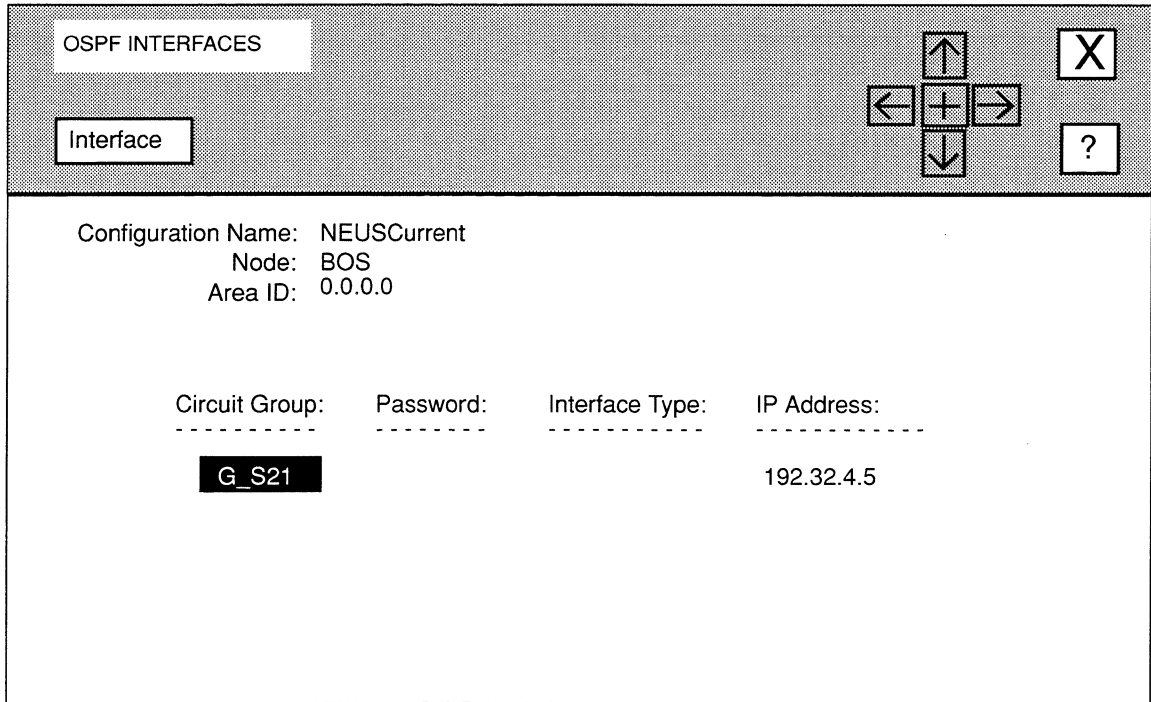


Figure 5-26. OSPF INTERFACES Window with Interface Selected

2. Select the interface that you wish to configure under **Circuit Group** (in Figure 5-26, the network operator selected **G_S21**).
3. Select and then .

NCU displays the **OSPF INTERFACE** window (see Figure 5-27).

4. At **Password**, enter the authentication key (a one-to-eight character ASCII string that appears in the authentication field of all OSPF packets) used across this interface.

If you did not enable authentication across the backbone or area, leave **Password** empty.

5. At **Interface Type**, select the interface type.

BROADCAST Specifies that a broadcast interface connects the router to an Ethernet or IEEE 802.X medium. Select **BROADCAST** if this interface connects to an OSPF broadcast-type media. If you select **BROADCAST**, NCU displays **Priority** after **Retransmit Interval**.

OSPF INTERFACE			
Configuration Name:	NEUSCurrent		
Node:	BOS		
Circuit Group:	G_S21		
Area ID:	0.0.0.0		
Password:			
Interface Type:	BROADCAST		
IP Address:	192.32.4.5	Hello Interval:	
Metric:		Dead Interval:	
		Retransmit Interval:	
		Priority:	

Figure 5-27. OSPF INTERFACE Window

POINT TO POINT Specifies that a point-to-point interface connects the router to a remote peer or to a packet switched network such as Telenet or the DDN. Select **POINT TO POINT** if this interface connects to a remote peer or to an X.25 service provider.

6. **Do nothing at IP Address; NCU displays the IP address of the interface you selected in step 2.**
7. **At Metric, enter a cost to the transit hop from the router across the interface.**
8. **At Hello Interval, enter the number of seconds between the IP router's transmission of OSPF *Hello* packets.**

Hello packets are transmitted across each OSPF interface. Broadcast interfaces use *Hello* packets to select the designated and the backup designated router, and to discover and maintain neighbor relationships.

Note

You must configure all IP routers connected to the OSPF backbone with the same values for **Hello Interval**.

9. **At Dead Interval, select the number of seconds the IP router waits before it declares a “silent” router down.**

NCU provides responses ranging from **20** seconds to **360** seconds.

Note

You must configure all IP routers connected to the OSPF backbone with the same values for **Dead Interval**.

10. **At Retransmit Interval, select the number of seconds between the IP router’s retransmission of OSPF link state advertisements.**

NCU provides three responses: **5** seconds, **10** seconds, **15** seconds, **20** second, and **30** seconds.

Note

You should set Retransmit Interval to a value greater than the expected round trip delay between any two routers on the backbone.

11. **At Priority (displayed only if you set Interface Type to Broadcast), select a weighted value used in the designated router and backup designated router selection algorithm.**

0 Specifies that this router is ineligible for election to designated or backup designated router.

5, 10, 15, 20, or 30..... Each of the five values specify that the router is eligible for election to designated or backup designated router. When two routers attached to the backbone both attempt to become designated router, the one with the highest **Priority** value takes precedence. In the case of equal **Priority** values, the router with the highest **Router ID** takes precedence.

12. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

OSPF Interface stored.

NCU returns to the **OSPF INTERFACES** window which now displays the **Password** and **Interface Type** you just configured. Repeat this procedure to configure additional OSPF interfaces.

5.5.3.4.4 Updating OSPF Interfaces

You update OSPF interfaces from the **OSPF INTERFACES** window. First, select the interface under **Circuit Group**. Next, select and then . NCU displays the **OSPF INTERFACE** window with the current settings for the interface. For instruction on how to reset the parameters, see *Section 5.5.3.4.3, Adding OSPF Interfaces*.

5.5.3.4.5 Deleting OSPF Interfaces

You delete OSPF interfaces from the **OSPF INTERFACES** window. First, select the interface you wish to delete. Next, select and then . NCU displays this window; press **[RETURN]** to clear it from the console:

```
Press return when done.
OSPF Interface deleted.
```

5.5.3.5 Updating OSPF Areas

You update OSPF areas from the **OSPF** window. First, select the area under **Area ID**. Next, select and . NCU displays the **OSPF Area** window which displays the current settings for that area. For information on how to reset the parameters, see the following sections:

- ❑ If the area is a backbone (has an **Area ID** of **0.0.0.0**), see *Section 5.5.3.4.1, Adding Backbone Connections*.
- ❑ If the area is not a backbone (has an **Area ID** other than **0.0.0.0**), see *Section 5.5.3.4.2, Adding OSPF Areas*.

5.5.3.6 Deleting OSPF Areas

You delete OSPF areas from the **OSPF** window. First, select the area under **Area ID**. Next, select and then . NCU displays this window; press **[RETURN]** to clear it from the console:

```
Press return when done.
OSPF Area deleted.
```


5.5.4 Editing Static and Default Routes

This section describes how to edit:

❑ Static routes

Static routes specify transmission paths that datagrams follow based on the datagram's destination address. You should configure a static route if:

- Your implementation requires a “protected” network, and you want to restrict the paths datagrams follow to paths that you specifically configure.

In this situation, you must disable address resolution so that you turn off all dynamic routing capabilities, and you must disable all RIP and default-route supply and listen activities. You would then configure all routes statically.

- Your topology includes a network or hosts that do not implement ARP.

In this situation, you must configure a type of static route (called an adjacent host route) to each non-ARP host.

- Your topology includes “hidden” subnets on a single network interface.

In this situation, you must configure an adjacent host route to a specified node; the node, in turn, routes datagrams to destination hosts on the “hidden” Net.

❑ Default Routes

A default route is a form of static route. Default routes minimize the size of the internal routing table, and reduce the data transmitted in periodic routing table updates. Default routes are most efficiently used when the IP router has a small number of directly connected networks, and has a single connection to another routing device as shown in Figure 5-28.

Upon receiving a datagram, the IP router scans its internal table for the destination address. With a default route specified, if the router does not find the destination address, it uses the default route. For example, in Figure 5-28, the router directs datagrams explicitly addressed to Networks **A**, **B**, or **C** to the proper recipient over network interfaces **A**, **B**, or **C**. It directs all other addresses to a neighboring router by way of the default route **D**.

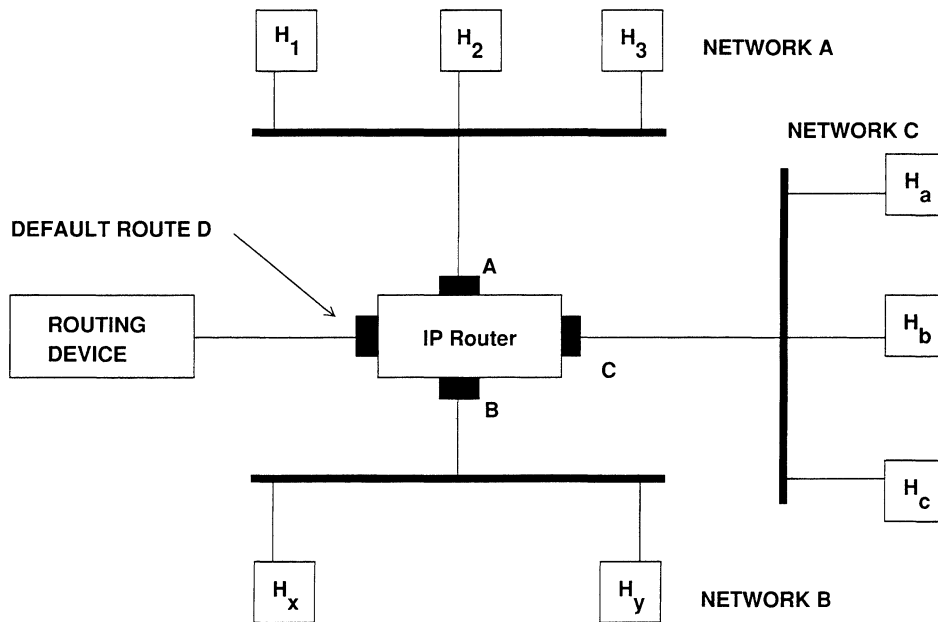
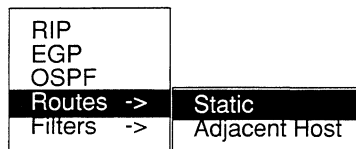


Figure 5-28. Sample Default Route Topology

You edit static and default routes from the **STATIC ROUTES** window. To display this window, do the following:

1. Select and then in the **NODE IP CONFIGURATION** window to display this sub-menu:



2. Select to display the **STATIC ROUTES** window.

The **STATIC ROUTES** window (see Figure 5-29) displays the current static and default routes. You may now edit these routes.

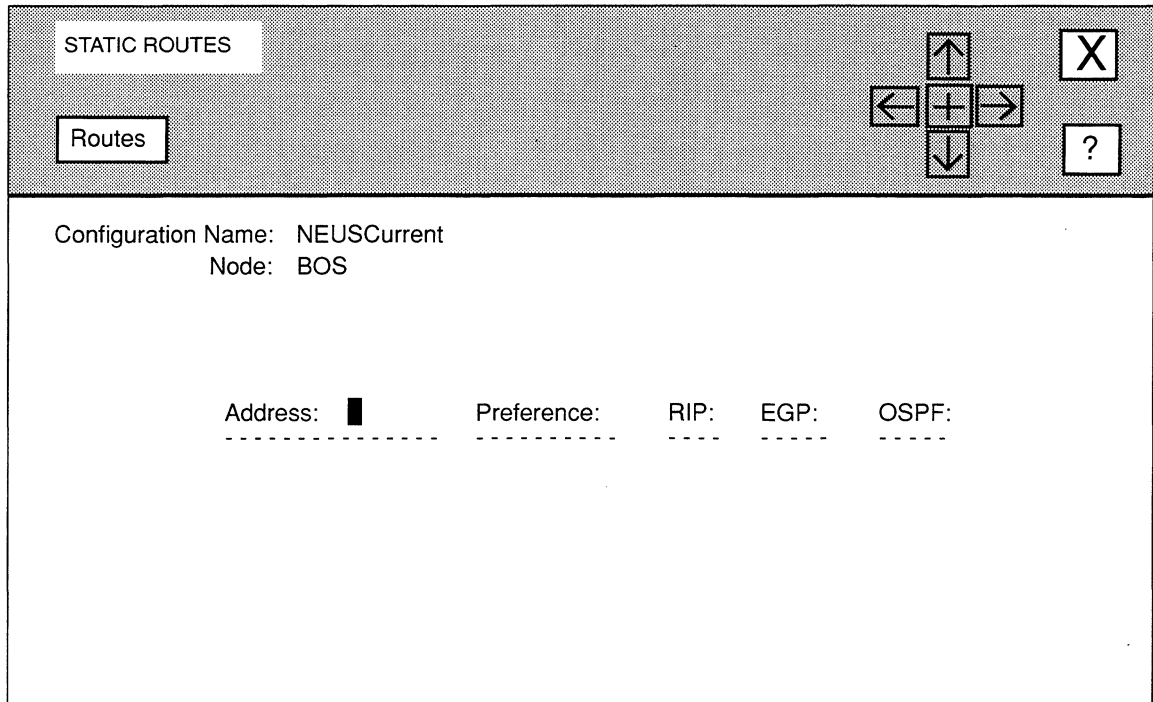


Figure 5-29. STATIC ROUTES Window

5.5.4.1 Adding Static Routes

You add static routes as follows:

1. Select and then to display the **STATIC ROUTE DEFINITION** window.

The **STATIC ROUTE DEFINITION** window (see Figure 5-30) allows you to configure static-route parameters:

2. At **Internet Address**, enter the destination IP address in dotted decimal notation.
3. At **Subnet Mask** specify the range of the static route.

For example, if **Internet Address** is equal to 192.32.1.0 and **Subnet Mask** is equal to 255.255.255.0, the static route applies to all 192.32.1.xx traffic.

STATIC ROUTE DEFINITION

Configuration Name: NEUSCurrent
Node: BOS

Internet Address: █-----
Subnet Mask:
Cost:
Next Hop:
Preference:

Propagate to RIP: YES
Propagate to EGP: YES
Propagate to OSPF: YES

Figure 5-30. STATIC ROUTE DEFINITION Window

4. **At Cost, enter the number of router hops a datagram traverses before reaching its destination.**
Enter the number of intermediate routers to the destination.
5. **At Next Hop, enter the IP address in dotted decimal notation of the next hop router.**
6. **At Preference, enter a weighted value (a number from 0, lowest preference, to 9, highest preference) that the IP router uses to select a single route from multiple routes to the same destination.**

The IP router maintains a routing pool which contains information supplied by up to three routing protocols (RIP, EGP, OSPF), in addition to manually configured static routes. Since the routing pool may contain multiple routes to the same destination, the IP router uses manually configured static and/or default routes in preference to routes gathered by protocol exchanges. Routes with higher preference values will be selected (used for routing) by the IP router over routes with lower preference values.

7. **At Propagate to RIP, select whether the RIP Protocol advertises this static route.**

YES Specifies the RIP protocol advertises this static route. If you select **YES**, you must configure and enable the RIP protocol.

NO Specifies the RIP protocol does not advertise this static route.

8.

9. **At Propagate to EGP, select whether the EGP protocol advertises this static route.**

YES Specifies the EGP protocol advertises this static route. If you select **YES**, you must configure and enable the EGP protocol.

NO Specifies the EGP protocol does not advertise this static route.

10. **At Propagate to OSPF, select whether the OSPF protocol advertises this static route.**

YES Specifies the OSPF protocol advertises this static route. If you select **YES**, you must configure and enable the OSPF protocol.

NO Specifies the OSPF protocol does not advertise this static route.

11. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the screen:

Press return when done.

Static Route parameters stored.

NCU returns to the **STATIC ROUTES** window which now lists the static route you configured.

Repeat this procedure for each additional static route you wish to configure. After you have configured all static routes, select and then Confirm to return to the **NODE IP CONFIGURATION** window.

5.5.4.2 Adding Default Routes

You add default routes, as follows:

1. Select and then in the **STATIC ROUTES** window:

NCU displays the **STATIC ROUTE DEFINITION** window which allows you to configure default-route parameters.

2. At **Internet Address**, enter **0.0.0.0** to designate a default route.
3. At **Subnet Mask**, press **[RETURN]**.

Default routes do not use a network mask

4. At **Cost**, enter the number of router hops that a datagram traverses before it reaches the edge of its destination autonomous system.

Enter the number of intermediate routers to the destination.

5. At **Next Hop**, enter the IP address in dotted decimal notation of the next hop router.

6. At **Preference**, enter a weighted value (a number from 0, lowest preference, to 9, highest preference) that the IP router uses to select a single route from multiple routes to the same destination.

The IP router maintains a routing pool which contains information supplied by up to three routing protocols (RIP, EGP, OSPF), in addition to manually configured static routes. Since the routing pool may contain multiple routes to the same destination, the IP router uses manually configured static and/or default routes in preference to routes gathered by protocol exchanges. Routes with higher preference values will be selected (used for routing) by the IP router over routes with lower preference values.

7. At **Propagate to RIP**, select whether the **RIP Protocol** advertises this default route.

YES Specifies the RIP protocol advertises this default route. If you select **YES**, you must configure and enable the RIP protocol.

NO Specifies the RIP protocol does not advertise this static route.

8. At **Propagate to EGP**, select whether the **EGP protocol** advertises this default route.

YES Specifies the EGP protocol advertises this default route. If you select **YES**, you must configure and enable the EGP protocol.

NO Specifies the EGP protocol does not advertise this static route.

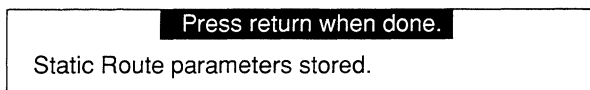
9. At Propagate to OSPF, select whether the OSPF protocol advertises this default route.

YES Specifies the OSPF protocol advertises this default route. If you select **YES**, you must configure and enable the OSPF protocol.

NO Specifies the OSPF protocol does not advertise this static route.

10. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen:



NCU returns to the **STATIC ROUTES** window which now lists the static default route you configured.

Repeat this procedure for each additional default route you wish to configure. After you have configured all default routes, select and then to return to the **NODE IP CONFIGURATION** window.

5.5.5 Updating Static and Default Routes

You update static and default routes from the **STATIC ROUTES** window. First select the route you wish to update (in Figure 5-31, the network operator selected the route **192.32.4.34**), next select and then . NCU displays the **STATIC ROUTE DEFINITION** window which allows you to change the parameters for the route you selected (see *Section 5.5.4, Editing Static and Default Routes* for instructions on how to set the parameters).

5.5.6 Deleting Static and Default Routes

You delete static and default routes from the **STATIC ROUTES** window. First select the route you wish to delete, then select and . NCU deletes the route.

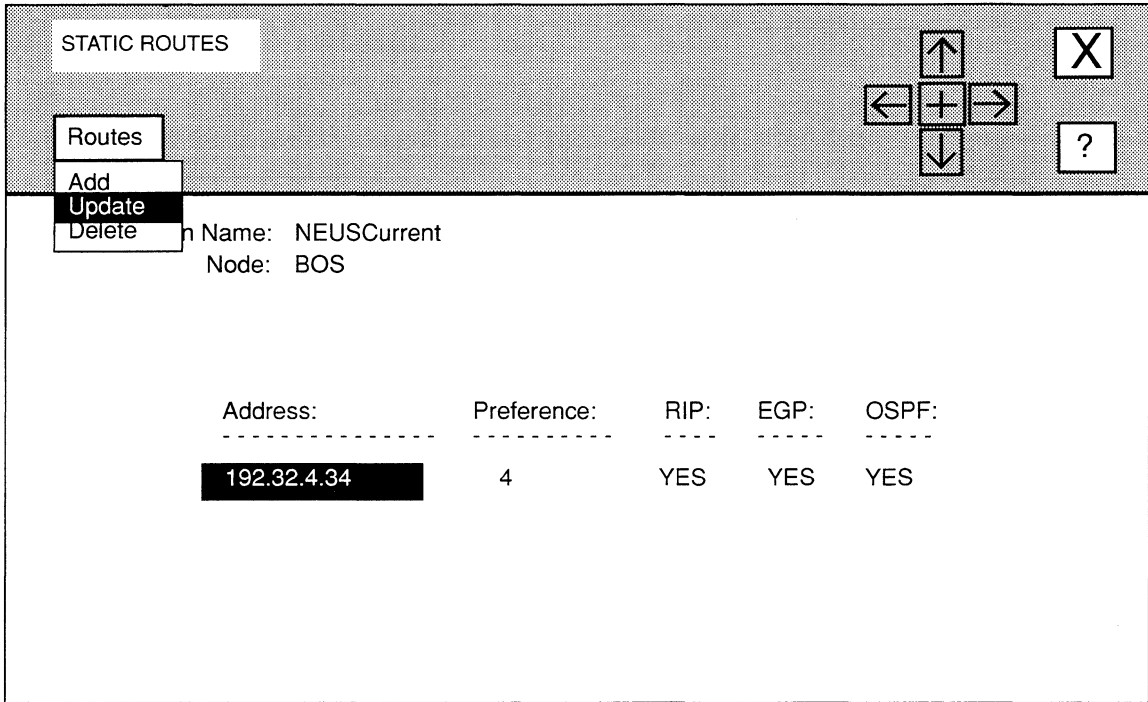


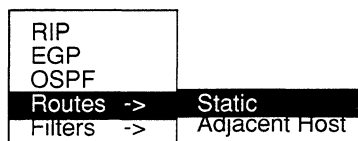
Figure 5-31. STATIC ROUTES Window with Route Selected

5.5.7 Editing Adjacent Host Routes

Adjacent hosts are nodes on a locally-attached network. You need to specify an adjacent host if you are setting up a protected network; or if a particular local host or hosts do not respond to ARP requests; or if the router network interface contains hidden subnets.

The **ADJACENT HOST ROUTES** window allows you to edit adjacent host routes. To display this window, do the following:

1. Select **Routing** and then **Routes ->** in the **NODE IP CONFIGURATION** window to display this sub-menu:



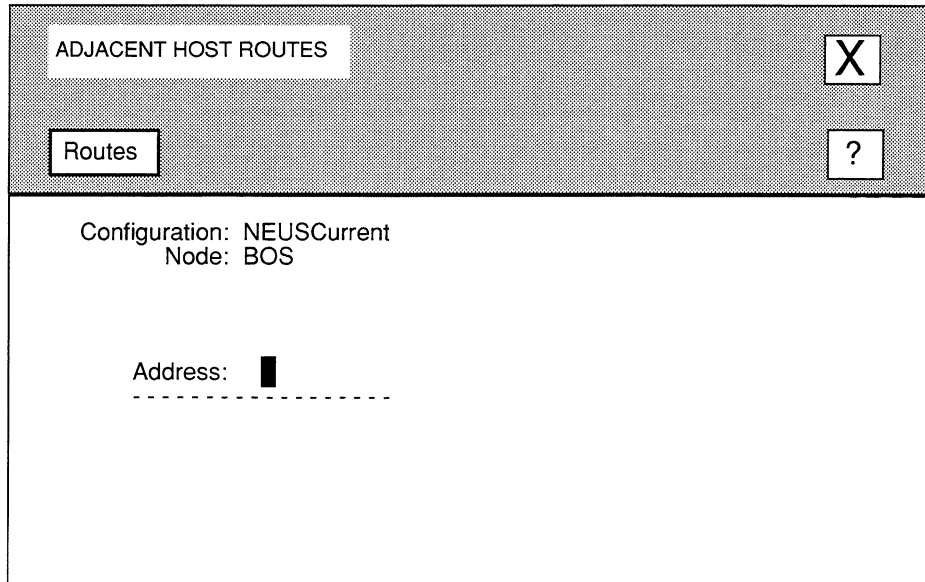


Figure 5-32. ADJACENT HOST ROUTES Window

2. Select to display the **ADJACENT HOST ROUTES** window.

The **ADJACENT HOST ROUTES** window (see Figure 5-32) displays the current adjacent host routes. You may now add, update, and delete adjacent host routes.

5.5.7.1 Adding Adjacent Host Routes

You add an adjacent host route from the **ADJACENT HOST ROUTES** window, as follows:

1. Select to display the **ADJACENT HOST ROUTE** window (see Figure 5-33).
2. At **Internet Address**, enter the IP address in dotted decimal notation of the adjacent host.
3. At **Subnet Mask**, enter the part of **Internet Address** that refers to the subnet.

If **LAN Address** is located in a subnet, at **Subnet Mask**, enter the subnet mask value in dotted decimal notation. If the host address is not in a subnet, leave **Subnet Mask** blank.

ADJACENT HOST ROUTE	
	X
	?
Configuration: NEUSCurrent Node: BOS	
Internet Address: <input type="text"/>	
Subnet Mask: <input type="text"/>	
LAN Address: <input type="text"/>	
Encapsulation: ETHERNET	

Figure 5-33. ADJACENT HOST ROUTE Window

4. At LAN Address, enter the 48-bit Ethernet address (as a 12-digit hexadecimal number) of the adjacent host.
5. At Encapsulation, enter the encapsulation method that the Internet Address uses.

Note

If you are defining a LAN interface (Ethernet or IEEE 802.x), you must specify the encapsulation method supported by the attached network. If you are defining any type of point-to-point network interface, you must select standard Ethernet 2.0 encapsulation.

- | | |
|-----------------------|---|
| ETHERNET | Specifies standard Ethernet 2.0 encapsulation (see Figure 5-34). Ethernet encapsulation prefixes an eight-octet preamble, six octets of destination-address information, six octets of source-address information, and two octets of protocol type information (hexadecimal 0800) to the IP packet. It appends a four-octet frame check sequence to the packet. |
| 802.2 | Specifies 802.2 encapsulation (see Figure 5-35), which prefixes one octet of destination service access point |

(DSAP) information, one octet of source service access point (SSAP) information, and one octet of control information to the IP packet. The 802.2 structure is further encapsulated within a medium-specific 802.x packet.

SNAP Specifies SNAP encapsulation (see Figure 5-36), which is an extension of 802.2 encapsulation. It prefixes one octet of DSAP information (hexadecimal AA), one octet of SSAP information (hexadecimal AA), one octet of control information, three octets of organizational information (hexadecimal 0), and two octets of Ethernet Type information (hexadecimal 0800) to the IP packet. The SNAP structure is further encapsulated within a medium-specific 802.x packet.

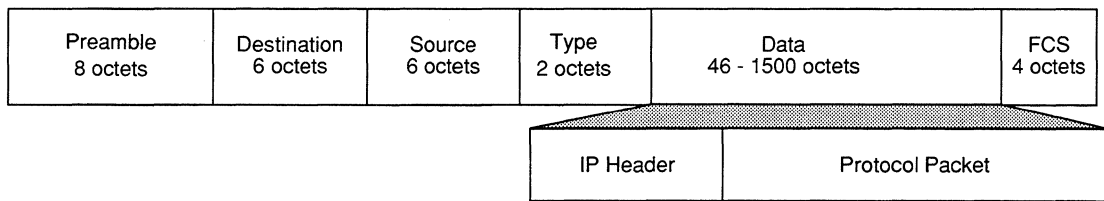


Figure 5-34. IP Ethernet Encapsulation

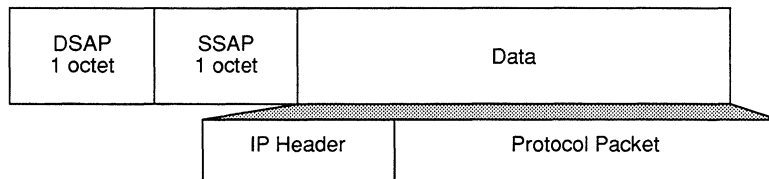


Figure 5-35. IP 802.2 Encapsulation

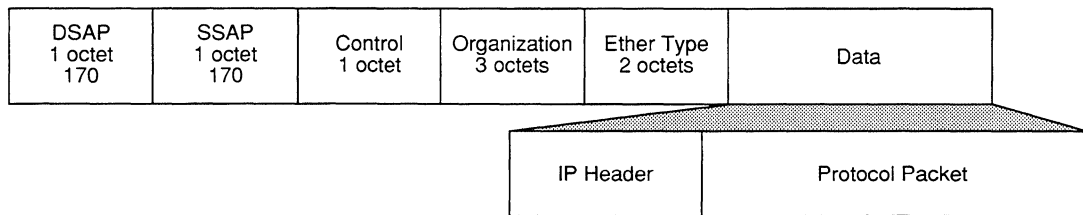


Figure 5-36. IP SNAP Encapsulation

6. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen:

Press return when done.
Adjacent Host Route parameters stored.

NCU returns to the **ADJACENT HOST ROUTES** window which now lists the adjacent host route you configured.

Repeat this procedure for each additional adjacent host route you wish to configure. After you have configured all adjacent host routes, select and then to return to the **NODE IP CONFIGURATION** window.

5.5.7.2 Updating Adjacent Host Routes

You update adjacent routes from the **ADJACENT HOST ROUTES** window. First select the route you wish to update, next select and then . NCU displays the **ADJACENT HOST ROUTE** window which allows you to change the parameters for the adjacent host route you selected (see *Section 5.5.7.1, Adding Adjacent Host Routes* for instructions on how to set the parameters).

5.5.7.3 Deleting Adjacent Host Routes

You delete adjacent host routes from the **ADJACENT HOST ROUTES** window. First select the route you wish to delete, then select and . NCU deletes the adjacent host route.

5.5.8 Configuring Filters

NCU allows you to configure address filters, import-route filters, and export-route filters. The following sections describe how to configure these filters.

5.5.8.1 Configuring Address Filters

Address and port filters provide security and traffic control by allowing you to specify which datagrams are to be routed across the internet. Source-address filters allow you to filter a datagram on the basis of the source address contained in the datagram's IP header. Destination-address filters allow you to filter a datagram on the basis of the destination address contained in the datagram's IP header.

Filtering is a sequential process in which the node examines, in the following order, the:

- IP source address
- IP destination address
- TCP or UDP destination port address

NCU allows you to add, update, and delete address filters. You edit address filters from the **ADDRESS FILTERS** window. You display this window, as follows:

1. Select and then to display this sub-menu:

Source/Dest. Address
Import Route
Export Route

2. Select .

NCU displays the **ADDRESS FILTERS** window (see Figure 5-37).

5.5.8.1.1 Adding Address Filters

You add source- and destination-address filters from the **ADDRESS FILTERS** window, as follows:

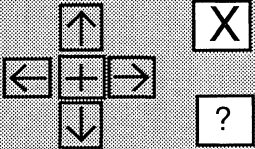
1. Select and then .

NCU displays the **ADD ADDRESS FILTER** window (see Figure 5-38).

2. **At IP Address, enter the filtered source or destination address in dotted decimal notation.**

To filter all source or destination addresses, enter **0.0.0.0**. To filter a specific IP source or destination address, enter the address in dotted decimal notation.

ADDRESS FILTERS



Configuration Name: NEUSCurrent

Node: BOS

IP Address:	Filter Type:	Action:

Figure 5-37. ADDRESS FILTERS Window

3. At Address Mask specify which portion of IP Address is filtered.

For example, consider Class C Network 192.32.1.0, which allocates the upper 2 bits of the host-identification field to Subnet_ID, and the final 6 bits to Host_ID. Within this network, IP address 132.32.1.129 specifies Host 1 of Subnet 2.

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.0 is subject to filtering. If you enter **255.255.255.0** at **Address Mask**, only the Net_ID portion of the address will be filtered. If you enter **255.255.255.192** at **Address Mask**, the Net_ID and Subnet_ID portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Address Mask**, the entire IP address will be filtered.

4. At Filter Type, select IP header field (SOURCE ADDRESS or DESTINATION ADDRESS) to be filtered.

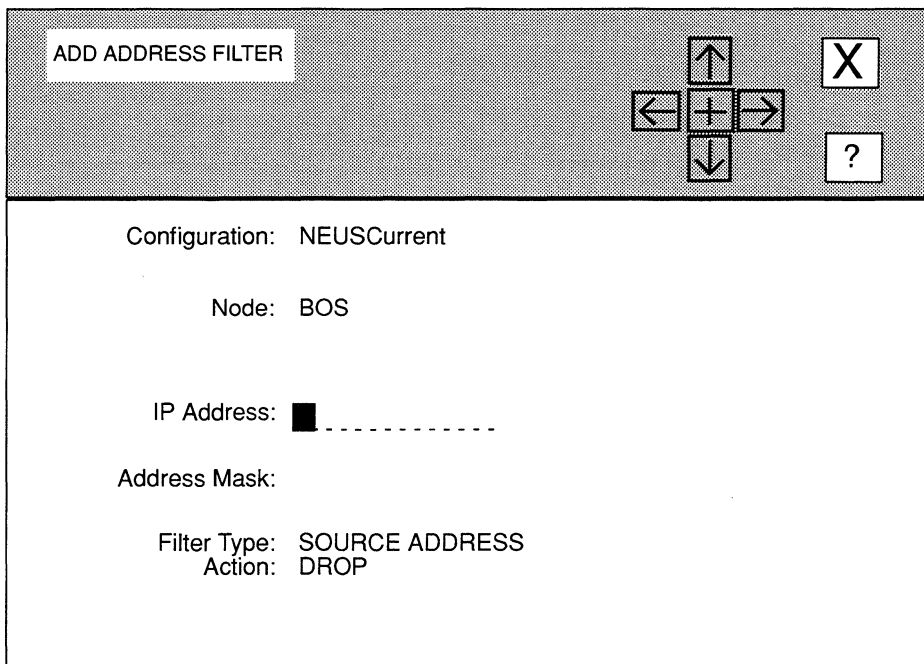


Figure 5-38. ADD ADDRESS FILTER Window

5. At Action, select how you wish the node to dispose of packets that match the filter rule.

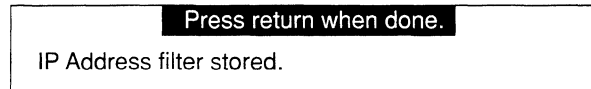
FORWARD Specifies the node transmits datagrams that match the filter rule.

DROP Specifies the node discards datagrams that match the filter rule.

CONTINUE Specifies the node takes no immediate action on the basis of a matched source **IP Address** (do not use **CONTINUE** for destination-address filters). Instead, the node examines the datagram's IP destination address in order to make a filtering decision. With no matching destination filter, the node forwards the datagram.

6. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **ADDRESS FILTERS** window which now displays the address filter you just configured. Repeat this procedure to configure additional address filters.

5.5.8.1.2 Updating Address Filters

You update address filters from the **ADDRESS FILTERS** window. First, select the address filter under **IP Address**. Next, select Filter and then Update . NCU displays the **ADD ADDRESS FILTER** window which displays the current parameter settings for the filter. For information on how to reset the parameters, see *Section 5.5.8.1.1, Adding Address Filters*.

5.5.8.1.3 Deleting Address Filters

You delete address filters from the **ADDRESS FILTERS** window. First, select the address filter under **IP Address**. Next, select Filter and then Delete . NCU deletes the filter.

5.5.8.2 Configuring Routing-Pool Filters

The node maintains routing information from any one, or each, of the three routing protocols that you enabled (RIP, OSPF, and EGP) in a common routing pool. Thus, the routing pool can contain multiple routes (one per protocol) to a specific destination. Each route carries an associated preference value which determines the “best” route (where more than one route to the same destination is available). On the basis of this preference, each instance of the IP router constructs a forwarding table which lists the “best” route to all known destinations.

The node updates the routing pool in response to received protocol traffic; routing pool updates are subsequently reflected in updated forwarding tables. The routing protocols also use the pool as a database from which they extract information to prepare their link/state advertisements.

Wellfleet's IP allows you to manage the flow of routing data to and from the routing pool. User-configured import and export rules provide this control:

- ❑ Import rules define how new routes are added to the routing pool.
Each routing protocol (OSPF, EGP, or RIP) maintains a distinct set of import rules. For example, when RIP receives a new routing update, RIP consults its specific import rules to validate the information before it inserts the update in the routing pool. Import rules contain search information (which is used to match fields in incoming routing updates) and action information (which specifies the action to take with matched fields).
- ❑ Export rules define how the routing protocols propagate known routes.
Each routing protocol maintains a distinct set of export rules. For example, when preparing a routing advertisement, RIP consults its specific export rules to determine whether routes to specific networks are to be advertised and how they are to be propagated. Export rules contain network numbers (which are used to associate a rule with a specific network) and action information (which specifies a route propagation procedure).

Figure 5-39 depicts, conceptually, the relationship between the routing pool, forwarding tables, and the import and export rules for routing data.

NCU allows you to configure import-route filters (filters that follow import rules) and export-route filters (filters that follow export rules). Table 5-5 provides general information for configuring import and export filters. The following sections describe how to add, update, and delete these filters.

Table 5-5. Import- and Export-Filter Configuration Rules

<i>Rule 1</i>	Each filtering rule must specify an incoming/originating protocol. Rules in the form "ACCEPT Network Number Mask" are invalid.
<i>Rule 2</i>	The routing pool holds one route per-protocol-per-network. If there are multiple import rules for a specific network number and protocol, the most-recent routing update received by means of any of these rules is the route active in the routing pool.
<i>Rule 3</i>	Most rules include a network number and mask; however, the network number and mask are not always required.
<i>Rule 4</i>	If you create a rule that contains a network address that is a superset of an address from another rule, the rule for the less-specific address does not affect any packet that matches the more-specific address. The only exception is a rule that specifies no network address.

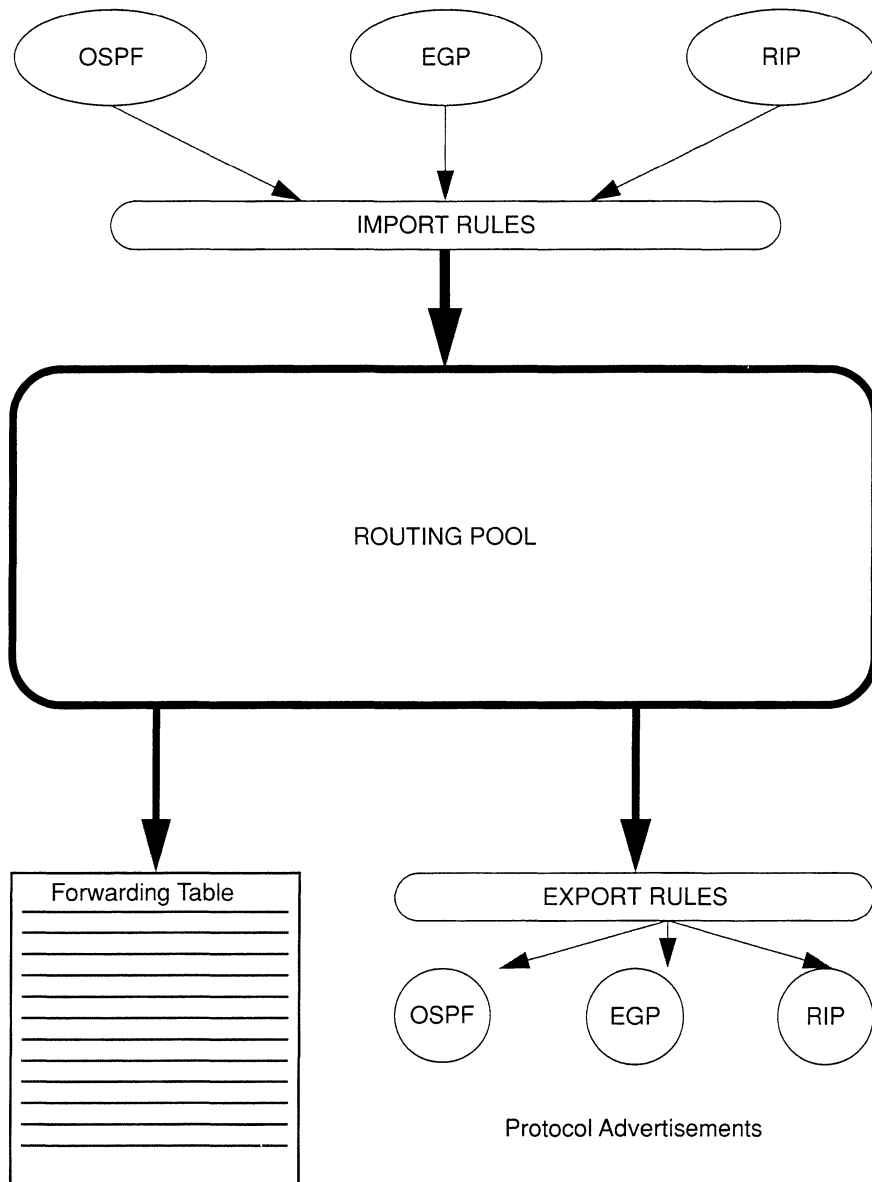


Figure 5-39. Routing Information Data Flow

5.5.8.2.1 Adding Import-Route Filters

You add import-route filters from the **IMPORT ROUTE FILTERS** window. To display this window, select and in the **NODE IP CONFIGURATION** window to display the following sub-menu:

```

Source/Dest. Address
Import Route
Export Route
    
```

Select to display the **IMPORT ROUTE FILTERS** window (see Figure 5-40). You may then add import-route filters. Select to display the sub-menu in Figure 5-40. Depending on the sub-menu option you select, refer to the appropriate section for information on how to add the particular import-route filter.

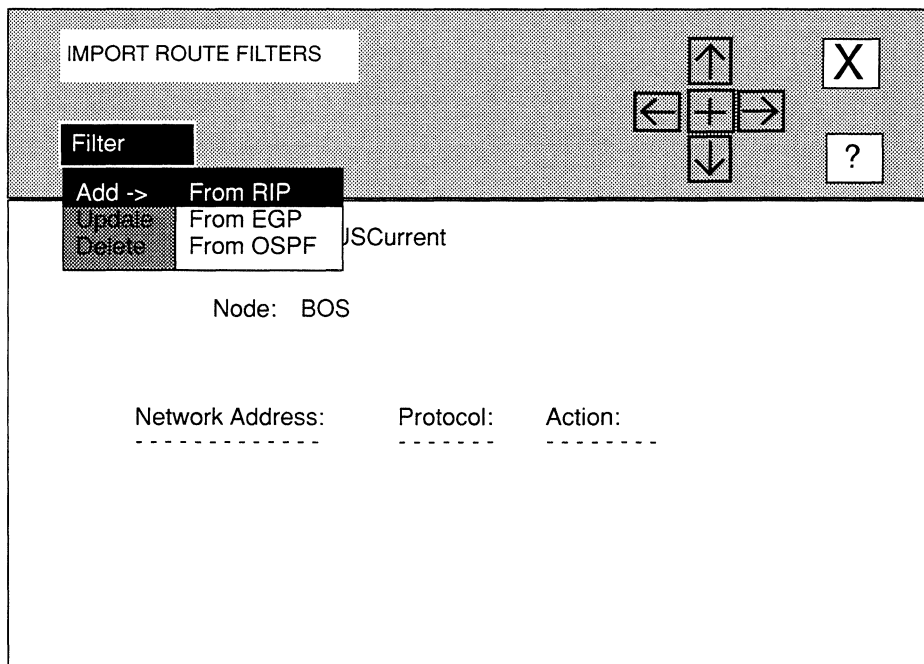


Figure 5-40. IMPORT ROUTE FILTERS Window

5.5.8.2.1.1 Adding a RIP Import Filter

You add a RIP import filter from the **IMPORT FROM RIP** window (see Figure 5-41), as follows:

1. **At Network Address, enter the filtered IP network address in dotted decimal notation.**

To filter all destination networks, leave **Network Address** blank.

2. **At Network Mask, enter a range of addresses upon which the filter acts.**

For example, consider Class C Network 192.32.1.0, which allocates the upper 3 bits of the host identification field to **Subnet_ID**, and the final 5 bits to **Host_ID**.

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.x is subject to filtering. If you enter **255.255.255.0** at **Network Mask**, only the **Net_ID** portion of the address will be filtered. If you enter **255.255.255.224** at **Network Mask**, the **Net_ID** and **Subnet_ID** portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Network Mask**, the entire IP address will be filtered.

3. **At Action, select how the route is transferred to the routing pool.**

ACCEPT Specifies that the information is sent to the routing pool.

IGNORE Specifies the routing information is dropped.

4. **At From Gateway, enter the IP address (in dotted decimal notation) of the gateway from which RIP updates are received.**

If you want the RIP import route filter to be “universal” (applicable to all RIP sources), leave **From Gateway** blank.

5. **At From Interface, enter the IP address (in dotted decimal notation) of an interface across which RIP updates are received.**

If you want the RIP import route filter to be “universal” (applicable to all local interfaces), leave **From Interface** blank.

6. **At Preference (meaningful only when you set Action to ACCEPT), enter a weighted precedence value (from 1 to 10) to a route included in the routing pool.**

By default, routing preference is granted to OSPF routes, then to EGP routes, and then to RIP routes.

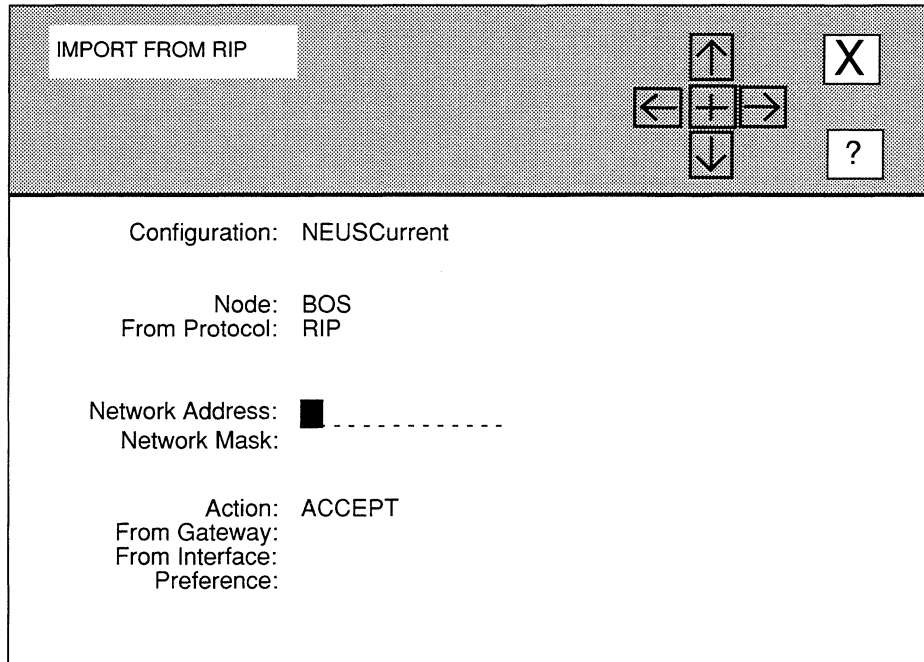
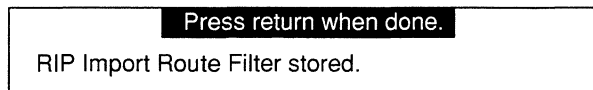


Figure 5-41. IMPORT FROM RIP Window

7. Select **X** and then **Save** .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **IMPORT ROUTE FILTERS** window which now displays the RIP import filter you just configured. Repeat this procedure to configure RIP import filters.

5.5.8.2.1.2 Adding an EGP Import Filter

You add an EGP import filter from the **IMPORT FROM EGP** window (see Figure 5-42), as follows:

1. **At Network Address, enter the filtered IP network address in dotted decimal notation.**

If you want to filter all destination networks, leave **Network Address** blank.

2. **At Network Mask, enter a range of addresses upon which the filter acts.**

For example, consider Class C Network 192.32.1.0, which allocates the upper 3 bits of the host identification field to `Subnet_ID`, and the final 5 bits to `Host_ID`.

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.x is subject to filtering. If you enter **255.255.255.0** at **Network Mask**, only the `Net_ID` portion of the address will be filtered. If you enter **255.255.255.224** at **Network Mask**, the `Net_ID` and `Subnet_ID` portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Network Mask**, the entire IP address will be filtered.

3. **At Action, select how the route is transferred to the routing pool.**

ACCEPT Specifies the information is sent to the routing pool.

IGNORE Specifies the routing information is dropped.

4. **At Peer, enter the IP address (in dotted decimal notation) of a router from which EGP updates are received.**

If you wish the EGP import route filter to be “universal” (applicable to all foreign EGP routers), leave **Peer** blank.

5. **At Autonomous System, enter the NIC-assigned identification number of an autonomous system from which RIP update are received.**

If you wish the EGP import route filter to be “universal” (applicable to all foreign autonomous systems), leave **Autonomous System** blank.

6. **At Preference (meaningful only when you set Action to ACCEPT), enter a weighted precedence value (from 1 to 10) to a route included in the routing pool.**

By default, routing preference is granted to OSPF routes, then to EGP routes, and then to RIP routes.

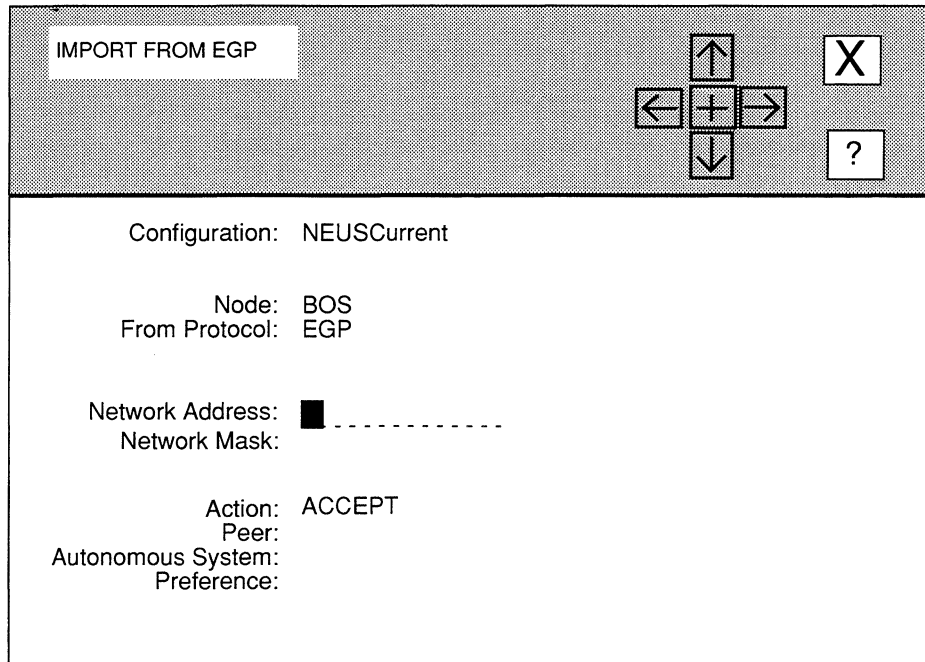


Figure 5-42. IMPORT FROM EGP Window

7. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **IMPORT ROUTE FILTERS** window which now displays the EGP import filter you just configured. Repeat this procedure to configure EGP import filters.

5.5.8.2.1.3 Adding an OSPF Import Filter

OSPF import route filters manage the flow of OSPF external routes into the routing pool. OSPF import route filters have no affect on internal routes; such routes are always places in the routing pool.

IMPORT FROM OSPF

Configuration: NEUSCurrent

Node: BOS
From Protocol: OSPF

Network Address: [blacked out] -----
Network Mask: -----

Action: ACCEPT
Type: INTERNAL
Tag: -----
Preference: -----

Figure 5-43. IMPORT FROM OSPF Window

You add an OSPF import filter from the **IMPORT FROM OSPF** window (see Figure 5-43), as follows:

1. **At Network Address, enter the filtered IP network address in dotted decimal notation.**

If you want to filter all destination networks, leave **Network Address** blank.

2. **At Network Mask, enter a range of addresses upon which the filter acts.**

For example, consider Class C Network 192.32.1.0, which allocates the upper 3 bits of the host identification field to `Subnet_ID`, and the final 5 bits to `Host_ID`.

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.x is subject to filtering. If you enter **255.255.255.0** at **Network Mask**, only the `Net_ID` portion of the address will be filtered. If you enter **255.255.255.224** at **Network Mask**, the `Net_ID` and `Subnet_ID` portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Network Mask**, the entire IP address will be filtered.

3. **At Action, select how the route is transferred to the routing pool.**

ACCEPT..... Specifies the information is sent to the routing pool.

IGNORE Specifies the routing information is dropped.

4. **At Type, select the OSPF external-metrics type to which you want the OSPF import-route filter to apply.**

INTERNAL..... Specifies Type 1 metrics, which are equivalent to the standard OSPF link-state metric.

EXTERNAL..... Specifies Type 2 metrics, which are greater than the cost of any path internal to the autonomous system. Selecting **EXTERNAL** assumes that the inter-autonomous system routing is the major cost of packet routing.

5. **At Tag, if you want to filter the contents of the *External Route Tag* field, enter the field contents in eight-digit hexadecimal format.**

Within OSPF external links advertisements, a 32-bit *External Route Tag* field is attached to each route. The contents of this field are not used by OSPF but can be used by source and destination routers. If you do not want to filter field contents, leave **Tag** empty.

6. **At Preference (meaningful only when you set Action to ACCEPT), enter a weighted precedence value (a number from 1 to 10) to a route included in the routing pool.**

By default, routing preference is granted to OSPF routes, then to EGP routes, and then to RIP routes.

7. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

OSPF Import Route Filter stored.

NCU returns to the **IMPORT ROUTE FILTERS** window which now displays the OSPF import filter you just configured. Repeat this procedure to configure additional OSPF import filters.

5.5.8.2.2 Updating Import-Route Filters

You update import-route filters from the **IMPORT ROUTE FILTERS** window. First, select the import-route filter under **Network Address**. Next, select and then . NCU displays the appropriate window for the type of import-route filter you selected; if you selected:

- ❑ RIP import filter
NCU displays the **IMPORT FROM RIP** window, which displays the current parameter settings for the filter. See *Section 5.5.8.2.1.1, Adding RIP Import-Route Filters* for information how to change the parameters.
- ❑ EGP import filter
NCU displays the **IMPORT FROM EGP** window, which displays the current parameter settings for the filter. See *Section 5.5.8.2.1.2, Adding EGP Import-Route Filters* for information how to change the parameters.
- ❑ OSPF import filter
NCU displays the **IMPORT FROM OSPF** window, which displays the current parameter settings for the filter. See *Section 5.5.8.2.1.3, Adding OSPF Import-Route Filters* for information how to change the parameters.

5.5.8.2.3 Deleting Import-Route Filters

You delete import-route filters from the **IMPORT ROUTE FILTERS** window. First, select the import-route filter under **Network Address**. Next, select and then . NCU deletes the filter.

5.5.8.2.4 Adding Export-Route Filters

You add export-route filters from the **EXPORT ROUTE FILTERS** window. To display this window, select and in the **NODE IP CONFIGURATION** window to display the following sub-menu:

Source/Dest. Address
Import Route
Export Route

Select to display the **EXPORT ROUTE FILTERS** window (see Figure 5-44). You may then add export-route filters. Select to display the sub-menu in Figure 5-44. Depending on the sub-menu option you select, refer to the appropriate section for information on how to add the particular export-route filter.

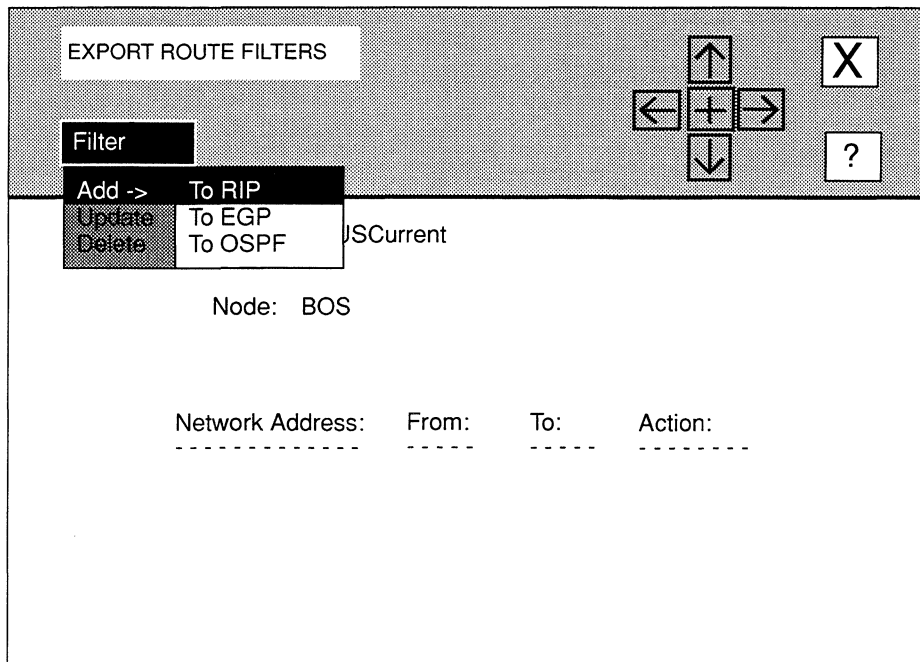


Figure 5-44. EXPORT ROUTE FILTERS Window

5.5.8.2.4.1 Adding a RIP Export Filter

You add a RIP export filter from the **EXPORT ROUTE TO RIP** window (see Figure 5-45), as follows:

1. **At Network Address, enter the filtered IP network address in dotted decimal notation.**

If you want to filter all destination networks, press **[RETURN]**. If you want to filter a specific IP network, enter the network address in dotted decimal notation, then press **[RETURN]**.

2. **At Network Mask, enter a range of addresses upon which the filter acts.**

For example, consider Class C Network 192.32.1.0, which allocates the upper 3 bits of the host identification field to Subnet_ID, and the final 5 bits to Host_ID.

EXPORT ROUTE TO RIP

Configuration: NEUSCurrent

Node: BOS

Network Address:

Network Mask:

To Protocol: RIP

From Protocol: EGP

Action: PROPAGATE

To Interface:

Metric:

Figure 5-45. EXPORT ROUTE TO RIP Window

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.x is subject to filtering. If you enter **255.255.255.0** at **Network Mask**, only the **Net_ID** portion of the address will be filtered. If you enter **255.255.255.224** at **Network Mask**, the **Net_ID** and **Subnet_ID** portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Network Mask**, the entire IP address will be filtered.

3. **Do nothing at To Protocol; NCU supplies the RIP protocol.**
4. **At From Protocol, select the source protocol (EGP or OSPF) of the routing information.**
5. **At Action, select how route is transferred to RIP.**
 - PROPAGATE** Specifies that the route is advertised to RIP.
 - IGNORE** Specifies that the route is not advertised to RIP.

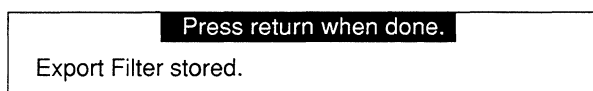
6. **At To Interface, enter the IP address (in dotted decimal notation) of an interface across which the filter operates.**

If you wish the RIP export route filter to be “universal” (applicable to all local interfaces), leave **To Interface** blank.

7. **At Metric (meaningful only when you set Action to PROPAGATE), enter a RIP cost to the route (keep in mind the diameter of the RIP network).**

8. **Select and then .**

NCU displays the following window; press [RETURN] to clear it from the screen:



NCU returns to the **EXPORT ROUTE FILTERS** window which now displays the RIP export filter you just configured. Repeat this procedure to configure additional RIP export filters.

5.5.8.2.4.2 Adding an EGP Export Filter

You add an EGP export filter from the **EXPORT ROUTE TO EGP** window (see Figure 5-46), as follows:

1. **At Network Address, enter the filtered IP network address in dotted decimal notation.**

If you want to filter all destination networks, leave **Network Address** blank.

2. **At Network Mask, enter a range of addresses upon which the filter acts.**

For example, consider Class C Network 192.32.1.0, which allocates the upper 3 bits of the host identification field to **Subnet_ID**, and the final 5 bits to **Host_ID**.

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.x is subject to filtering. If you enter **255.255.255.0** at **Network Mask**, only the **Net_ID** portion of the address will be filtered. If you enter **255.255.255.224** at **Network Mask**, the **Net_ID** and **Subnet_ID** portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Network Mask**, the entire IP address will be filtered.

3. **Do nothing at To Protocol, NCU automatically sets this parameter to EGP.**
4. **At From Protocol, select the source protocol (RIP or OSPF) of the routing information.**

EXPORT ROUTE TO RIP

↑
← + →
↓
X
?

Configuration: NEUSCurrent

Node: BOS

Network Address:

Network Mask:

To Protocol: EGP

From Protocol: RIP

Action: PROPAGATE

Peer:

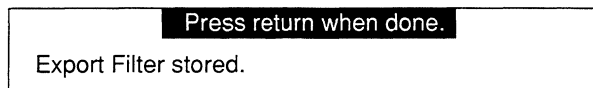
Metric:

Figure 5-46. EXPORT ROUTE TO EGP Window

5. **At Action, select how the route is transferred to EGP.**
PROPAGATE Specifies that the route is advertised to EGP.
IGNORE Specifies that the route is not advertised to EGP.
6. **At Peer, enter the IP address (in dotted decimal notation) of an EGP router to which the EGP export route filter applies.**
 If you wish the EGP export route filter to be “universal” (applicable to all foreign EGP routers), leave **Peer** blank.
7. **At Metric (meaningful only if you set Action to PROPAGATE), enter an EGP cost to the route.**

8. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **EXPORT ROUTE FILTERS** window which now displays the EGP export filter you just configured. Repeat this procedure to configure additional EGP export filters.

5.5.8.2.4.3 Adding an OSPF Export Filter

You add an OSPF export filter from the **EXPORT ROUTE TO OSPF** window (see Figure 5-47), as follows:

1. **At Network Address, enter the filtered IP network address in dotted decimal notation.**

If you want to filter all destination networks, leave **Network Address** empty.

2. **At Network Mask, enter a range of addresses upon which the filter acts.**

For example, consider Class C Network 192.32.1.0, which allocates the upper 3 bits of the host identification field to **Subnet_ID**, and the final 5 bits to **Host_ID**.

The address mask directs the filtering process to a specific portion of the IP address. In other words, any IP address that matches the masked portion of 192.32.1.x is subject to filtering. If you enter **255.255.255.0** at **Network Mask**, only the **Net_ID** portion of the address will be filtered. If you enter **255.255.255.224** at **Network Mask**, the **Net_ID** and **Subnet_ID** portions of the address will be filtered. Finally, if you enter **255.255.255.255** at **Network Mask**, the entire IP address will be filtered.

3. **At From Protocol, select the source protocol (EGP or OSPF) of the routing information.**

4. **At Action, select how the route is transferred to OSPF.**

PROPAGATE Specifies that the route is advertised to OSPF.

IGNORE Specifies that the route is not advertised to OSPF.

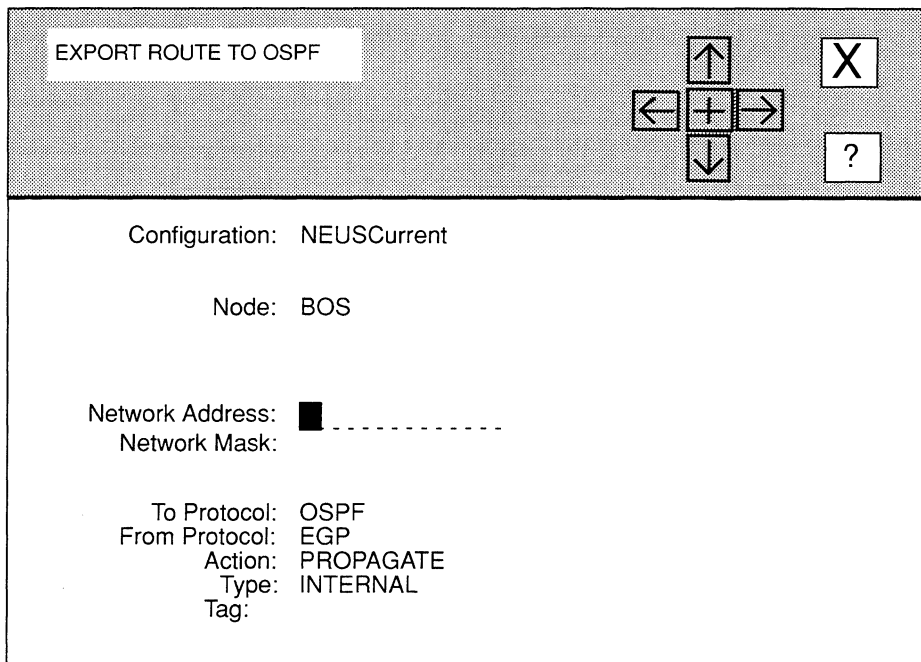


Figure 5-47. EXPORT ROUTE TO OSPF Window

- At **Type**, select the OSPF external-metrics type to which you want the OSPF export-route filter to apply.

INTERNAL..... Specifies Type 1 metrics, which are equivalent to the standard OSPF link-state metric.

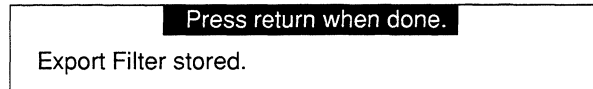
EXTERNAL..... Specifies Type 2 metrics, which are greater than the cost of any path internal to the autonomous system. Selecting **EXTERNAL** assumes that the inter-autonomous system routing is the major cost of packet routing.

- At **Tag**, if you want to filter the contents of the *External Route Tag* field, enter the field contents in eight-digit hexadecimal format.

Within OSPF external links advertisements, a 32-bit *External Route Tag* field is attached to each route. The contents of this field are not used by OSPF but can be used by source and destination routers. If you do not want to filter field contents, leave **Tag** empty.

7. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **EXPORT ROUTE FILTERS** window which now displays the OSPF export filter you just configured. Repeat this procedure to configure additional OSPF export filters.

5.5.8.2.5 Updating Export-Route Filters

You update export-route filters from the **EXPORT ROUTE FILTERS** window. First, select the export-route filter under **Network Address**. Next, select and then . NCU displays the appropriate window for the type of export-route filter you selected; if you selected:

- RIP export filter

NCU displays the **EXPORT ROUTE TO RIP** window, which displays the current parameter settings for the filter. See *Section 5.5.8.2.4.1, Adding RIP Export-Route Filters* for information how to change the parameters.

- EGP export filter

NCU displays the **EXPORT ROUTE TO EGP** window, which displays the current parameter settings for the filter. See *Section 5.5.8.2.4.2, Adding EGP Export-Route Filters* for information how to change the parameters.

- OSPF export filter

NCU displays the **EXPORT ROUTE TO OSPF** window, which displays the current parameter settings for the filter. See *Section 5.5.8.2.4.3, Adding OSPF Export-Route Filters* for information how to change the parameters.

5.5.8.2.6 Deleting Export-Route Filters

You delete export-route filters from the **EXPORT ROUTE FILTERS** window. First, select the export-route filter under **Network Address**. Next, select and then . NCU deletes the filter.

5.6 Configuring IP Applications

This section describes how to configure the following IP applications:

❑ TFTP

The Wellfleet node includes a client and server implementation of the Trivial File Transfer Protocol (TFTP), a protocol to transmit files across an internet. TFTP is implemented on top of UDP to read and write files from/to a remote device. It cannot list directories nor does it provide for user authentication.

❑ BOOTP

The Wellfleet node provides both a client and server implementation of the Bootstrap Protocol (BOOTP) as specified in RFC 951. The BOOTP client implementation enables the router to reboot itself over one of its network connections. The BOOTP server implementation enables the router to act as a bootserver for another router on a directly attached network.

With the BOOTP client enabled, the node broadcasts a BOOTP request packet over each of its interfaces, and awaits a reply from an adjacent BOOTP server. If no response is received within a time-out period, the router repeats the broadcast up to a specified number of times. Each repetition is spaced exponentially and randomized by the BOOTP client to avoid collisions.

Upon receipt of a BOOTP reply packet, the node invokes TFTP to obtain the image and configuration files identified with the reply packet. It then reboots with these new files.

If no BOOTP reply packet is received, the node aborts the network boot attempt and uses the currently loaded image and configuration files.

❑ SNMP agent

The Wellfleet node supports the Simple Network Management Protocol (SNMP) management agent software. RFCs 1155, 1156, and 1157 describe SNMP:

- RFC 1155 describes the structure and identification of management information for IP networks.
- RFC 1156 describes the standard Internet Management Information Base (MIB).
- RFC 1157 describes the Simple Network Management Information Protocol.

SNMP is a transaction-based protocol that specifies the transfer of structured management information between two types of SNMP entities: *applications* and *agents*.

Application software runs in a network management center; it issues queries to gather data about the status, configuration, and performance of external devices — or *network elements* (in SNMP terminology). Agent software, on the other hand, runs in network elements (for example, a Wellfleet node). Agent software responds to monitoring center queries, and, if so configured, generates unsolicited reports of significant activity (referred to as *traps*) back to the monitoring center.

5.6.1 Configuring TFTP

You configure TFTP, as follows:

1. Select and then in the **NODE IP CONFIGURATION** window.

NCU displays the **TFTP** window (see Figure 5-48).

2. At **Auto Enable**, select the state of **TFTP** when the node boots.

This TFTP-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable TFTP when the node boots, as follows:

- When global **Auto Enable** is **set to NO**, TFTP (and every other application software module) is unconditionally disabled.

You will subsequently need to enable TFTP manually with the NCL Interpreter after the node boots.

- When global **Auto Enable** is set to **YES**, TFTP (and every other application software module) is conditionally enabled.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable TFTP.
- Select **NO** to disable TFTP (you will subsequently need to enable the TFTP manually with the NCL Interpreter after the node boots).

Note

Because TFTP allows write access to the node's diskette, it is recommended that TFTP *not* be configured to auto enable in environments where security a concern.

3. At **Max Retransmissions** specify the number of times TFTP retransmits an unacknowledged data message before abandoning the transfer attempt.

TFTP

Configuration: NEUSCurrent

Node: BOS

Auto Enable: NO

Maximum Retransmission: 5
Retransmission Timeout: 5
Connection Close Timeout: 25

Figure 5-48. TFTP Window

4. At **Retransmission Time Out** specify the number of seconds TFTP waits for an acknowledgment before retransmitting a data message.
5. At **Connection Close Time-out** specify the number of seconds TFTP waits before relinquishing resources after it has successfully completed a file transfer.
6. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.
TFTP Parameters stored.

5.6.2 Configuring the SNMP Agent

Note

As all SNMP transmissions between application and agent entities are conveyed via the connectionless *User Datagram Protocol (UDP)*, you must load the TCP/IP Router application software to Slot 2 (the master slot) of the Wellfleet node to enable SNMP operations.

When you configure the SNMP agent, you define SNMP communities and their members. Select and then in the **NODE IP CONFIGURATION** window. NCU displays the **SNMP** window (see Figure 5-49). This window allows you to add, update, and delete SNMP communities.

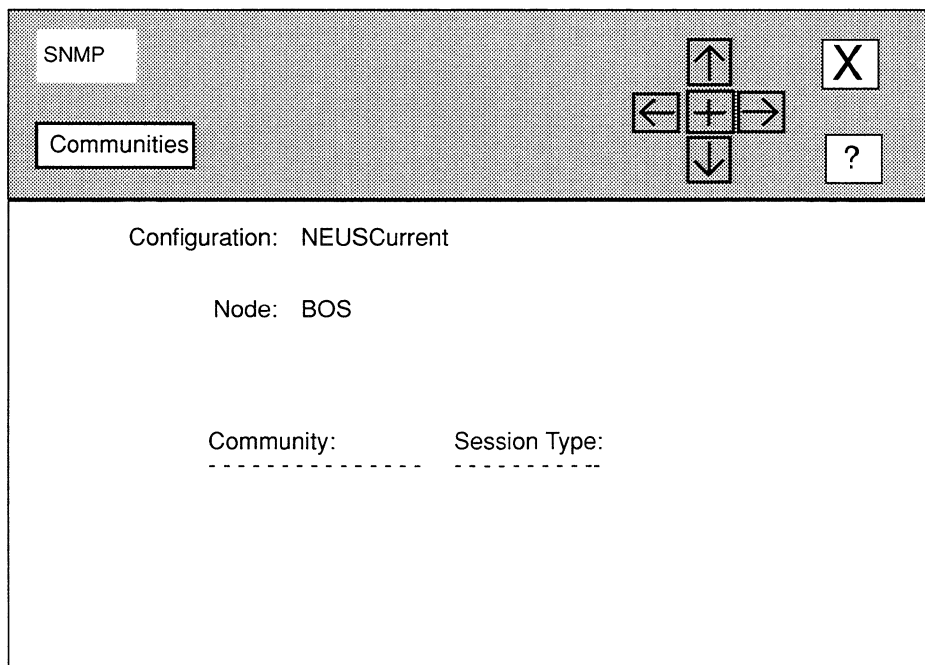


Figure 5-49. SNMP Window

5.6.2.1 Adding SNMP Communities

You add SNMP communities from the **SNMP** window, as follows:

1. Select and then .

NCU displays the **SNMP COMMUNITY** window (see Figure 5-50).

2. **At Community Name, enter the SNMP community name for the network monitoring centers authorized to query the node.**

An SNMP community is a group of monitoring centers authorized to issue queries to the SNMP agent. A community has a *name* which identifies a logical set of application entities, and *members*, which are the IP addresses of management stations authorized to query the node's resident agent software.

As you define communities, the SNMP application software ensures that your entry matches an existing community previously defined by the SNMP application software.

In addition to a name and members, a community also has a *mode* which specifies what type of remote access commands can be carried out by application entities and a *type* which specifies how application entities gather management data.

3. **At Session Mode, select remote access privilege to the local MIB.**

READ Specifies read access. The current SNMP agent implementation supports only read access. You must select **READ**.

READ/WRITE The current SNMP agent implementation supports only read access. Do not select **READ/WRITE**.

4. **At Session Type, select the data-exchange model between the SNMP application and agent entities.**

REGULAR Specifies a query/response model in which agent output is triggered by the receipt of application requests.

TRAP Specifies a data-exchange model in which the local agent not only responds to application requests, but also generates asynchronous, unsolicited notifications of significant local events (as defined by RFC 1157).

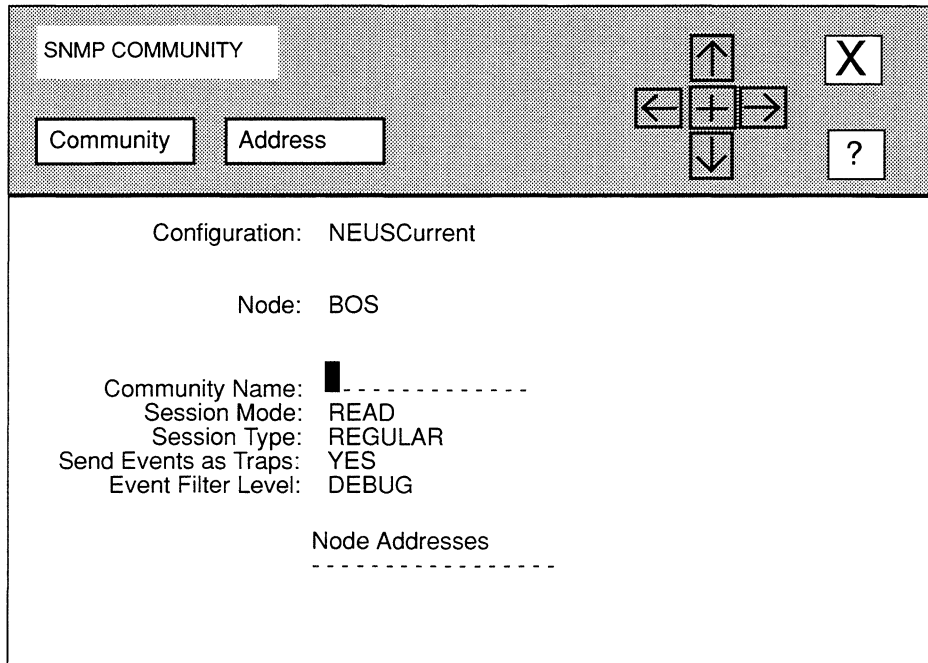


Figure 5-50. SNMP COMMUNITY Window

5. At Send Events As Traps, enable or disable enterprise-specific trap generation.

YES Enables enterprise-specific trap generation in which some or all of the event messages generated by the node are encapsulated within an SNMP protocol data unit and sent as traps to SNMP application entities.

NO Disables enterprise-specific trap generation.

If you set **Enable Logging** to **YES** when you configured the global parameters, the node writes event messages (about network service and performance changes, and anomalous events) to a file on the system diskette. These event messages have five levels of severity, as follows:

Severity Level	Indicates
Major	A service has appeared or disappeared.
Warning	A service has behaved unexpectedly.
Performance	A service has upgraded/degraded.
Information	General system information.
Debug	Installation/diagnostic information.

6. **At Event Filter Level, if you set Send Events as Traps to YES, select which event messages are transmitted as traps to SNMP application entities; if you set Send Events as Traps to NO, do nothing.**

NCU provides six responses to **Event Filter Level**:

DEBUG Specifies that the node displays messages on the console screen with these severity levels: Major, Warning, Performance, Information, and Debug.

SHOW ALL EVENTS Specifies that the node displays messages on the console screen with these severity levels: Major, Warning, Performance, and Information.

NOT INFO..... Specifies that the node displays messages on the console screen with these severity levels: Major, Warning, and Performance.

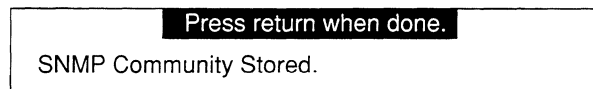
PERF AND MAJOR Specifies that the node displays messages on the console screen with these severity levels: Major and Performance.

JUST MAJOR Specifies that the node displays messages on the console screen with Major severity levels.

DROP ALL..... Specifies that the node displays no event messages.

7. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.



After you add the SNMP community, you specify community members; go to *Section 5.6.2.1.1, Adding SNMP Community Members*.

5.6.2.1.1 Adding SNMP Community Members

Once you have added an SNMP community, you may define its members, as follows:

1. Select and then in the **SNMP COMMUNITY WINDOW**.

NCU displays the **SNMP ACCESS** window (see Figure 5-51).

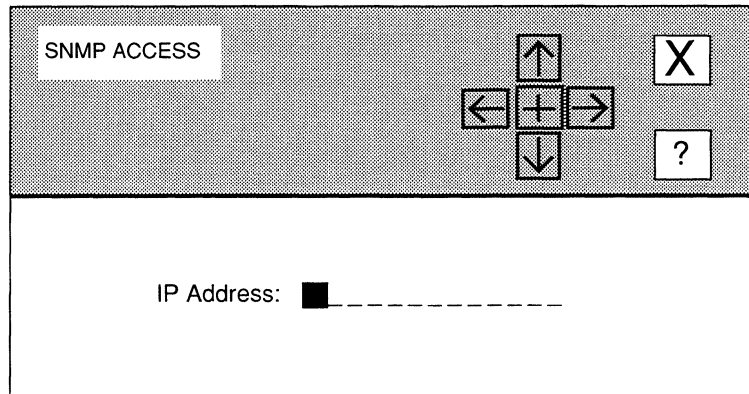
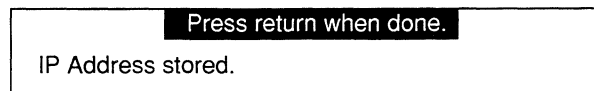


Figure 5-51. SNMP ACCESS Window

2. At **IP Address**, enter the dotted decimal address of a community member (a specific host device) granted access to the local MIB.
3. Select and then .
NCU displays the following window; press **[RETURN]** to clear it from the screen.



NCU returns to the **SNMP COMMUNITY** window which now displays the SNMP community member you just added under **Node Addresses**.

5.6.2.1.2 Deleting SNMP Community Members

You delete SNMP community members from the **SNMP COMMUNITY** window. First, select the SNMP community member under **Node Addresses**. Next, select and then . NCU deletes the SNMP community member.

5.6.2.2 Updating SNMP Communities

You update SNMP communities from the **SNMP** window, as follows:

1. Select the community you wish to update under **Community**, and then select and .

NCU displays the **SNMP COMMUNITY** window for that community.

2. Follow steps 1 through 6 of *Section 5.6.2.1, Adding SNMP Communities* to reset the desired parameters.
3. Add and delete SNMP community members as you see fit (see *Section 5.6.2.1.1, Adding SNMP Community Members*, and *Section 5.6.2.1.2, Deleting SNMP Community Members*).
4. Once you have updated the SNMP community with the desired information, select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

SNMP Community updated.

5. Select and then to return to the SNMP window.

5.6.2.3 Deleting SNMP Communities

You delete SNMP communities from the **SNMP** window. First, select the SNMP community you wish to delete under **Community**. Next, select and . NCU deletes the SNMP community.

5.6.3 Configuring BOOTP

Note

Both BOOTP client and server operations require that you configure and enable the IP Router, TFTP, and BOOTP. Proper operation of the BOOTP client implementation requires that a BOOTP server reside on a directly attached network, and that you configure the server to load the appropriate software image and, optionally, configuration file to the client.

The following sections describe how to add the BOOTP server and client to a node, as well as how to update and delete the BOOTP server.

5.6.3.1 Adding the BOOTP Server

You add the BOOTP server, as follows:

1. Select and then in the **NODE IP CONFIGURATION** window.

NCU displays the **BOOTP** window (see Figure 5-52).

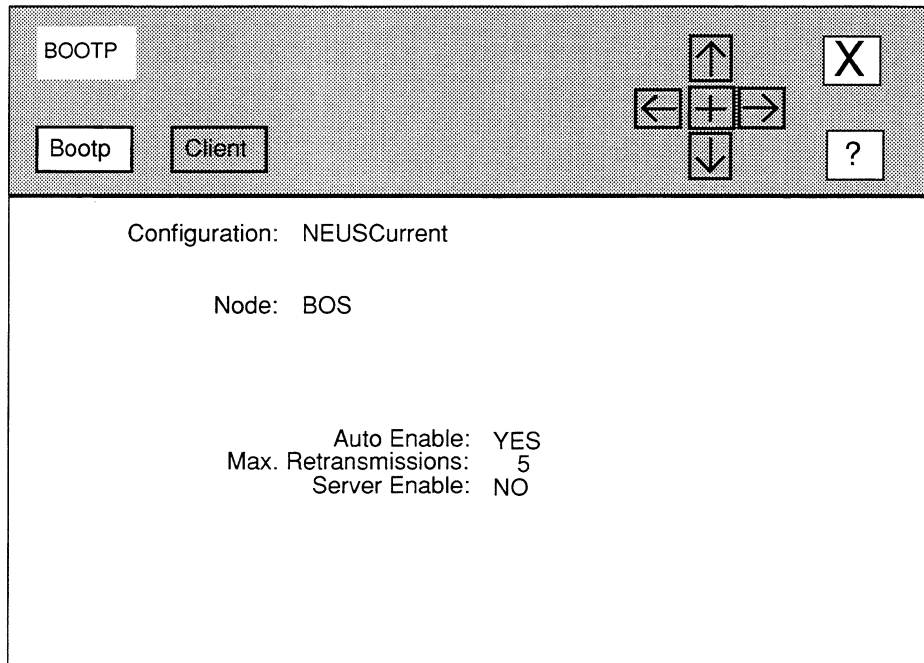


Figure 5-52. BOOTP Window

2. At Auto Enable, enable or disable the BOOTP client.

YES Enables the BOOTP client.

NO Disables the BOOTP client.

3. At Max Retransmissions, enter the number of times that the BOOTP client retransmits a BOOTP request packet.

4. At Server Enable, enable or disable the BOOTP server.

YES Enables the BOOTP server. With the BOOTP server enabled, the Wellfleet node functions as a bootserver for specified routers on directly attached networks. It listens for BOOTP request packets from known routers on well-known UDP port 67.

Upon reception of a request packet, the node extracts the client's hardware and IP address from the packet and attempts to match them against an entry in its BOOTP server database. If the lookup fails, the node simply drops

the request; it issues no BOOTP reply packet. If the lookup succeeds, the node constructs a BOOTP reply packet which contains the IP address of a BOOTP server, the name of a boot file, and (optionally) the name of a configuration file to be used by the client (Section 5.6.3.1.1, *Adding the BOOTP Client* describes how to add the BOOTP client).

Note

“*Booting through gateways*” isn’t supported. The BOOTP server doesn’t broadcast request packets over its other interfaces; it responds to the request packet directly, or not at all.

NO Disables the BOOTP server.

5. Select and then .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

Press return when done.

BOOTP Parameters Stored.

If you have enabled the BOOTP server, go to Section 5.6.3.1.1, *Adding the BOOTP Client*.

5.6.3.1.1 Adding the BOOTP Client

Once you have enabled the BOOTP server, you can add BOOTP client information from the **BOOTP** window, as follows:

1. Select and then .

NCU displays the **BOOTP CLIENT** window (see Figure 5-53).

2. **At Client IP Address, enter the IP address (in dotted decimal notation) of a of a BOOTP-client router on a directly attached network, for which the IP router provides BOOTP service.**
3. **At Client LAN Address, enter the 49-bit physical address of the device identified by Client IP Address.**
4. **At File Server IP Address, enter the IP address (in dotted decimal notation) of the device that supplies the boot file.**

Note

The source of the boot file need not be the router itself; the router can return the address of another network device (for example, a SUN workstation) which is configured to supply the named files upon request.

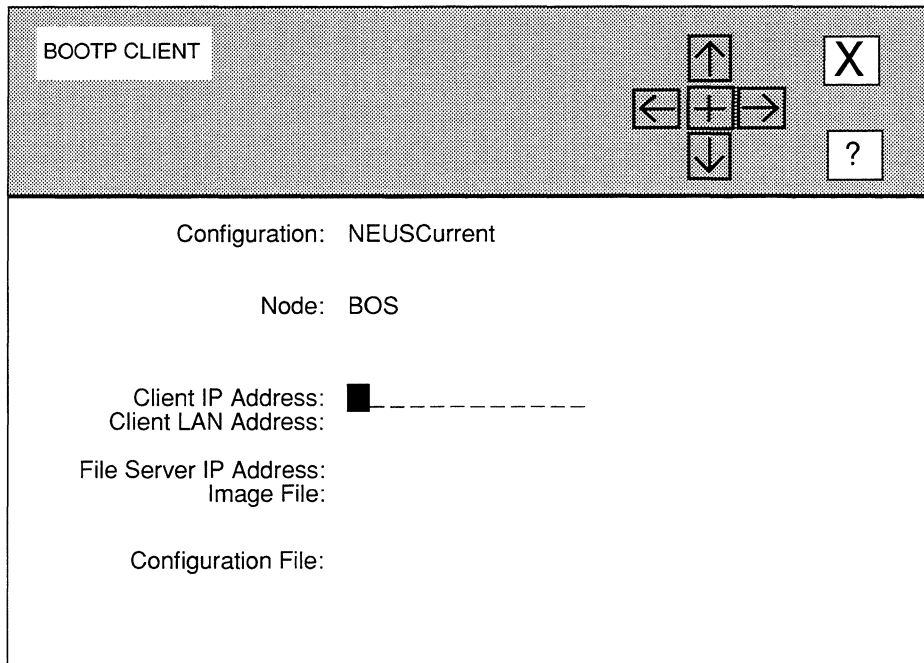


Figure 5-53. BOOTP CLIENT Window

5. At Image File, enter the name of the boot file.

If the router supplies the boot image, enter the name of the file on the local disk. If another device supplies the boot image, enter the full pathname to the file (for example, `/usr3/wf/wf_exec/5.60/cat.out`). Path names are restricted to 60 characters in length.

Note

If a network device, not the router, supplies the boot image, Wellfleet recommends you store the file under a file name other than `ace.out`.

6. At Configuration File, enter the name of the configuration file.

If you do not wish to specify a configuration file, simply ignore this field; the Wellfleet BOOTP server implementation makes use of the vendor-specific field of the BOOTP reply packet (numbered tag 129) to convey the name of a configuration file.

If the router supplies the configuration file, enter the name of the file on the local disk. If another device supplies the configuration file, enter the full pathname to the file (for example, `/usr3/wf/wf_cfg/5.60/cfg_2`). Path names are restricted to 60 characters in length.

Note

If the configuration file is supplied by a device other than the router, it cannot be stored under the file name `config`.

7. Select and then Save .

NCU displays the following window; press **[RETURN]** to clear it from the screen.

```

Press return when done.
BOOTP Client stored.

```

NCU returns to the **BOOTP** window which now displays the BOOTP client information under **Client Address** and **File Server Address**.

5.6.3.1.2 Updating the BOOTP Client

You update a BOOTP client from the **BOOTP** window. First, select the client under **Client Address**. Next, select Client and then Update . NCU displays the **BOOTP CLIENT** window, which displays the current parameters for the BOOTP client. See *Section 5.6.3.1.1, Adding the BOOTP Client* for information on how to reset the parameters.

5.6.3.1.3 Deleting the BOOTP Client

You delete a BOOTP client from the **BOOTP** window. First, select the client under **Client Address**. Next, select Client and then Delete . NCU deletes the BOOTP client.

5.6.3.2 Updating the BOOTP Server

You update the BOOTP server from the **BOOTP** window. Simply reset the parameters in the window (see steps 2 through 4 of *Section 5.6.3.1, Adding the BOOTP Server* for information on how to reset the parameters). Next, update the BOOTP client as you see fit (see *Section 5.6.3.1.2, Updating the BOOTP Client*). Finally, select Bootp and Update in the **BOOTP** window. NCU displays the following window; press **[RETURN]** to clear it from the console:

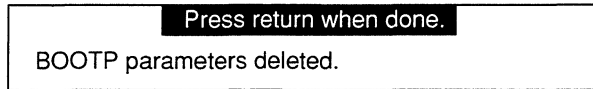
```

Press return when done.
BOOTP parameters stored.

```

5.6.3.3 Deleting the BOOTP Server

You delete the BOOTP server from the **BOOTP** window. Simply, select and then . NCU displays the following window; press **[RETURN]** to clear it from the console:



5.7 Configuring TCP

The Transmission Control Protocol (TCP) is the Internet standard connection-mode, transport-level protocol. You need not assign values to TCP parameters while configuring a node. Accepting default TCP parameters ensures operational efficiency.

To display the TCP parameters, select in the **NODE IP CONFIGURATION** window. NCU displays the TCP window (see Figure 5-54). To restore default settings, select .

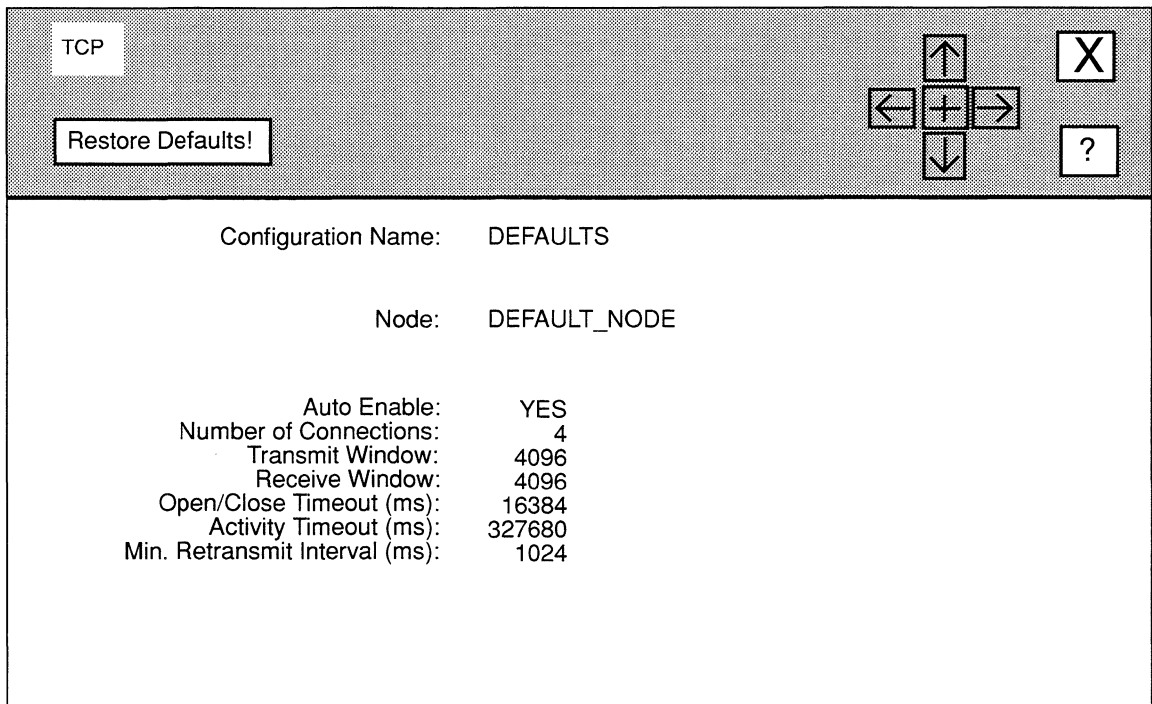


Figure 5-54. TCP Window

6 Editing Bridge Parameters

Bridge parameters consist of:

- ❑ Basic parameters
Basic parameters apply to the entire bridge software module for both transparent and source-routing bridging services.
- ❑ Interface parameters
Interface parameters consist of certain source-routing and spanning-tree specific parameters, as well as filtering, and load-balance parameters. You must configure these parameters individually for each bridge interface.

This chapter describes how to access and edit these parameters. The first section provides an overview of the bridge.

6.1 Bridge Overview

Bridges are data-link layer relay devices that use Media Access Control (MAC) source and destination addresses to filter and relay frames between network and/or point-to-point connections. There are two general types of bridges: *transparent* bridges and *source-routing bridges* (the Wellfleet Source-Routing Transparent, SRT, Bridge provides concurrent transparent and source-routing bridging services).

The following sections describe:

- ❑ Bridge services
How transparent services differ from source-routing services, and how the SRT bridge provides concurrent transparent and source-routing services.
- ❑ Spanning-tree algorithm
This algorithm provides a loop-free topology in networks that contain parallel bridges.
- ❑ Filtering mechanisms
Filtering allows bridges to either selectively relay or selectively drop certain frames.

6.1.1 Transparent Bridges

Transparent bridges provide network interconnection and/or extension services to LANs that employ identical protocols at the data-link and physical layers. Transparent bridges place no burden on hosts — hosts take no part in the route discovery or selection process. From a host's point of view, it appears that all hosts reside on a single extended network with each node identified by a unique MAC-level address.

Basically, a transparent bridge provides a relatively simple relay function, as follows:

- ❑ Learns the addresses of end-stations on connected networks.
- ❑ Relays frames based on its acquired knowledge of end-station addresses.
- ❑ Ensures (if the spanning-tree algorithm is enabled) a loop-free topology throughout the extended network.

The transparent bridge learns end-station addresses by observing the source address of each frame it receives. As it receives frames, the transparent bridge builds and updates a database (called the *forwarding table*) that lists each source address, the circuit group on which the bridge observed the address, and a timer value that indicates the age of the observation.

The transparent bridge relays frames based on forwarding-table entries. When it receives a frame, the bridge compares the frame's destination address with addresses in the forwarding table:

- ❑ If the bridge does not find a match between the destination address and a forwarding-table entry, it relays the frame on all circuit groups — except the circuit group on which it received the frame.

Relaying a frame on multiple circuit groups is called *flooding*.

- ❑ If the bridge finds a match between the destination address and a forwarding-table entry, it compares the circuit group on which it received the frame with the circuit group associated with the table entry:

- Identical circuit groups indicate that the source and destination end-stations are on the same physical network.

Because relay is not necessary in this instance, the bridge drops the frame.

- Different circuit groups indicate that the source and destination end-stations are *not* on the same physical network.

In this instance, the bridge relays the frame on the circuit group in the forwarding table.

With the spanning-tree algorithm enabled, the transparent bridge ensures a loop-free topology. The algorithm provides a single path between any two end-stations.

6.1.2 Source-Routing Bridges

IBM coined the term *source routing* to describe a method of bridging frames across token-ring networks. Source-routing bridges differ from transparent bridges in two critical ways:

- ❑ *Source-routing bridges* tolerate multiple paths between any two nodes on the extended network; *transparent bridges* require a loop-free topology.
- ❑ *Source-routing bridges* require hosts to supply the information needed to deliver a frame to its intended recipient; *transparent bridges* place no burden on hosts.

Source-routing bridges do not use forwarding tables; rather, they decide to forward or drop a frame based solely on data inside the frame. In a source-routing extended network, each source node determines the route to a destination node through a process labeled route discovery.

Four types of routing directives enable the route-discovery process:

- ❑ *All Paths Broadcast Routing*

Generates multiple frames, called *all-paths explorer* (APE) frames, that traverse all paths between source and destination stations. When a source-routing bridge receives an APE frame, it appends a *routing designator*; an information triplet which takes the following form:

[LAN_i] [Bridge_ID] [LAN_j]

where:

LAN_i

Is a unique number that identifies the LAN from which the frame arrived.

Bridge_ID

Is a number that identifies the intervening bridge.

LAN_j

Is a unique number that identifies the LAN upon which the bridge relays the APE frame.

After adding a routing designator, the bridge forwards the frame onto all ports, except the port on which it received the frame. Consequently, multiple copies of the same APE frame can appear on a LAN, and the frame recipient can receive multiple copies of the frame (one copy for each possible path through the extended network). Each APE frame that the recipient receives contains a unique sequenced list of routing designators tracing the frame's path through the extended network.

- ❑ *Spanning Tree Broadcast Routing*

Generates a single frame, called a *transparent-spanning frame* (TSF), which follows a loop-free (spanning-tree-derived) path from source node to destination node. When each bridge on the spanning tree receives a TSF, it forwards the frame onto all active (non-blocked) ports — except the port on which it received the frame. With spanning-tree-broadcast routing, one copy of the TSF appears on each LAN, and the frame recipient receives only a single copy of the frame.
- ❑ *Specific Routing*

Generates a single frame, called a *specifically-routed frame* (SRF), which traverses a specific path designated by the source node. SRFs contain a list of routing designators that map a unique path through the extended network from source to destination node. When a bridge receives an SRF, it examines the list of routing designators: if the bridge is on the specified path, it forwards the SRF; if the bridge is not on the specified path, it ignores the frame.
- ❑ *Null Routing*

Indicates that the source node does not desire any routing services from network bridges. As a result, null-routed frames are restricted to the resident LAN of the originating node.

6.1.2.1 How Source Routing Works

Source-routing networks consist of LAN segments interconnected by source-routing bridges. Each LAN segment has a unique network-wide identification number. Each source-routing bridge has an identification number; the Wellfleet source-routing bridge is always (by default) bridge number 1. Additionally, each Wellfleet source-routing bridge, has a unique network-wide internal LAN identification number.

As a source-routed frame traverses the network, it collects a sequence of routing designators that track its path through the network. Each source-routing bridge that the frame passes through inserts routing designators in the frame's MAC header. Transparent bridges do not write to the MAC header.

Each routing designator pairs a LAN segment number with a bridge number in order to identify a portion of the frame's path through the bridge. For example, in Figure 6-1, SRT **Bridge_B** receives a source routed frame on LAN Segment **5** and relays the frame on LAN segment **8**. Consequently, **Bridge_B** adds three routing designators (**5—1, A—1, 0—0**) to the frame's MAC header. Figure 6-1 illustrates how these routing designators map the frame's path through **Bridge_B**.

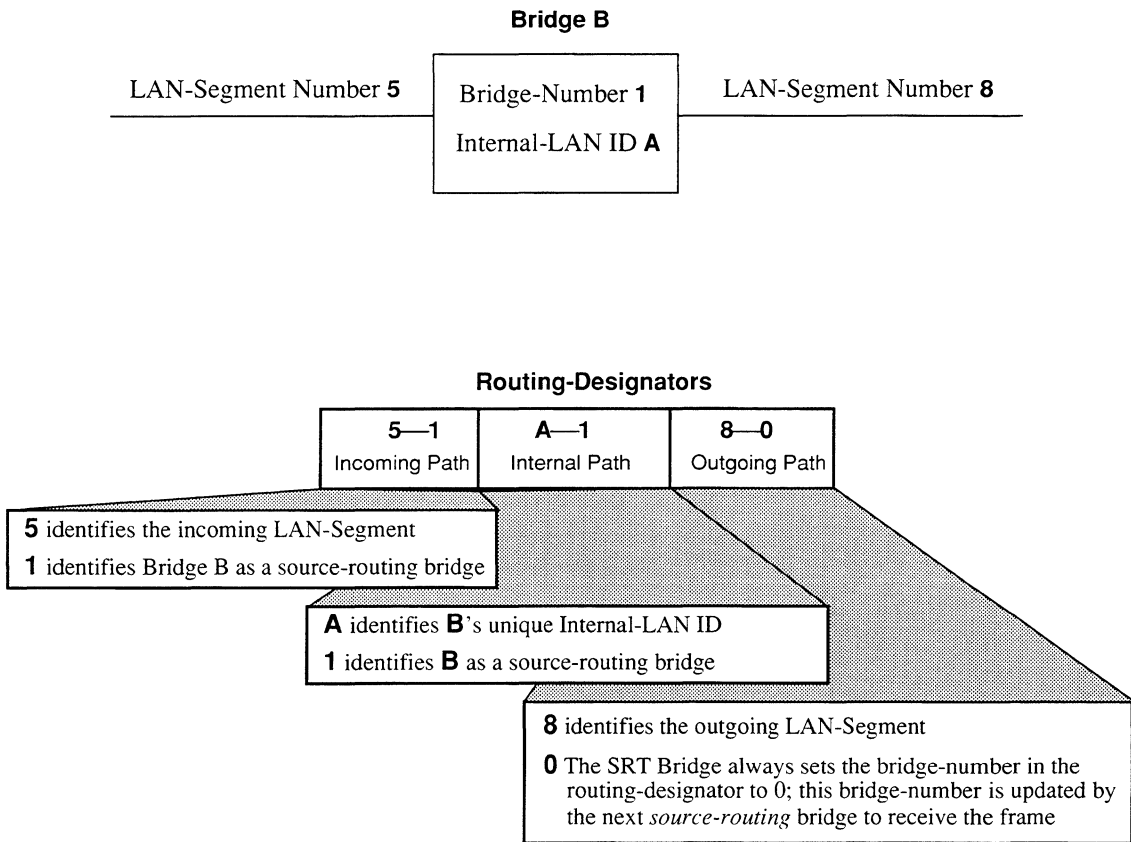


Figure 6-1. SRT Bridge Routing Designators

When a source-routed frame reaches its destination, its MAC header contains the route (identified by a sequence of routing designators) that the frame traversed to get there. For example, Figure 6-2 depicts a multi-ring network with three Wellfleet SRT bridges (bridges **A**, **B**, and **C**) and one transparent bridge (bridge **D**). In the figure, node **H_1** wants to use source routing to exchange frames with node **H_2**.

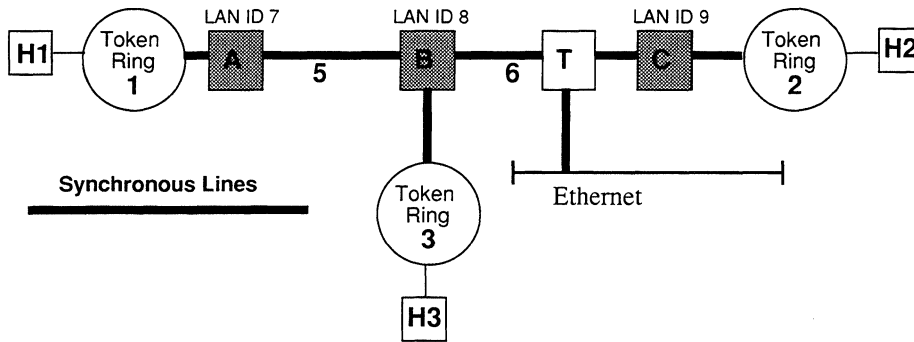


Figure 6-2. Multi-Ring Source-Routed Network

To begin the process, **H_1** transmits an APE frame. As the APE frame crosses the network, every source-routing bridge that it passes through inserts routing designators in the frame's MAC header. If the frame crosses a transparent bridge, the bridge simply forwards the frame based on its MAC destination address.

In Figure 6-2, Bridge **A** is the first source-routing bridge to receive the APE frame from **H_1**. Bridge **A** inserts its routing-designators (**1 — 1, 7 — 1, 5 — 0**) in the APE frame's MAC header (**APE Frame 1—1,7—1,5—0**). Figures 6-3, 6-4, and 6-5 depict how these routing-designators identify the frame's path through the bridge. Note that in Figure 6-3, Bridge **A** inserts the incoming LAN-segment/bridge pairing (**1-1**) only because it is the first source-routing bridge to receive the frame. If **A** was not the first source-routing bridge to receive this frame, the incoming LAN-segment/bridge pairing would be taken from the outgoing LAN-segment/bridge pairing inserted by the previous source-routing bridge.

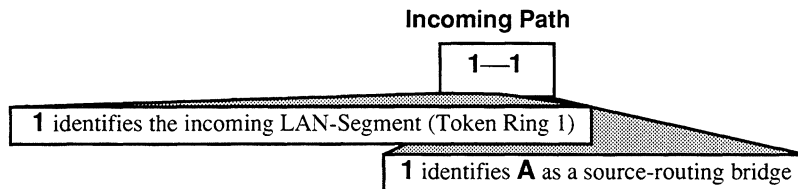


Figure 6-3. Routing Designator 1

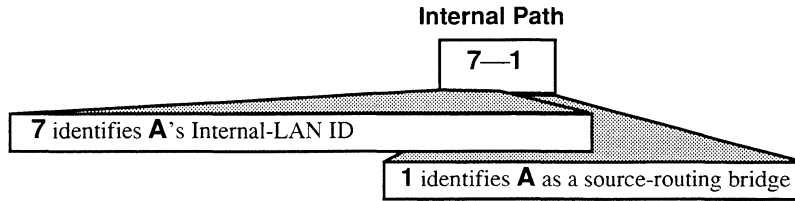


Figure 6-4. Routing Designator 2

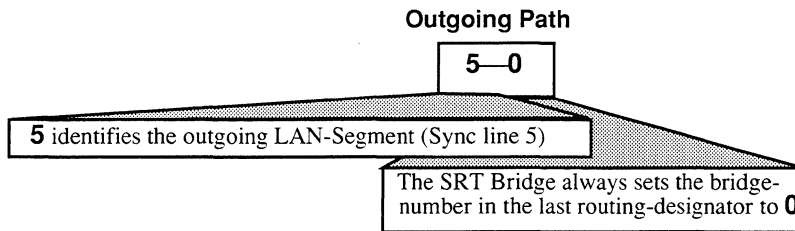


Figure 6-5. Routing Designator 3

Bridge **B** is the next source-routing bridge to receive the frame. **B** update **A**'s last routing designator by changing bridge number **0** to **1**; then **B** inserts the remainder of its routing designators (**8—1, 6—0**) in the frame's MAC header. **B** then floods APE frames onto LAN segments **3** and **6** with updated routing designators in their MAC headers:

- ❑ **B** transmits this frame (APE Frame **1—1, 7—1, 5—1, B—1, 3—0**) to LAN segment **3**, where it is eventually dropped since it does not provide a path to **H_2**.
- ❑ **B** transmits this frame (APE Frame **1—1, 7—1, 5—1, B—1, 6—0**) to LAN segment **6**; this frame passes through transparent bridge **T**.

Because **T** is a transparent bridge, it does not recognize the routing designators in the source-routed frame. **T** treats the APE frame as it would a transparent-bridging frame, and simply forwards the frame to Bridge **C** and the Ethernet (the frame transmitted onto the Ethernet is discarded).

Bridge **C** is the last source-routing bridge to receive the APE frame. **C** updates **B**'s last routing designator by changing bridge number **0** to **1**, and then inserts the remainder of its routing designators (**9—1, 2—0**) in the frame's MAC header. **C** then transmits the APE frame to the destination node **H_2**.

H_2 inspects the frame's MAC header to learn the route that this particular frame traversed. The route is mapped out by the sequence of routing designators (1 — 1, 7 — 1, 5 — 1, 8 — 1, 6 — 1, 9 — 1, 2 — 0) inserted by source-routing bridges **A**, **B**, and **C**.

Note

Other route-discovery protocols are available and used extensively. All such protocols, however, exchange TSFs, APE frames, and SRFs as described in this section.

6.1.3 Source-Routing/Transparent Bridges

Wellfleet's Source-Routing/Transparent (SRT) bridge provides concurrent transparent and source-routing services. In Figure 6-6, four Wellfleet SRT bridges link a multi-ring, multi-Ethernet extended network linked: Bridge **T** provides only transparent-bridging services; the three other bridges (all labelled **S**) provide both source-routing and transparent services.

Note

With source routing enabled, Wellfleet's SRT Bridge provides both source routing and transparent bridging.

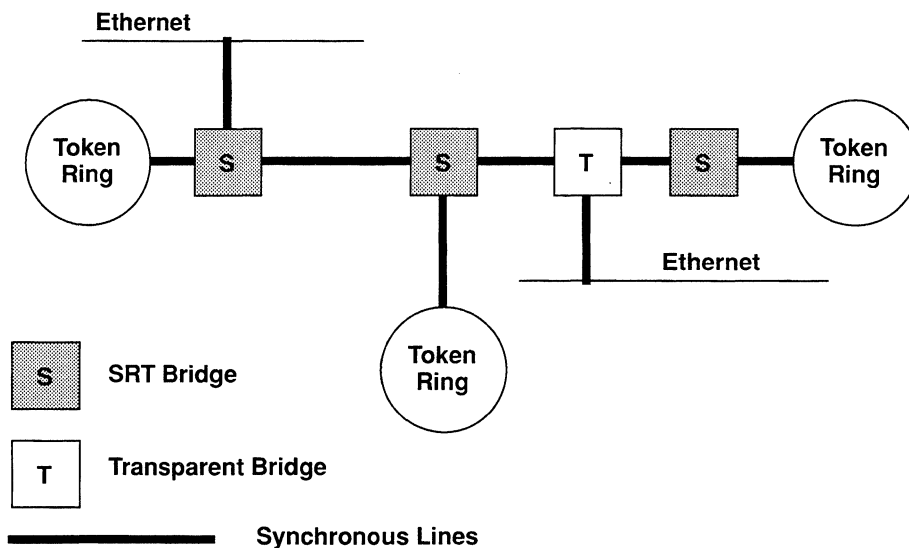


Figure 6-6. Sample SRT Topology

In the figure, Bridge **T** treats all frames as transparent-bridging frames. However, to effect route discovery, the SRT bridges (those bridges labelled **S**) inspect the most-significant bit of the frame's source address (the *routing-information indicator*, or RII) in order to separate frames that require source-routing service from frames that require transparent-bridging service:

- An RII value of 1 specifies source routing.
- An RII value of 0 specifies transparent bridging.

6.1.4 Spanning-Tree Algorithm

The IEEE 802.1 committee issued the *Spanning Tree Algorithm* standard. Much of this standard, applicable to all MAC-level bridges, deals with how bridges operate in topologically complex environments that may contain parallel-bridge connections between multiple LANs. A transparent-bridging environment cannot tolerate parallel connections.

For example in Figure 6-7, two parallel bridges (**1** and **2**) connect the **Finance** and **Engineering** LANs. Without the spanning-tree algorithm, the following occurs when Node **J** on the **Finance** LAN first sends a frame to Node **K** on the **Engineering** LAN:

- Bridges **1** and **2** read the frame from **J** and to **K**.

As this is the first frame between **J** and **K**, the forwarding table of neither bridge contains an entry for **J** or **K**.
- Bridges **1** and **2** update their forwarding tables to indicate that **J** is in the direction of the **Finance** LAN.
- Bridges **1** and **2** flood the frame:
 - Bridge **1** relays the frame over interface **1**.
 - Bridge **2** relays the frame over interfaces **2** and **3** (for simplicity, this example will not trace the frame relayed over interface **3**).

- Node **K** receives two copies of the frame from Node **J**.

Generally, when a node receives duplicate frames, it is not fatal; although, duplicate frames use bandwidth inefficiently. However, the graver consequence, is how duplicate frames effect Bridges **1** and **2**.

Interface **2** on Bridge **2** ultimately reads the frame flooded by Bridge **1** onto interface **1**, and thus, Bridge **2** updates its forwarding table to show Node **J** in the direction of the **Engineering** LAN. Bridge **1** reads the frame flooded by Bridge **2** and updates its forwarding table to show Node **J** in the direction of the **Engineering** LAN. Consequently, the forwarding table of both bridges are now corrupt and neither bridge can properly forward a frame to Node **J**. The alternate routes (or *loops*) between hosts cause this corruption.

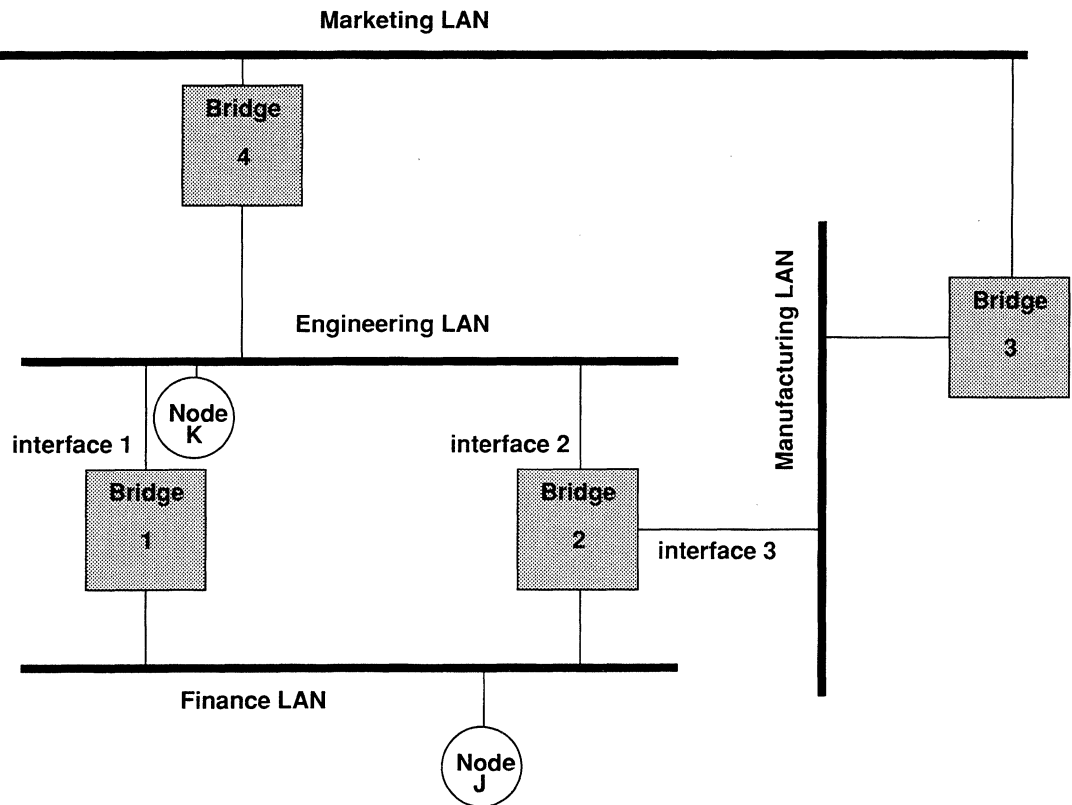


Figure 6-7. Parallel Bridge Topology

The *Spanning Tree Algorithm* (fully described in IEEE 802.1 *MAC Bridges*) ensures the existence of a loop-free topology in networks that contain parallel bridges. The algorithm provides a single path (composed of bridges and intervening LANs) between any two nodes in such an extended network. It also provides a high degree of fault tolerance by allowing for the automatic reconfiguration of the spanning-tree topology in the face of bridge or data-path failure.

The algorithm requires five management-assigned values to derive the spanning-tree topology:

- ❑ *Multicast address*
A multicast address specifies all bridges within the extended network.
- ❑ *Network-unique identifier*
Each bridge within the extended network must be identified by a network-unique identifier.
- ❑ *Unique port identifier*
Each bridge/LAN interface (or *port*) must be identified by a unique port identifier.
- ❑ *Priority*
Each port must be assigned a priority.
- ❑ *Cost*
Each port must be assigned a cost.

With these values assigned, bridges broadcast and process formatted frames (called *Bridge Protocol Data Units*, or BPDUs) to derive a single loop-free topology throughout the extended network. BPDU frame exchange is quick; thus, minimizing the time during which service is unavailable between hosts.

In constructing a loop-free topology, bridges within the extended network first determine the root bridge — the bridge with the best (that is, lowest) priority value. This bridge serves as the root of the loop-free topology.

After identifying the root bridge, all other bridges calculate path costs — the cost of the each path offered by each bridge port to the root bridge. Each bridge designates the port that offers the lowest-cost path to the root bridge as the *root port*. In the event of equal-path costs, the bridge designates the port with the best, or lowest, priority value as the root port. On each LAN within the extended network, one bridge (the one whose root port offers the lowest-cost path to the root bridge) is the designated bridge. The port that connects the LAN to the designated bridge is the designated port. This port carries all extended-network traffic to and from the LAN, and is said to be in the *forwarding* state.

This process ensures that all redundant ports (those providing parallel connections) are removed from service (placed in the *blocking* state). If a topological change occurs, or a bridge or data path fails, the algorithm derives a new spanning tree that may move some such ports from the blocking to the forwarding state.

Using Figure 6-7 as an example, implementing the Spanning Tree Algorithm could remove Bridge 1 from service and block Bridge 2/Interface 3. Figure 6-8 shows the resulting logical topology — a loop-free topology with only a single path between any two hosts.

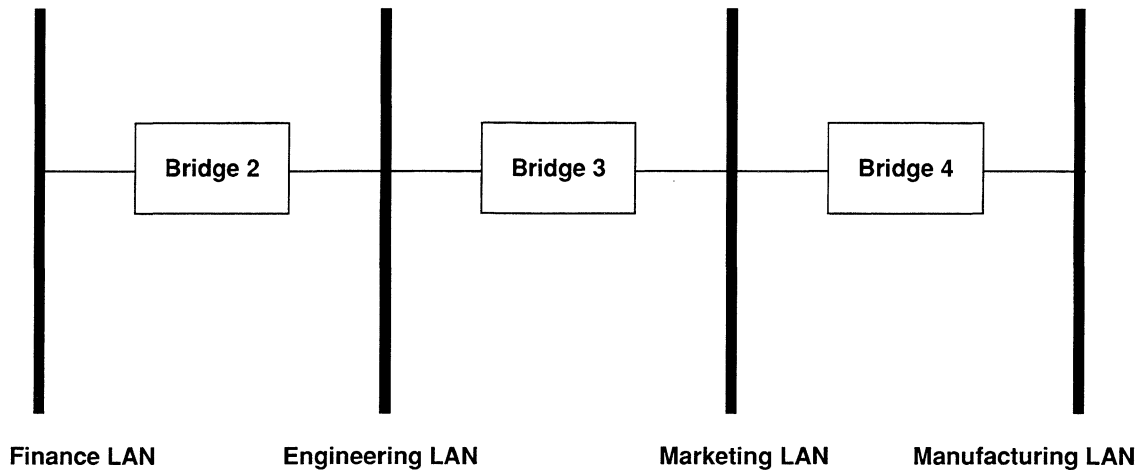


Figure 6-8. Spanning Tree (Loop-Free) Logical Topology

6.1.5 Filtering

Filters enable the bridge to either selectively relay or selectively drop a frames based on header fields within each of the four encapsulation methods that the bridge supports. These encapsulation methods are:

- Ethernet
- IEEE 802.2 logical-link control (LLC)
- IEEE 802.2 LLC with SNAP header
- Novell proprietary

Ethernet encapsulation (see Figure 6-9) prefixes an eight-octet preamble, six octets of destination-address information, six octets of source-address information, and two octets of protocol-type information to the frame. It appends a four-octet frame-check sequence to the frame.

Preamble 8 octets	Destination 6 octets	Source 6 octets	Type 2 octets	Data 46-1500 octets
----------------------	-------------------------	--------------------	------------------	------------------------

Figure 6-9. Ethernet Encapsulation

802.2 encapsulation (see Figure 6-10) prefixes one octet of destination service-access point identification, one octet of source service-access point identification, and one octet of control information to the frame. The 802.2 frame, in turn, will be encapsulated within a MAC-level media-specific frame.

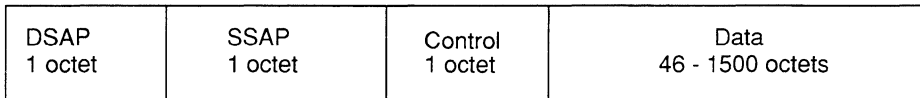


Figure 6-10. 802.2 Encapsulation

SNAP encapsulation (see Figure 6-11) is an extension of 802.2 encapsulation. It prefixes one octet of DSAP information, one octet of SSAP information, one octet of control information, three octets of organizational information, and two octets of Ethernet Type information to the frame. The SNAP structure is further encapsulated within a MAC-level medium-specific 802.x frame.

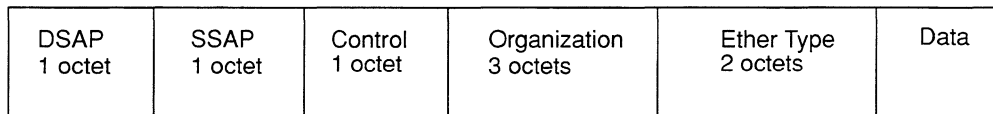


Figure 6-11. SNAP Encapsulation

Novell proprietary encapsulation (see Figure 6-12) prefixes an eight-octet preamble, six octets of destination-address information, six octets of source-address information, and two octets of frame-length information to the unchecksummed IPX frame (indicated by a value of FFFF). It appends a four-octet frame check sequence to the frame.

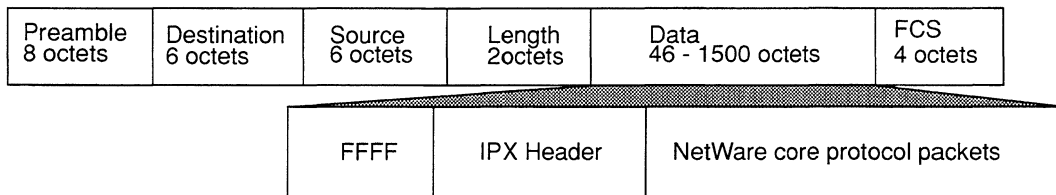


Figure 6-12. Novell Proprietary Encapsulation

Table 6-1 lists encapsulation support for each physical-access medium.

Table 6-1. Encapsulation/Media Matrix

	Ethernet Encapsulation	802.2 Encapsulation	SNAP Encapsulation	Novell Encapsulation
Ethernet/802.3 Media	Yes	Yes	Yes	Yes
Token Ring Media	No	Yes	Yes	No
FDDI Media	No	Yes	Yes	No
Point-to-Point Media	Yes	Yes	Yes	Yes

The bridge provides a set of pre-defined filter fields. Table 6-2 lists encapsulation methods and associated pre-defined fields. The bridge supplements basic filter functionality by allowing you to specify user-defined fields within each of the supported encapsulation formats. It also allows you to specify lists which contain a collection of value ranges to be filtered.

Table 6-2. Pre-defined Filter Fields

Encapsulation Method	Pre-defined Fields
All	MAC source address MAC destination address
Ethernet	Type
802.2	SSAP
SNAP	DSAP Organization Ethertype

6.2 Accessing Bridge Parameters

In order to access Bridge parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 6-13 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default parameters in NCU; any changes the network operator makes will affect every node configured thence on.

To access the bridge parameters, select **Protocols** and then **Bridge**. NCU displays the **BRIDGE** window which provides access to the bridge parameters (see Figure 6-14).

6.3 Editing Bridge Basic Parameters

Bridge basic parameters apply to both the transparent bridge and source-routing bridge domains. When you connect a node to a network segment that runs the bridge, and activate the bridge default settings, NCU automatically sets the bridge basic parameters for the node. This section describes how to modify and delete bridge basic parameters.

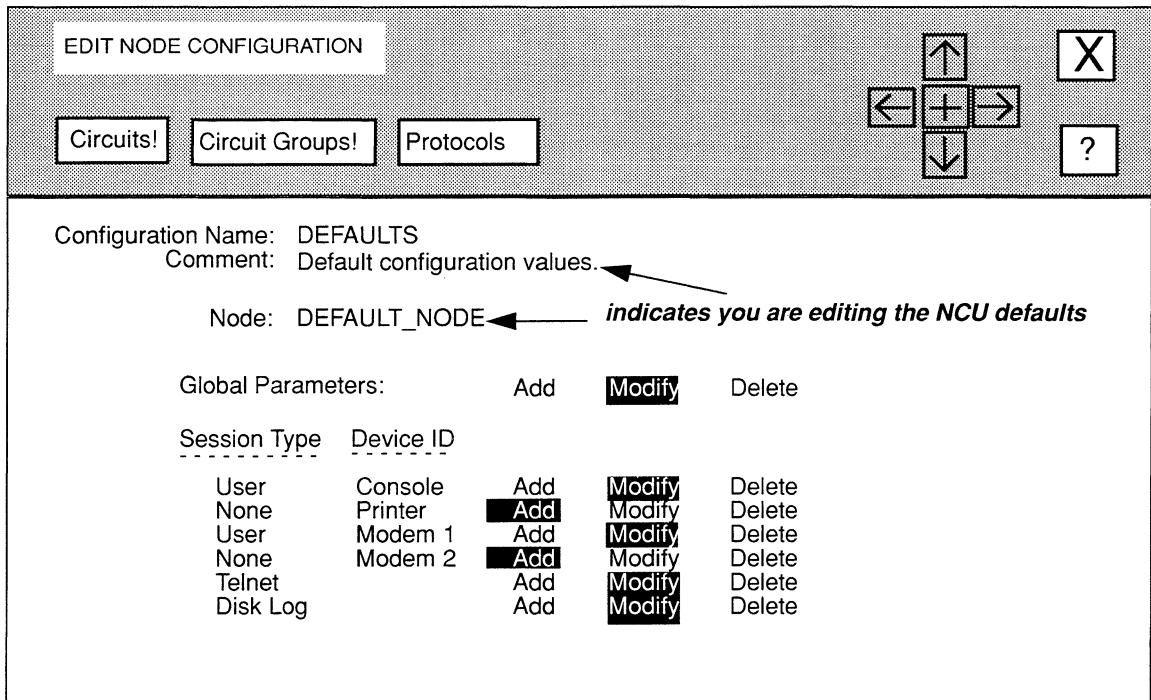


Figure 6-13. EDIT NODE CONFIGURATION Window for Default Settings

6.3.1 Modifying Bridge Basic Parameters

You modify bridge basic parameters from the **BRIDGE** window, as follows:

1. At Auto Enable, specify the state of the bridge software when the node boots.

This bridge-specific **Auto Enable** works with the global **Auto Enable** parameter to enable or disable the bridge software module when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the bridge software module (and every other application software module) is unconditionally disabled (you will need to enable the bridge manually with the NCL Interpreter after the node boots).
- When global **Auto Enable** is set to **YES**, the bridge software module (and every other application software module) is conditionally enabled; do one of the following:
 - Select **YES** to enable the bridge.
 - Select **NO** to disable the bridge (you will need to enable the bridge manually with the NCL Interpreter after the node boots).

2. At Forward Table Size, select the maximum size of the forwarding table.

The forwarding table contains the list of end-station addresses learned by the bridge, plus all source-address filters and destination-address filters. NCU provides seven responses: **53, 211, 523, 887, 1327, 3327, and 9551**.

To specify the maximum size of the forwarding table, estimate the number of end-stations that this bridge services, then double the figure, and select the next-highest response that NCU provides.

3. At Filter Table Size, select the maximum size of the filtering table.

The filter table contains the protocol and multicast-address filters that the bridge uses. NCU provides seven responses: **53, 211, 523, 887, 1327, 3327, and 9551**.

To specify the maximum size of the filter table, estimate the number of protocol filters and multicast-address filters that need to be in this table; then select the next highest response that NCU provides.

4. At Flood Interval, enter the interval (a number) during which (at most) a single frame will be flooded to an unlearned address.

Based on a user-configured timer, the bridge floods each frame to an unlearned MAC address (with the exception of multicast or broadcast addresses which are always flooded). To disable flood limiting, enter **0**.

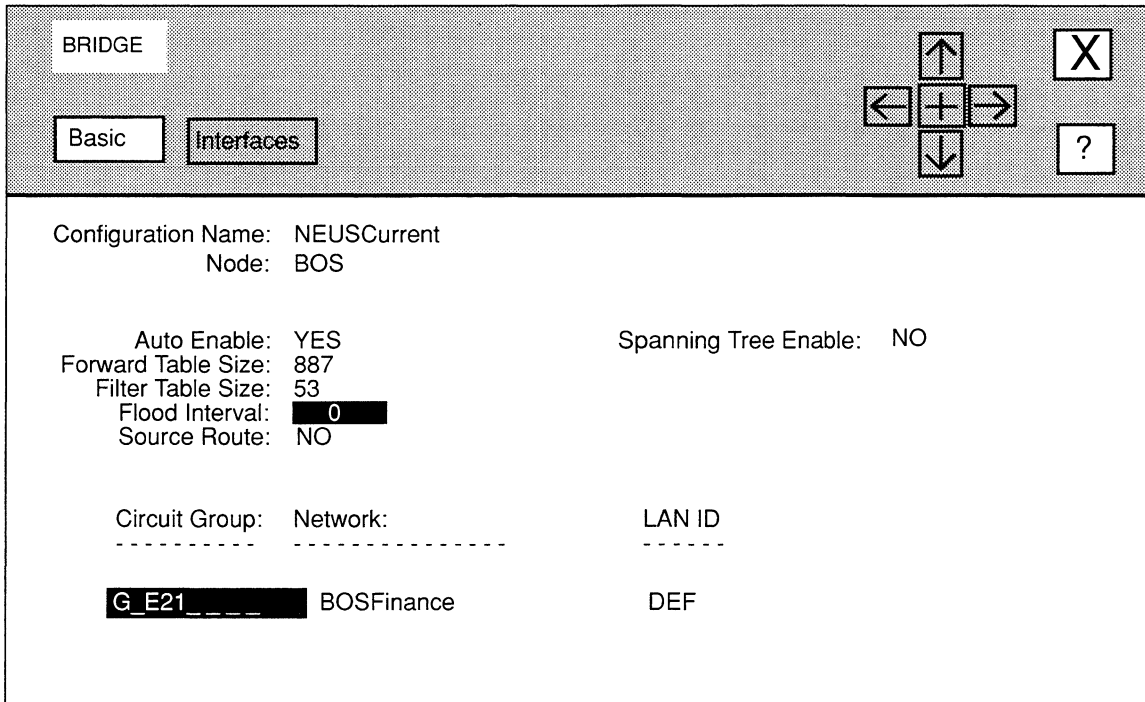


Figure 6-14. BRIDGE Window with Spanning Tree Enable set to NO

5. At Source Route, enable or disable source routing.

NO Disables source routing.

YES Enables source routing. If you select YES, NCU displays LAN ID which allows you to assign a numeric identifier that the bridge uses to construct routing designators. Enter a value from 0 to 4095 at LAN ID.

Note

Parallel source-routing bridges require unique network-wide LAN ID values. Non-parallel bridges do not require unique identifiers.

6. At Spanning Tree Enable, enable or disable the spanning tree algorithm.

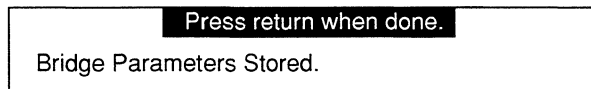
Note

If you enable source routing, the bridge automatically enables the spanning tree algorithm regardless of how you set **Spanning Tree Enable**.

- NO** Disables the spanning tree algorithm. If your network contains a single bridge or multiple, non-redundant bridges, select **NO**.
- YES** Enables the spanning tree algorithm. If your network contains redundant bridge/LAN connections as shown in Figure 6-8, select **YES**; NCU displays additional parameters. For instructions on how to set these parameters, go directly to *Section 6.3.1.1, Setting Spanning Tree Parameters*.

7. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.



6.3.1.1 Setting Spanning Tree Parameters.

When you set the **Spanning Tree Enable** parameter to **YES**, NCU displays additional parameters (see Figure 6-15). You set these parameters, as follows:

1. At Priority, select the bridge's priority within the spanning tree algorithm; keep in mind that the smaller the value, the more likely the bridge will be root.

Priority supplies the most-significant 16-bits of the unique 64-bit identifier that the spanning tree algorithm uses to identify the root bridge (the bridge with the best priority). NCU provides 18 responses: **1, 3, 7, 15, 31, 63, 127, 255, 511, 1023, 2047, 4095, 8191, 16383, 32767, 32768, 32769, 65535**.

BRIDGE		
Basic	Interfaces	
Configuration Name: NEUSCurrent Node: BOS		
Auto Enable: YES	Spanning Tree Enable: YES	
Forward Table Size: 887	Priority: 32768	
Filter Table Size: 53	Hello Time: [REDACTED]	
Flood Interval: 0	Max Age: [REDACTED]	
Source Route: NO	Forward Delay: [REDACTED]	
Circuit Group: _____	Network: _____	LAN ID: _____
G_E21_____	BOSFinance	DEF

Figure 6-15. BRIDGE Window with Spanning Tree Enable set to YES

Note

When you assign the following spanning tree parameters (**Hello Time**, **Max Age**, and **Forward Delay**), refer to the recommended values listed in Table 6-3.

2. **At Hello Time, enter the number of seconds (from 1 to 10) between BPDUs that the bridge transmits; refer to Table 6-3.**

BPDUs are periodic, formatted transmissions that bridges in the extended network exchange to convey configuration- and topology-change data.

3. **At Max Age, enter the maximum length of time the bridge stores configuration information; refer to Table 6-3**

The bridge declares a line down if it does not receive a BPDU for **Max Age** seconds; the bridge then sets the port state to *Listen*.

Table 6-3. Suggested Spanning Tree Parameter Values

When You Set Hello Time to:	Set Max Age to:	And Set Forward Delay to:
1	>=4	>=3
2	>=6	>=4
3	>=8	>=5
4	>=10	>=6
5	>=12	>=7
6	>=14	>=8
7	>=16	>=9
8	>=18	>=10
9	>=20	>=11
10	>=22	>=12

4. At Forward Delay, enter the amount of time a circuit group spends in the *Listening* and *Learning* states; refer to Table 6-3.

Forward Delay provides a timer that clocks a circuit group as it moves from state to state. Setting **Forward Delay** to the minimum causes the spanning tree to converge at its fastest rate.

As the spanning tree algorithm operates, it eventually places all circuit groups in either a *Forwarding* (enabled) or *Blocking* (disabled) state. Later, in response to network-topology changes, the algorithm may change the state of specific circuit groups. In order to prevent sudden state changes from causing network looping, the algorithm does not transition directly circuit groups from *Blocking* to *Forwarding*. Rather, it places them in two intermediates states:

❑ *Listening*

While in the *Listening* (stand-by) state, the circuit group receives network-generated BPDUs, but does not receive end-station-generated message traffic.

❑ *Learning*

When the **Forward Delay** Timer expires, the circuit group is placed in the *Learning* state. While in the *Learning* state, the circuit group receives network-generated BPDUs, and also receives end-station-generated traffic, which is subjected to the learning process but not relayed. When the **Forward Delay** Timer expires, the circuit group is placed in the *Forwarding* state.

5. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.
Bridge Parameters Stored.

6.3.2 Deleting Bridge Basic Parameters

You delete bridge basic parameters from the **BRIDGE** window. Simply select and then . NCU displays this window, press **[RETURN]** to clear it from the console:

Press return when done.
Bridge deleted.

6.4 Configuring Bridge Interfaces

You configure each bridge interface individually. The following sections describe how to modify and delete bridge interfaces.

6.4.1 Modifying Bridge Interfaces

NCU allows you to modify a bridge interfaces, as follows:

- Set certain source-routing and spanning-tree specific parameters (if you enabled these features when you set the bridge basic parameters).
- Configure filters for the interface.
- Configure the load-balancing option for the interface.

You modify bridge interfaces from the **BRIDGE CIRCUIT GROUPS** window (see Figure 6-16), which allows you to configure interface-specific parameters. To display this window for a particular interface, select the interface you wish to modify under **Circuit Group** in the **BRIDGE** window, and then select and .

The features you enabled (source routing and/or spanning tree), determine what parameters the **BRIDGE CIRCUIT GROUPS** window displays; if you enabled:

- Source routing only, or source routing and the spanning tree.
The window displays the **LAN ID**, **Cost**, and **Priority** parameters.
- Spanning tree only.
The window displays the **Cost** and **Priority** parameters.

You set these parameters, as follows:

1. **At LANID, assign a numeric identifier to the network the interface connects.**

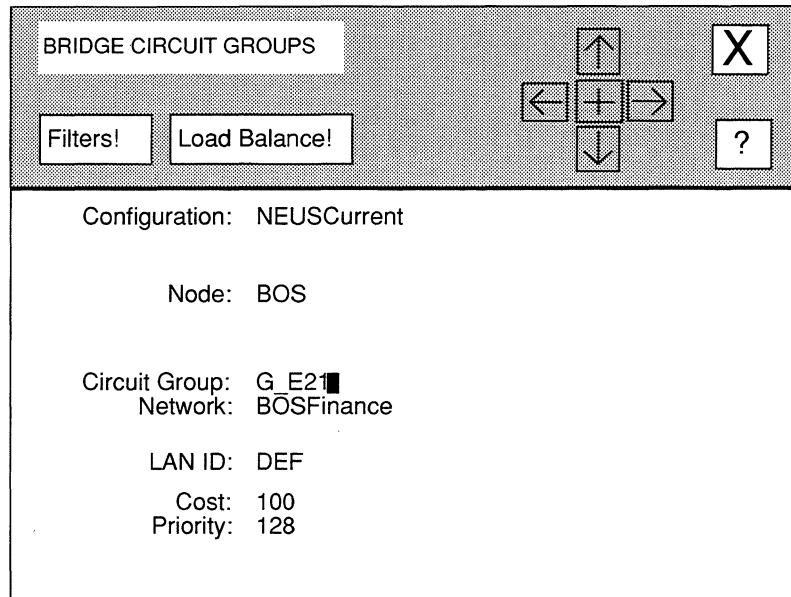


Figure 6-16. BRIDGE CIRCUIT GROUPS Window

2. **At Cost, select a relative cost value for the circuit group.**

Cost reflects the relative speed of the media, in that lower costs indicate high-speed media, while higher costs indicate low-speed media. Set **Cost** to direct network traffic to higher-speed media.

NCU provides 14 responses: 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 250, and 65535.

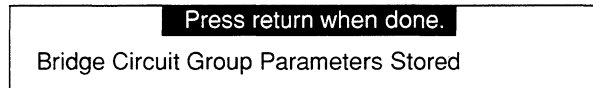
3. **At Priority, select a relative priority value for the circuit group.**

In the event of identical-**Cost** circuit groups, the spanning tree algorithm selects the circuit group with the better (lower) priority value. NCU provides 17 responses: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 16, 32, 64, 127, 128, 129, and 255.

At this point, you may construct filters for the interface (see *Section 6.4.1.1, Configuring Filters*) and/or configure the load-balance option (see *Section 6.4.1.2, Configuring the Load-Balance Option*). If you do not wish to configure filters or the load-balance option for this interface, go directly to step 4.

4. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.



NCU returns to the **BRIDGE** window, which displays the interface-specific parameters you just configured. Repeat this procedure for each additional bridge interface you wish to modify.

6.4.1.1 Configuring Filters

Filters apply to all in-coming bridge traffic across a circuit group. Conceptually, a filter consists of a:

- Rule*

A rule which identifies those packets the bridge will filter. A rule consists of these entities:

- A specified *field* (or fields) in the frame header.
- A *value* (or range of values) associated with the *field*.
- An *operator* (one of three values: **IGNORE**, **MATCH**, or **DON'T MATCH**) which specifies the relationship between *field* and *value*.

- Action*

An action the bridge will take when it receives a frame that meets the conditions of the rule.

- Precedence*

A precedence that identifies which action the bridge will take when a frame meets the conditions of more than one rule. You can construct up to 31 filters for each bridge circuit group. A filter precedence is designated by a decimal value from **1** to **31** — the higher the value, the greater the precedence.

You configure filters from the **BRIDGE CIRCUIT GROUPS** window for an interface. Simply select to display the **BRIDGE INTERFACE FILTERS** window (see Figure 6-17), which allows you to add, modify, and delete filters, as well as configure the interface to forward filtered traffic.

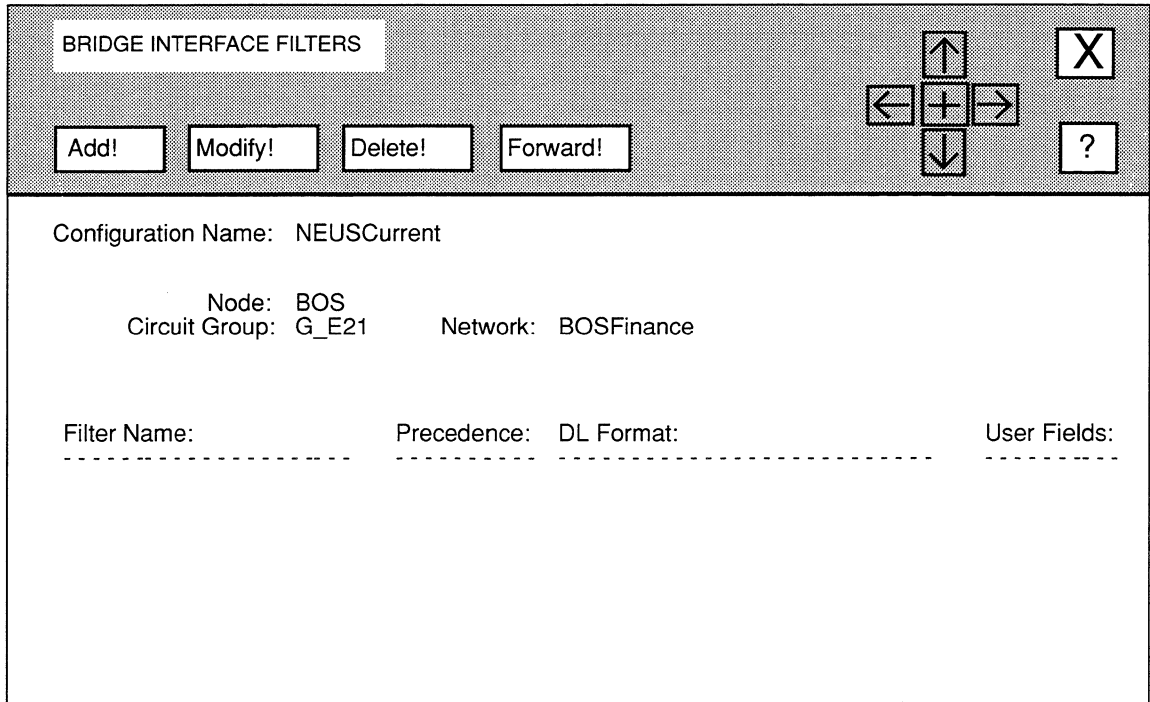


Figure 6-17. BRIDGE INTERFACE FILTERS Window

You can construct the following types of filters:

- MAC-Level Source and Destination Filters

These filters enable the bridge either to drop or to forward a frame based on its source and destination addresses; they filter either source addresses only, destination addresses only, or a specified combination of source and destination addresses. You can construct MAC-level source and/or destination address filters for any of the four supported encapsulation methods.

- Ethernet Filters

These filters enable the bridge either to drop or to forward a frame based on its Ethernet type; they filter Ethernet type values only, or a specified combination of Ethernet type values and MAC-level source and destination addresses.

❑ 802.2 LLC Filters

These filters enable the bridge either to drop or to forward a frame based on its destination and/or source service-access points; they filter source service-access points (SSAP) only, destination service-access points (DSAP) only, a specified combination of SSAP and DSAP values, or a specified combination of SSAP/DSAP values and MAC-level source and destination addresses.

❑ 802.2 SNAP Filters

These filters enable the bridge either to drop or to forward a frame based on its protocol or Ethernet type; they filter protocol ID values only, Ethernet type values only, a specified combination of protocol ID/Ethernet type values, or a specified combination of protocol ID/Ethernet type values and MAC-level source and destination addresses.

❑ Novell Filters

These filters enable the bridge either to drop or to forward Novell frames, or a specified combination of Novell-encapsulated frames and MAC-level source and destination addresses.

❑ User-Defined Filters

These filters enable the bridge either to drop or to forward traffic based on a specified bit pattern(s) in either the MAC or data-link header. You can construct user-defined filters to work with any pre-defined filters.

The following sections describe how to construct each of the above filters, as well as how to modify and delete filters, and configure the bridge to forward filtered traffic.

6.4.1.1.1 Adding MAC-Level Source and Destination Address Filters

You add MAC-level source and destination address filters from the **BRIDGE INTERFACE FILTERS** window, as follows:

1. Select .

NCU displays the **BRIDGE FILTER** window (see Figure 6-18).

2. At **Filter Name**, enter the filter name (a maximum of 12 keyboard characters).
3. At **DL Format**, select **MAC ONLY**.
4. At **User Defined Fields**, select **NO**.

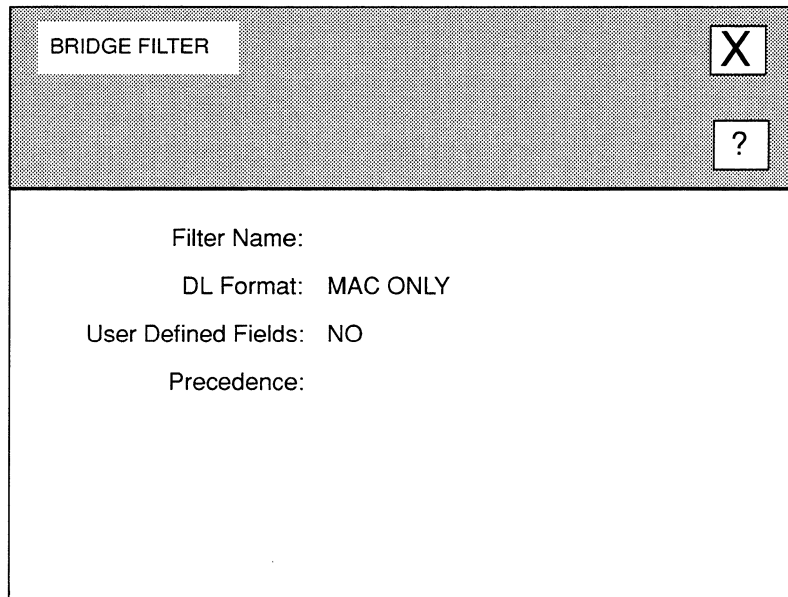


Figure 6-18. BRIDGE FILTER Window

5. **At Precedence, enter a priority value for this MAC-level source and destination address filter.**

You can construct up to 31 filters per bridge interface. The bridge uses the **Precedence** value when an in-coming frame meets multiple filter rules. In such an instance, the bridge applies the rule with the highest **Precedence** value. When two filters have equal **Precedence**, the bridge applies the filter you configured first.

6. **Select and then .**

NCU displays the **MAC ADDRESS FILTER** window (see Figure 6-19).

7. **At Action, select the response that identifies how the bride should dispose of frames that meet the filter rule.**

DROP..... Specifies the bridge discards such a frame.

DROP AND LOG Specifies the bridge discards such a frame and records an event message in the event log.

ACCEPT..... Specifies the bridge relays such a frame.

ACCEPT AND LOG Specifies the bridge relays such a frame and records an event message in the event log.

-
8. **At MAC Destination: Effect, select the operator to be applied to the MAC destination pattern specified by the MAC Dest (low) and MAC Dest (high) parameters.**

IGNORE Specifies you do not wish to filter MAC-level destination addresses; if you select **IGNORE**, go directly to step 9.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the destination-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:

- Enter the name of a MAC address list that contains addresses you wish to filter; type **[CONTROL] V** to display a values list of the current MAC address lists in your database. If you need to construct a MAC address list, see *Section 6.4.1.1.1, Constructing Filter Lists* for instructions.
- Enter a MAC address at the lower boundary of the address range you wish to filter.
- Enter a single MAC destination address that you wish to filter.

- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:

- If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
-

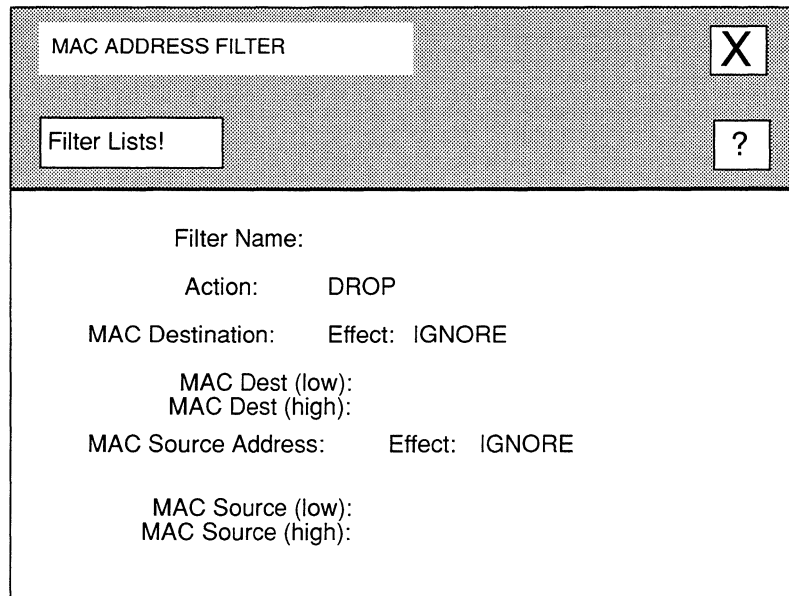


Figure 6-19. MAC ADDRESS FILTER Window

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the destination-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter; type **[CONTROL] V** to display a values list of the current MAC address lists in your database. If you need to construct a MAC address list, see *Section 6.4.1.1.1.1, Constructing Filter Lists* for instructions.
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

9. At **MAC Source Address: Effect**, select the operator to be applied to the MAC destination pattern specified by **MAC Dest (low)** and **MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level source addresses; if you select **IGNORE**, select and . NCU returns to the **BRIDGE INTERFACE FILTERS** window which now displays the filter you just constructed.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the source-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the source-address field of the frame header. Therefore, if you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as described in the bulleted information below.

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC

address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.

- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

10. Select and then .

NCU returns to the **BRIDGE INTERFACE FILTERS** window which now displays the filter you just constructed. Repeat this procedure for each additional MAC-level source and destination address filter you wish to configure.

6.4.1.1.2 Adding Ethernet Filters

You add Ethernet filters from the **BRIDGE INTERFACE FILTERS** window, as follows:

1. Select .

NCU displays the **BRIDGE FILTER** window (see Figure 6-18).

2. At **Filter Name**, enter the filter name (a maximum of 12 keyboard characters).
3. At **DL Format**, select **ETHERNET**.
4. At **User Defined Fields**, select **NO**.
5. At **Precedence**, enter a priority value for this Ethernet filter.

You can construct up to 31 filters per bridge interface. The bridge uses the **Precedence** value when an in-coming frame meets multiple filter rules. In such an instance, the bridge applies the rule with the highest **Precedence** value. When two filters have equal **Precedence**, the bridge applies the filter that was configured first.

6. Select and then .

NCU displays the **MAC ADDRESS FILTER** window (see Figure 6-19).

7. At **Action**, select the response that identifies how the bride should dispose of frames that meet the filter rule.

DROP..... Specifies the bridge discards such a frame.

DROP AND LOG..... Specifies the bridge discards such a frame and records an event message in the event log.

ACCEPT..... Specifies the bridge relays such a frame.

ACCEPT AND LOG..... Specifies the bridge relays such a frame and records an event message in the event log.

8. **At MAC Destination: Effect, select the operator to be applied to the MAC destination pattern specified by MAC Dest (low) and MAC Dest (high).**

IGNORE Specifies you do not wish to filter MAC-level destination addresses; if you select **IGNORE**, go directly to step 9.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP, DROP AND LOG, ACCEPT, or ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the destination-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

At **MAC Dest (low)**, do one of the following:

- Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
- Enter a MAC address at the lower boundary of the address range you wish to filter.
- Enter a single MAC destination address that you wish to filter.

At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:

- If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
- If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP, DROP AND LOG, ACCEPT, or ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the destination-address field of

the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

9. At **MAC Source Address: Effect**, select the operator to be applied to the MAC destination pattern specified by **MAC Dest (low)** and **MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level source addresses.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the source-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the source-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.

- Enter a single MAC destination address that you wish to filter.
- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

10. Select and then Save .

NCU displays the **ETHERNET TYPE FILTER** window (see Figure 6-20).

11. At **Ethernet Type (low)**, do one of the following:

- Enter the name of an Ethernet type list that contains the type values you wish to filter. Type **[CONTROL] V** to display a Values List of current Ethernet Type Lists; if you need to construct a new Ethernet Type list, select Filter Lists! and refer to *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions).
- Enter an Ethernet type value at the lower boundary of the type range you wish to filter
- Enter a single Ethernet type that you wish to filter

12. At **Ethernet Type (high)**, depending on how you set **Ethernet Type (low)**, do one of the following:

- If you entered the name of an Ethernet type list at **Ethernet Type (low)**, leave **Ethernet Type (high)** blank.
- If you entered a lower boundary range value at **Ethernet Type (low)**, enter an Ethernet type value at the upper boundary of the type range that you wish to filter at **Ethernet Type (high)**.
- If you wish to filter the single Ethernet type you entered at **Ethernet Type (low)**, leave **Ethernet Type (high)** blank.

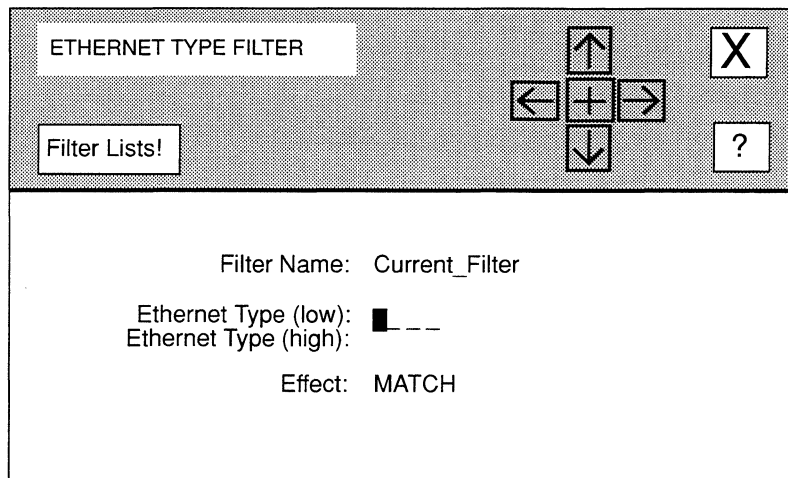


Figure 6-20. ETHERNET TYPE FILTER Window

13. At Effect, select the operator to be applied to the Ethernet type pattern specified by Ethernet Type (low) and Ethernet Type (high).

MATCH..... Specifies that the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **Ethernet Type (low)** and **Ethernet Type high** *includes* the type field of the frame header.

DON'T MATCH..... Specifies that the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **Ethernet Type (low)** and **Ethernet Type high** *does not include* the type field of the frame header.

14. Select and then .

NCU returns to the **BRIDGE INTERFACE FILTERS** window which now displays the filter you just constructed. Repeat this procedure for each additional Ethernet filter you wish to configure.

6.4.1.1.3 Adding 802.2 LLC Filters

You add 802.2 LLC filters from the **BRIDGE INTERFACE FILTERS** window, as follows:

1. Select to display the **BRIDGE FILTER** window (see Figure 6-18).
2. At **Filter Name**, enter the filter name (a maximum of 12 keyboard characters).
3. At **DL Format**, select **802.2 LLC**.
4. At **User Defined Fields**, select **NO**.
5. At **Precedence**, enter a priority value for this filter.

You can construct up to 31 filters per bridge interface. The bridge uses the **Precedence** value when an in-coming frame meets multiple filter rules. In such an instance, the bridge applies the rule with the highest **Precedence** value. When two filters have equal **Precedence**, the bridge applies the filter that was configured first.

6. Select and then .

NCU displays the **MAC ADDRESS FILTER** window (see Figure 6-19).

7. At **Action**, select the response that identifies how the bride should dispose of frames that meet the filter rule.

DROP Specifies the bridge discards such a frame.

DROP AND LOG Specifies the bridge discards such a frame and records an event message in the event log.

ACCEPT Specifies the bridge relays such a frame.

ACCEPT AND LOG Specifies the bridge relays such a frame and records an event message in the event log.

8. At **MAC Destination: Effect**, select the operator to be applied to the **MAC destination pattern** specified by **MAC Dest (low)** and **MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level destination addresses; if you select **IGNORE**, go directly to step 9.

MATCH Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the destination-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the destination-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.

- Enter a single MAC destination address that you wish to filter.
- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

9. At **MAC Source Address: Effect**, select the operator to be applied to the MAC destination pattern specified by **MAC Dest (low)** and **MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level source addresses.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the source-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the source-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address

range that you wish to filter at **MAC Dest (high)**.

- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

10. Select and then Save .

NCU displays the **802.2 LLC FILTER** window (see Figure 6-21).

11. At **DSAP Effect**, select the operator to be applied to the destination service access point pattern specified by **DSAP (low)** and **DSAP (high)**:

IGNORE Specifies you do not wish to filter destination service access points.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **DSAP (low)** and **DSAP (high)** *includes* the destination service access point of the frame header. If you select **MATCH**, you must set **DSAP (low)** and **DSAP (high)**, as follows:

- At **DSAP (low)**, do one of the following:
 - Enter the name of a SAP list that contains destination service access points you wish to filter. Type [**CONTROL**] **V** to display a Values List of the current SAP lists. If you need to construct a SAP list, see *Section 6.4.1.1.7, Constructing Filter Lists* for instructions.
 - Enter a destination service access point at the lower boundary of the DSAP range you wish to filter.
 - Enter a single destination service access point that you wish to filter.

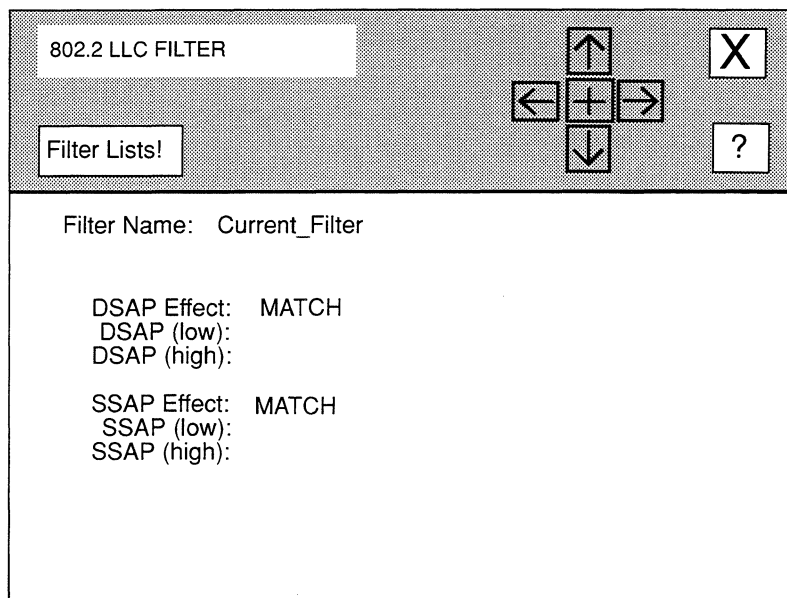


Figure 6-21. 802.2 LLC FILTER Window

- ❑ At **DSAP (high)**, depending on how you set **DSAP (low)**, do one of the following:
 - If you entered the name of a SAP list at **DSAP (low)**, leave **DSAP (high)** blank.
 - If you entered a lower boundary range value at **DSAP (low)**, enter a destination service access point at the upper boundary of the DSAP range that you wish to filter at **DSAP (high)**.
 - If you wish to filter the single destination service access point you entered at **DSAP (low)**, leave **DSAP (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP, DROP AND LOG, ACCEPT, or ACCEPT AND LOG**) if the pattern you specify at **DSAP (low)** and **DSAP (high)**

does not include the destination service access point of the frame header. If you select **DON'T MATCH**, you must set **DSAP (low)** and **DSAP (high)** as follows:

- At **DSAP (low)**, do one of the following:
 - Enter the name of a SAP list that contains destination service access points you wish to filter. Type [**CONTROL**] **v** to display a Values List of the current SAP lists. If you need to construct a SAP list, see *Section 6.4.1.1.7, Constructing Filter Lists* for instructions.
 - Enter a destination service access point at the lower boundary of the DSAP range you wish to filter.
 - Enter a single destination service access point that you wish to filter.
- At **DSAP (high)**, depending on how you set **DSAP (low)**, do one of the following:
 - If you entered the name of a SAP list at **DSAP (low)**, leave **DSAP (high)** blank.
 - If you entered a lower boundary range value at **DSAP (low)**, enter a destination service access point at the upper boundary of the DSAP range that you wish to filter at **DSAP (high)**.
 - If you wish to filter the single destination service access point you entered at **DSAP (low)**, leave **DSAP (high)** blank.

12. At SSAP Effect, select the operator to be applied to the source service access point pattern specified by SSAP (low) and SSAP (high):

- IGNORE** Specifies you do not wish to filter source service access points.
- MATCH**..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **SSAP (low)** and **SSAP (high)** *includes* the source service access point of the frame

header. If you select **MATCH**, you must set **SSAP (low)** and **SSAP (high)**, as follows:

- ❑ At **SSAP (low)**, do one of the following:
 - Enter the name of a SAP list that contains destination service access points you wish to filter. Type [**CONTROL**] **V** to display a Values List of the current SAP lists. If you need to construct a SAP list, see *Section 6.4.1.1.7, Constructing Filter Lists* for instructions.
 - Enter a source service access point at the lower boundary of the SSAP range you wish to filter.
 - Enter a single source service access point that you wish to filter.
- ❑ At **SSAP (high)**, depending on how you set **SSAP (low)**, do one of the following:
 - If you entered the name of a SAP list at **SSAP (low)**, leave **SSAP (high)** blank.
 - If you entered a lower boundary range value at **SSAP (low)**, enter a source service access point at the upper boundary of the SSAP range that you wish to filter at **SSAP (high)**.
 - If you wish to filter the single source service access point you entered at **SSAP (low)**, leave **SSAP (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **SSAP (low)** and **SSAP (high)** *does not include* the source service access point of the frame header. If you select **DON'T MATCH**, you must set **SSAP (low)** and **SSAP (high)** as follows:

- ❑ At **SSAP (low)**, do one of the following:
 - Enter the name of a SAP list that contains destination service access points you wish to filter. Type [**CONTROL**] **V** to display a

Values List of the current SAP lists. If you need to construct a SAP list, see *Section 6.4.1.1.7, Constructing Filter Lists* for instructions.

- Enter a source service access point at the lower boundary of the SSAP range you wish to filter.
- Enter a single source service access point that you wish to filter.
- At **SSAP (high)**, depending on how you set **SSAP (low)**, do one of the following:
 - If you entered the name of a SAP list at **SSAP (low)**, leave **SSAP (high)** blank.
 - If you entered a lower boundary range value at **SSAP (low)**, enter a source service access point at the upper boundary of the SSAP range that you wish to filter at **SSAP (high)**.
 - If you wish to filter the single source service access point you entered at **SSAP (low)**, leave **SSAP (high)** blank.

13. Select and then .

NCU returns to the **BRIDGE INTERFACE FILTERS** window which now displays the filter you just constructed. Repeat this procedure for each additional 802.2 LLC filter you wish to configure.

6.4.1.1.4 Adding 802.2 SNAP Filters

You add 802.2 SNAP filters from the **BRIDGE INTERFACE FILTERS** window, as follows:

1. Select to display the **BRIDGE FILTER** window (see Figure 6-18).
2. At **Filter Name**, enter the filter name (a maximum of 12 keyboard characters).
3. At **DL Format**, select **802.2 SNAP**.
4. At **User Defined Fields**, select **NO**.
5. At **Precedence**, enter a priority value for this filter.

You can construct up to 31 filters per bridge interface. The bridge uses the **Precedence** value when an in-coming frame meets multiple filter rules. In such an instance, the bridge applies the rule with the highest **Precedence** value. When two

filters have equal **Precedence**, the bridge applies the filter that was configured first.

6. Select and then **Continue**.

NCU displays the **MAC ADDRESS FILTER** window (see Figure 6-19).

7. **At Action, select the response that identifies how the bride should dispose of frames that meet the filter rule.**

DROP..... Specifies the bridge discards such a frame.

DROP AND LOG Specifies the bridge discards such a frame and records an event message in the event log.

ACCEPT..... Specifies the bridge relays such a frame.

ACCEPT AND LOG Specifies the bridge relays such a frame and records an event message in the event log.

8. **At MAC Destination: Effect, select the operator to be applied to the MAC destination pattern specified by MAC Dest (low) and MAC Dest (high).**

IGNORE Specifies you do not wish to filter MAC-level destination addresses; if you select **IGNORE**, go directly to step 9.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the destination-address field of the frame header. Therefore, if you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:

- Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
- Enter a MAC address at the lower boundary of the address range you wish to filter.
- Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the destination-address field of the frame header. Therefore, if you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC

address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.

- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

9. **At MAC Source Address: Effect, select the operator to be applied to the MAC destination pattern specified by MAC Dest (low) and MAC Dest (high).**

IGNORE Specifies you do not wish to filter MAC-level source addresses.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the source-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.

- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the source-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

10. Select and then .

NCU displays the **802.2 SNAP FILTER** window (see Figure 6-22).

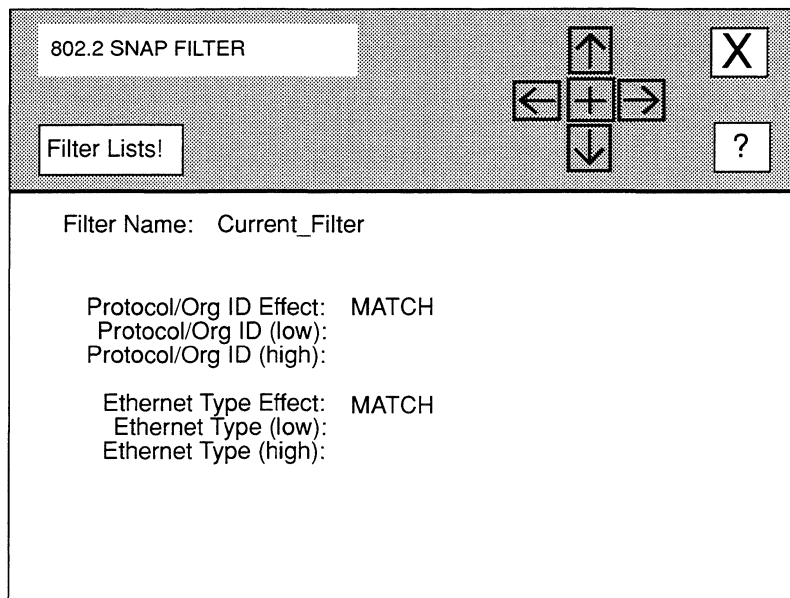


Figure 6-22. 802.2 SNAP FILTER Window

11. At Protocol/Org ID Effect, select the operator to be applied to the protocol ID pattern specified by Protocol/Org ID (low) and Protocol/Org ID (high):

IGNORE Specifies you do not wish to filter protocol IDs.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **Protocol/Org ID (low)** and **Protocol/Org ID (high)** includes the protocol ID of the frame header. If you select **MATCH**, you must set **Protocol/Org ID (low)** and **Protocol/Org ID (high)**, as follows:

At **Protocol/Org ID (low)**, do one of the following:

- Enter the name of a protocol ID list that contains the protocol IDs you wish to filter. Type [**CONTROL**] **v** to display a Values List of the current protocol ID lists. If you need to construct a protocol ID list, see *Section 6.4.1.1.7, Constructing Filter Lists* for instructions.

- Enter a protocol ID at the lower boundary of the protocol ID range you wish to filter.
- Enter a single protocol ID that you wish to filter.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **Protocol/Org ID (low)** and **Protocol/Org ID (high)** *does not include* the protocol ID of the frame header. If you select **DON'T MATCH**, you must set **Protocol/Org ID (low)** and **Protocol/Org ID (high)** as follows:

- At **Protocol/Org ID (high)**, depending on how you set **Protocol/Org (low)**, do one of the following:
 - If you entered the name of a protocol ID list at **Protocol/Org (low)**, leave **Protocol/Org (high)** blank.
 - If you entered a lower boundary range value at **Protocol/Org (low)**, enter a protocol ID at the upper boundary of the protocol ID range that you wish to filter at **Protocol/Org (high)**
 - If you wish to filter the single protocol ID that you entered at **Protocol/Org (low)**, leave **Protocol/Org (high)** blank.

12. At Ethernet Type Effect, select the operator to be applied to the Ethernet type pattern specified by Ethernet Type (low) and Ethernet Type (high):

IGNORE Specifies you do not wish to filter Ethernet types.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **Ethernet Type (low)** and **Ethernet Type (high)** *includes* the Ethernet type of the frame header. If you select **MATCH**, you must set **Ethernet Type (low)** and **Ethernet Type (high)**, as follows:

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **Ethernet Type (low)** and **Ethernet Type (high)** *does not include* the Ethernet type of the

frame header. If you select **DON'T MATCH**, you must set **Ethernet Type (low)** and **Ethernet Type (high)** as follows:

- ❑ At **Ethernet Type (low)**, do one of the following:
 - Enter the name of an Ethernet Type list that contains Ethernet types you wish to filter. Type [**CONTROL**] **V** to display a Values List of the current Ethernet Type lists. If you need to construct an Ethernet Type list, see *Section 6.4.1.1.7, Constructing Filter Lists* for instructions.
 - Enter an Ethernet type at the lower boundary of the Ethernet type range you wish to filter.
 - Enter a single Ethernet type that you wish to filter.
- ❑ At **Ethernet Type (high)**, depending on how you set **Ethernet Type (low)**, do one of the following:
 - If you entered the name of an Ethernet Type list at **Ethernet Type (low)**, leave **Ethernet Type (high)** blank.
 - If you entered a lower boundary range value at **Ethernet Type (low)**, enter an Ethernet type at the upper boundary of the Ethernet type range that you wish to filter at **Ethernet Type (high)**
 - If you wish to filter the Ethernet type you entered at **Ethernet Type (low)**, leave **Ethernet Type (high)** blank.

13. Select and then .

NCU returns to the **BRIDGE INTERFACE FILTERS** window which now displays the filter you just constructed. Repeat this procedure for each additional 802.2 SNAP filter you wish to configure.

6.4.1.1.5 Adding Novell Filters

You add Novell filters from the **BRIDGE INTERFACE FILTERS** window, as follows:

1. Select to display the **BRIDGE FILTER** window (see Figure 6-18).

2. At **Filter Name**, enter the filter name (a maximum of 12 keyboard characters).
3. At **DL Format**, select **NOVELL**.
4. At **User Defined Fields**, select **NO**.
5. At **Precedence**, enter a priority value for this filter.

You can construct up to 31 filters per bridge interface. The bridge uses the **Precedence** value when an in-coming frame meets multiple filter rules. In such an instance, the bridge applies the rule with the highest **Precedence** value. When two filters have equal **Precedence**, the bridge applies the filter that was configured first.

6. Select and then **Continue**.

NCU displays the **MAC ADDRESS FILTER** window (see Figure 6-19).

7. At **Action**, select the response that identifies how the bride should dispose of frames that meet the filter rule.

DROP..... Specifies the bridge discards such a frame.

DROP AND LOG Specifies the bridge discards such a frame and records an event message in the event log.

ACCEPT..... Specifies the bridge relays such a frame.

ACCEPT AND LOG Specifies the bridge relays such a frame and records an event message in the event log.

8. At **MAC Destination: Effect**, select the operator to be applied to the **MAC destination pattern specified by MAC Dest (low) and MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level destination addresses; if you select **IGNORE**, go directly to step 9.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the destination-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:

- Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).

- Enter a MAC address at the lower boundary of the address range you wish to filter.
- Enter a single MAC destination address that you wish to filter.
- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the destination-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

9. At **MAC Source Address: Effect**, select the operator to be applied to the MAC destination pattern specified by **MAC Dest (low)** and **MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level source addresses.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the source-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the source-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address

range that you wish to filter at **MAC Dest (high)**.

- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

10. Select and then .

NCU returns to the **BRIDGE INTERFACE FILTERS** window which now displays the filter you just constructed. Repeat this procedure for each additional 802.2 SNAP filter you wish to configure.

6.4.1.1.6 Adding User-Defined Filters

You add user-defined filters from the **BRIDGE INTERFACE FILTERS** window, as follows:

1. Select to display the **BRIDGE FILTER** window (see Figure 6-18).
2. At **Filter Name**, enter the filter name (a maximum of 12 keyboard characters).
3. At **DL Format**, do one of the following:

- If you are constructing a *complex filter*, a user-defined filter that includes one of the pre-defined filter types (**MAC ONLY**, **ETHERNET**, **802.2 LLC**, **802.2 SNAP**, or **NOVELL**), select the appropriate encapsulation method.
- If you are constructing a filter that only examines user-defined values, select **MAC ONLY**.

4. At **User Defined Fields**, select **YES**.
5. At **Precedence**, enter a priority value for this user-defined filter.

You can construct up to 31 filters per bridge interface. The bridge uses the **Precedence** value when an in-coming frame meets multiple filter rules. In such an instance, the bridge applies the rule with the highest **Precedence** value. When two filters have equal **Precedence**, the bridge applies the filter you configured first.

6. Select and then .

NCU displays the **MAC ADDRESS FILTER** window (see Figure 6-19).

7. At **Action**, select the response that identifies how the bride should dispose of frames that meet the filter rule.

DROP..... Specifies the bridge discards such a frame.

DROP AND LOG Specifies the bridge discards such a frame and records an event message in the event log.

ACCEPT..... Specifies the bridge relays such a frame.

ACCEPT AND LOG Specifies the bridge relays such a frame and records an event message in the event log.

8. **At MAC Destination: Effect, select the operator to be applied to the MAC destination pattern specified by the MAC Dest (low) and MAC Dest (high) parameters.**

IGNORE Specifies you do not wish to filter MAC-level destination addresses; if you select **IGNORE**, go directly to step 9.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the destination-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter; type **[CONTROL] V** to display a values list of the current MAC address lists in your database. If you need to construct a MAC address list, see *Section 6.4.1.1.1.1, Constructing Filter Lists* for instructions.
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

MAC ADDRESS FILTER [X] [?]

Filter Lists!

Filter Name:

Action: DROP

MAC Destination: Effect: IGNORE

MAC Dest (low):

MAC Dest (high):

MAC Source Address: Effect: IGNORE

MAC Source (low):

MAC Source (high):

Figure 6-23. MAC ADDRESS FILTER Window

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the destination-address field of the frame header. If you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as follows:

- At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter; type **[CONTROL] V** to display a values list of the current MAC address lists in your database. If you need to construct a MAC address list, see *Section 6.4.1.1.1.1, Constructing Filter Lists* for instructions.
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

9. At **MAC Source Address: Effect**, select the operator to be applied to the MAC destination pattern specified by **MAC Dest (low)** and **MAC Dest (high)**.

IGNORE Specifies you do not wish to filter MAC-level source addresses; if you select **IGNORE**, go directly to step 10.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *includes* the source-address field of the frame header. If you select **MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)**, as follows:

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.

- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.
 - If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP**, **DROP AND LOG**, **ACCEPT**, or **ACCEPT AND LOG**) if the pattern you specify at **MAC Dest (low)** and **MAC Dest (high)** *does not include* the source-address field of the frame header. Therefore, if you select **DON'T MATCH**, you must set **MAC Dest (low)** and **MAC Dest (high)** as described in the bulleted information below.

- ❑ At **MAC Dest (low)**, do one of the following:
 - Enter the name of a MAC address list that contains addresses you wish to filter (see *Section 6.4.1.1.7, Constructing Filter Lists*, for instructions on how to construct filter lists).
 - Enter a MAC address at the lower boundary of the address range you wish to filter.
 - Enter a single MAC destination address that you wish to filter.
- ❑ At **MAC Dest (high)**, depending on how you set **MAC Dest (low)**, do one of the following:
 - If you entered the name of a MAC address list at **MAC Dest (low)**, leave **MAC Dest (high)** blank.
 - If you entered the lower boundary of an address range at **MAC Dest (low)**, enter a MAC

address at the upper boundary of the address range that you wish to filter at **MAC Dest (high)**.

- If you wish to filter the single MAC address you entered at **MAC Dest (low)**, leave **MAC Dest (high)** blank.

10. Select and then .

If you are constructing a complex filter, NCU displays windows, as follows:

- If you selected **ETHERNET**, NCU displays the **ETHERNET TYPE FILTER** window.

Complete steps 11 through 14 of *Section 6.4.1.1.2, Adding Ethernet Filters* before going to step 11 of this section.

- If you selected **802.2 LLC**, NCU displays the **802.2 LLC FILTER** window.

Complete steps 11 through 13 of *Section 6.4.1.1.3, Adding 802.2 LLC Filters* before going to step 11 of this section.

- If you selected **802.2 SNAP**, NCU displays the **802.2 SNAP FILTER** window.

Complete steps 11 through 13 of *Section 6.4.1.1.4, Adding 802.2 SNAP Filters* before going to step 11 of this section.

- If you selected **NOVELL**, NCU displays the **USER DEFINED FIELDS** window (see Figure 6-24).

Go directly to step 11 of this section.

If you are not constructing a complex filter, NCU displays the **USER DEFINED FIELDS** window (see Figure 6-24), and you may go directly to step 11.

11. Select to display the **USER DEFINED FILTER** window.

12. At Header, select which header contains the bit pattern you are about to define.

MAC..... Specifies the user-defined bit pattern is in the MAC-level header.

DATA LINK Specifies the user-defined bit pattern is in the data-link header.

USER DEFINED FIELDS

Add! Modify! Delete!

Filter Name: Current_Filter

Field:	Header:	Offset:	Length:	Effect:	Action:
-----	-----	-----	-----	-----	-----

Figure 6-24. USER DEFINED FIELDS Window

13. At **Offset**, enter the starting location of the filtered bit pattern in reference to the first (most-significant) bit of the header (the most-significant bit of either the MAC-level header or the data-link header is referenced as bit 0).

For example, you designate an Ethernet multicast address by setting the lowest-order bit in the highest-order byte of the Ethernet address. Consequently, the following are valid Ethernet multicast addresses:

- 010000009999
- 0F0000009999

To filter multicast addresses, you would examine the multicast bit by entering 7 at **Offset**.

14. At **Length**, enter the bit length of the filtered field.

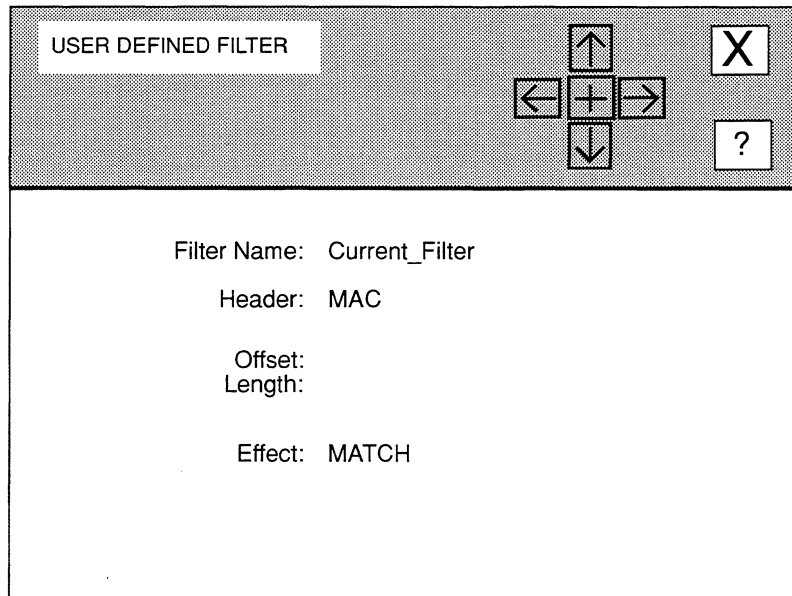


Figure 6-25. USER DEFINED FILTER Window

15. At Effect, select the operator to be applied to the user-defined bit pattern.

MATCH..... Specifies the bridge initiates the specified filter action (**DROP, DROP AND LOG, ACCEPT, or ACCEPT AND LOG**) if the user-defined pattern matches the frame header contents.

DON'T MATCH..... Specifies the bridge initiates the specified filter action (**DROP, DROP AND LOG, ACCEPT, or ACCEPT AND LOG**) if the user-defined pattern does not match the frame header contents.

16. Select and then .

NCU displays the **FILTER VALUES** window (see Figure 6-26).

17. Select to display the **ADD USER FILTER VALUE** window (see Figure 6-27), which allows you to specify a range of values for the bit field you defined with the **Offset** and **Length** parameters.

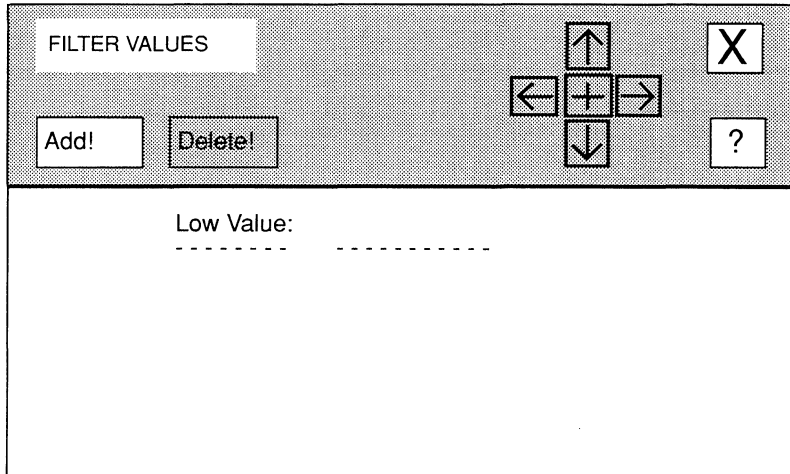


Figure 6-26. FILTER VALUES Window

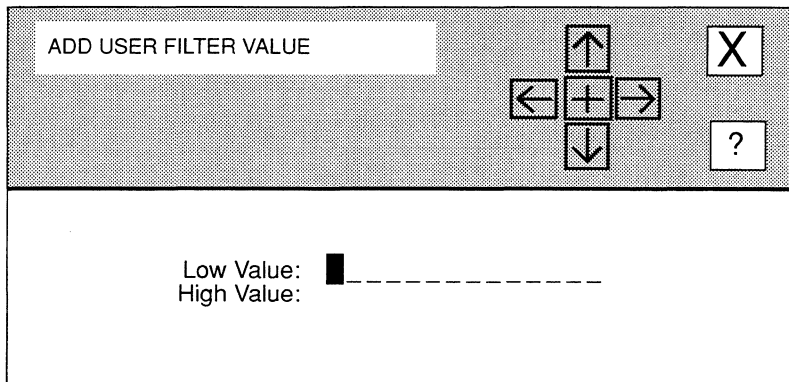


Figure 6-27. ADD FILTER USER VALUE Window

18. **At Low Value, enter the lower-boundary range of the user-defined bit pattern, as follows:**
 - Enter a hexadecimal value at the lower boundary of the user-defined range that you wish to filter.
 - Enter a single hexadecimal value that you wish to filter.
19. **At High Value, enter the upper-bound range of the user-defined bit pattern, as follows:**
 - If you entered a lower boundary at **Low Value**, enter a hexadecimal value at the upper boundary of the user-defined range that you wish to filter.
 - If you want to filter the single value that you entered at **Low Value**, leave **High Value** blank.
20. Select and then .

NCU returns to the **FILTER VALUES** window, which now displays the user-defined range you just entered.
21. Select and then .

NCU returns to the **USER DEFINED FIELDS** window, which now displays the filter you just constructed.
22. Select and then to return to the **BRIDGE INTERFACE FILTERS** window.

6.4.1.1.7 Constructing Filter Lists

A filter list contains a range of values that can be used in conjunction with the filtering of pre-defined fields (Table 6-2 lists the pre-defined fields). A filter list contains a symbolic name and a collection of ranges. When you specify a list name in a bridge filter, the bridge checks packets against the range of values specified by the list.

Table 6-4. Public Ethernet Type Field Values

Type Field	Company Name	Type Field	Company Name
0600	XNS Internet	807A	Matra
0800	DoD Internet	807C	University of Michigan
0801	X.75 Internet	807D to 8080	Vitalink Communications
0802	NBS Internet	8081 to 8083	Counterpoint Communications
0803	ECMA Internet	809B	Kinetics
0804	CHAOSNet	809C to 809E	Datability
0805	X.25 Level 3	809F	Spider Systems Ltd.
0806	Ethernet ARP	80A3	Nixdorf Computers
0888 to 088A	Xyplex	80A4 to 80B3	Siemens Gammasonics, Inc.
6010 to 6014	3Com Corporation	80C0 to 80C3	Digital Comm. Assoc. Inc.
7020 to 7029	LRT	80C6	Pacer Software
8006	Nestar	80C7	Applitek Corporation
8008	AT&T	80C8 to 80CC	Intergraph Corporation
8013 to 8016	Silicon Graphics	80CD to 80CE	Harris Corporation
8019	Apollo Computer	80CF to 80D2	Taylor Instrument
802E	Tymshare	80D3 to 80D4	Rosemount Corporation
802F	Tigan	80DD	Varian Associates
8035	Stanford University	80DE to 80DF	Integrated Solutions
8036	Aeonic Systems	80E0 to 80E3	Allen-Bradley
8044	Planning Research Corp.	80E4 to 80F0	Datability
8046 to 8047	AT&T	80F2	Retix
8049	ExperData	80F3 to 80F5	Kinetics
805B to 805C	Stanford University	80F7	Apollo Computer
805D	Evans & Sutherland	80FF to 8103	Wellfleet Communications
8060	Little Machines	8069	AT&T
8062	Counterpoint Communications	807B	Dansk Data Elektronik A/S
8065 to 8066	University of Mass. at Amherst	8130	Waterloo Microsystems, Inc.
8067	Veeco Integrated Automation	8131	VG Laboratory Sys.
8068	General Dynamics	8137 to 8138	Novell, Inc.
806A	Autophon	8139 to 813D	KTI
806C	ComDesign	0101 to 01FF	Experimental
806D	Compugraphic Corporation	9000	Loopback
806E to 8077	Landmark Graphic		

You can construct the following types of filter lists:

- MAC Address Filter Lists
Specify ranges of media-access-control (physical level) addresses.
- Ethernet Type Filter Lists
Specify ranges of Ethernet type values. Table 6-4 provides a partial list of Ethernet type values.
- SAP Filter Lists
Specify ranges of destination or source service-access points.
- Protocol ID/Organization Code Filter Lists
Specify ranges of SNAP protocol/organization identifiers.

You construct filters from the **BRIDGE FILTER LISTS** window (see Figure 6-28). You can display this window by selecting **Filters!** in the **MAC ADDRESS FILTER** window, the **ETHERNET TYPE FILTER** window, the **802.2 LLC FILTER** window, or the **802.2 SNAP FILTER** window. For instructions on how to construct a specific type of filter list, refer to the appropriate section of this guide.

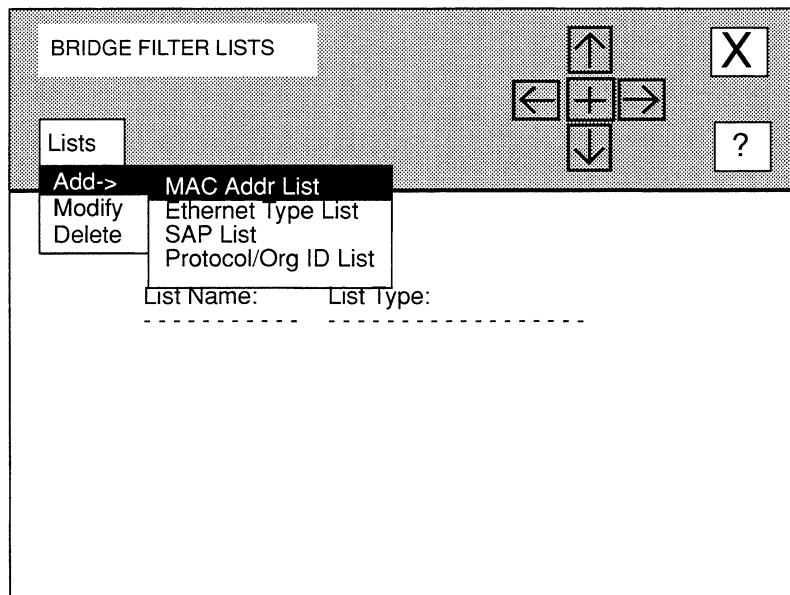


Figure 6-28. BRIDGE FILTER LISTS Window

6.4.1.1.7.1 Constructing MAC Address Filter Lists

You construct MAC address filter lists from the **BRIDGE FILTER LISTS** window, as follows:

1. Select and then to display the sub-menu in Figure 6-28.
2. Select to display the LIST NAME window (see Figure 6-29).
3. At List Name, enter the name of the MAC address filter list.
4. Select and then .

NCU displays the **MAC ADDRESS LIST** window (see Figure 6-30).

5. Select to display the **ADD MAC ADDRESS RANGE** window (see Figure 6-31).
6. At **Low MAC Addr**, enter the MAC address at the lower boundary of the filtered MAC address range.

If you want the list range to consist of a single value, enter the value at **Low MAC Addr** and proceed directly to step 8.

7. At **High MAC Addr**, enter the MAC address at the upper boundary of the filtered range.
8. Select and then .

NCU returns to the **MAC ADDRESS LIST** window, which now displays the ranges in the filter list you just constructed:

- to add additional ranges to this MAC address filter list, repeat steps 5 through 8
- to delete a range from the Ethernet type filter list, simply select the range under **Low MAC Addr** and select

The image shows a graphical user interface window titled "LIST NAME". The window has a title bar with the text "LIST NAME" on the left and two buttons, "X" and "?", on the right. Below the title bar is a text input field labeled "List Name:" followed by a dashed line for text entry.

Figure 6-29. LIST NAME Window

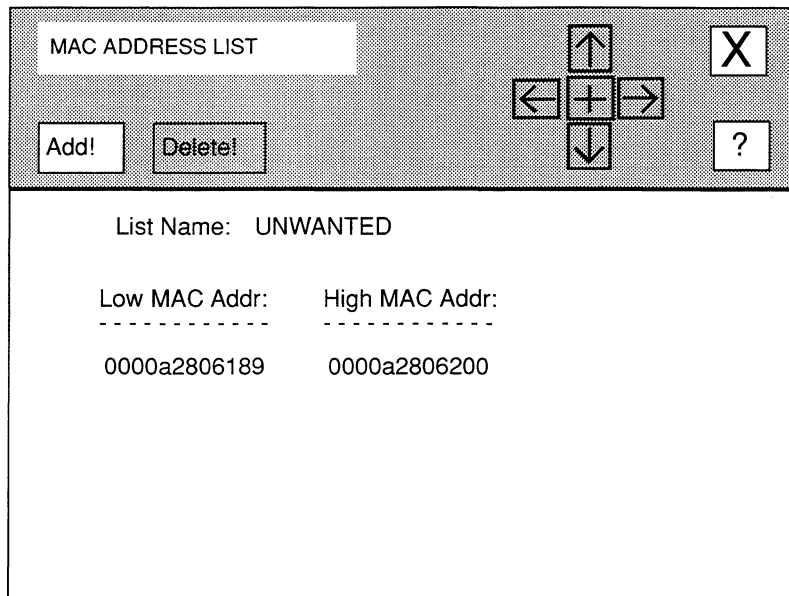


Figure 6-30. MAC ADDRESS LIST Window

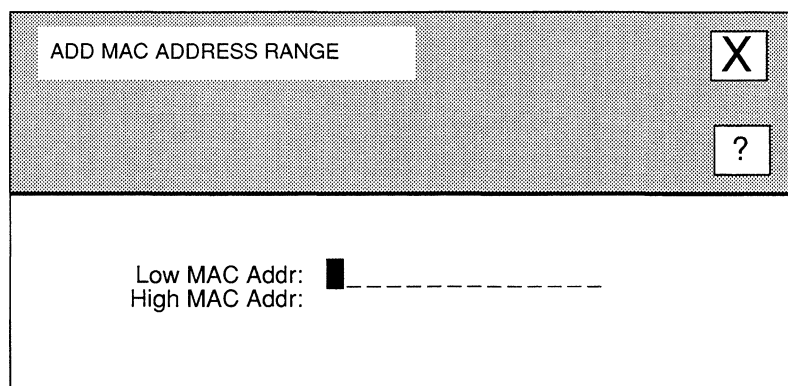


Figure 6-31. ADD MAC ADDRESS RANGE Window

6.4.1.1.7.2 Constructing Ethernet Type Filter Lists

You construct Ethernet type filter lists from the **BRIDGE FILTER LISTS** window, as follows:

1. Select and then to display the sub-menu in Figure 6-28.
2. Select to display the LIST NAME window (see Figure 6-29).
3. At List Name, enter the name of the Ethernet type filter list.
4. Select and then .

NCU displays the **ETHERNET TYPE LIST** window (see Figure 6-32).

5. Select to display the **ETHERNET TYPE RANGE** window (see Figure 6-33).
6. At **Ethernet Type Low**, enter the Ethernet type at the lower boundary of the Ethernet type range.
If you want the list range to consist of a single value, enter the value at **Ethernet Type Low** and proceed directly to step 9.
7. At **Ethernet Type High**, enter the Ethernet type at the upper boundary of the Ethernet type range.

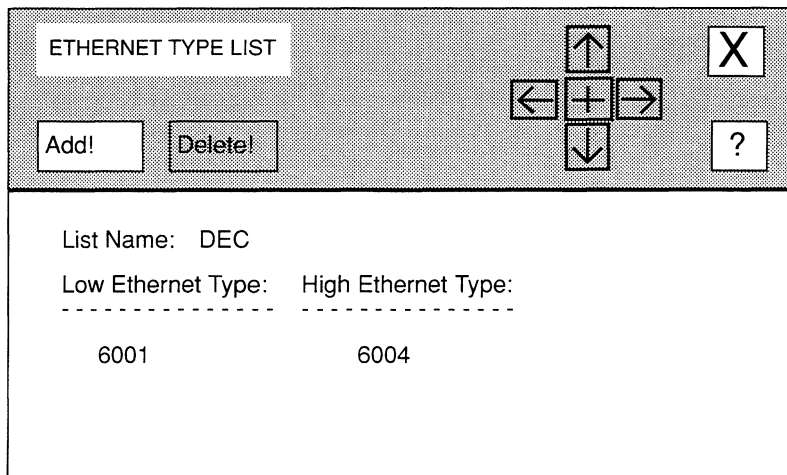


Figure 6-32. ETHERNET TYPE LIST Window

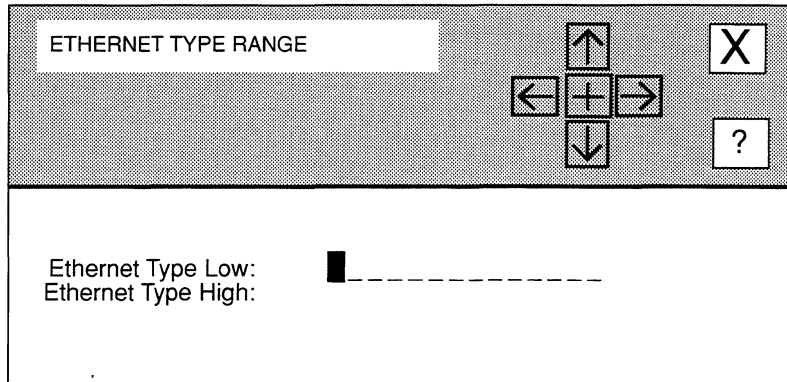


Figure 6-33. ETHERNET TYPE RANGE Window

8. Select **X** and then **Save** .

NCU returns to the **ETHERNET TYPE LIST** window, which now displays the ranges in the filter list you just constructed:

- to add additional ranges to this Ethernet type filter list, repeat steps 5 through 8
- to delete a range from the Ethernet type filter list, simply select the range under **Low Ethernet Type** and select **Delete**

NCU returns to the **ETHERNET TYPE LIST** window, which now displays the filter list you just constructed. To construct additional Ethernet Type filter lists, repeat this procedure.

6.4.1.1.7.3 Constructing SAP Filter Lists

You construct SAP filter lists from the **BRIDGE FILTER LISTS** window, as follows:

1. Select **Lists** and then **Add ->** to display the sub-menu in Figure 6-28.
2. Select **SAP List** to display the **LIST NAME** window (see Figure 4-29).
3. At **List Name**, enter the name of the Ethernet type filter list.
4. Select **X** and then **Save** .

NCU displays the **SAP LIST** window (see Figure 6-34).

SAP LIST

Add! Delete!

← + →

↑ ↓

X ?

List Name: SAPFILTER

SAP Value: 08 SAP High: 34

Figure 6-34. SAP LIST Window

5. Select to display the **SAP RANGE** window (see Figure 6-35).
 6. At **SAP Low**, enter the SAP value at the lower boundary of the filtered SAP range.
If you want the list range to consist of a single value, enter the value at **SAP Low** and proceed directly to step 8.
 7. At **SAP High**, enter the SAP value at the upper boundary of the filtered SAP range.
 8. Select and then .
- NCU returns to the **SAP LIST** window, which now displays the ranges in the filter list you just constructed:
- to add additional ranges to this SAP filter list, repeat steps 5 through 8
 - to delete a range from the SAP filter list, simply select the range under **SAP Low** and select .

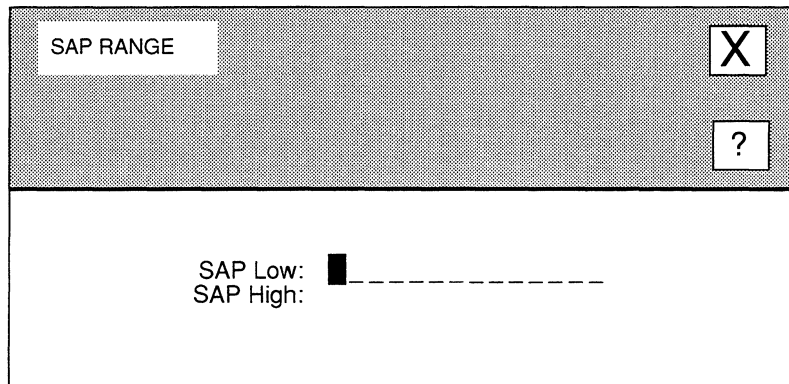


Figure 6-35. SAP RANGE Window

6.4.1.1.7.4 Constructing Protocol ID/Organization Code Filter Lists

You construct protocol ID/organization code filter lists from the **BRIDGE FILTER LISTS** window, as follows:

1. Select and then to display the sub-menu in Figure 6-28.
 2. Select to display the LIST NAME window (see Figure 6-29).
 3. At List Name, enter the name of the Ethernet type filter list.
 4. Select and then .
- NCU displays the **PROTOCOL/ORG ID LIST** window (see Figure 6-36).
5. Select to display the **PROTOCOL/ORG ID RANGE** window (see Figure 6-37).
 6. At Protocol/Org. ID Low, enter the protocol ID range value for the lower boundary of the filtered protocol ID range.

If you want the list range to consist of a single value, enter the value at **Protocol/Org. ID Low** and proceed directly to step 8.
 7. At Protocol/Org. ID High, enter the protocol ID range value for the upper boundary of the filtered protocol ID range.

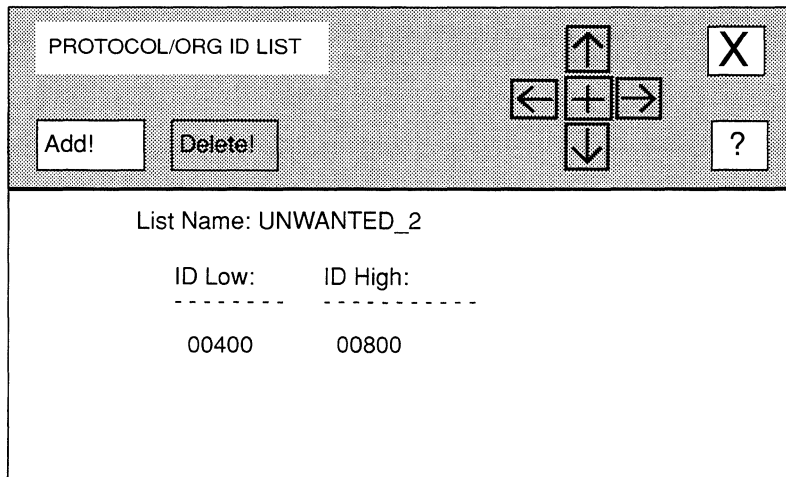


Figure 6-36. PROTOCOL/ORG ID LIST Window

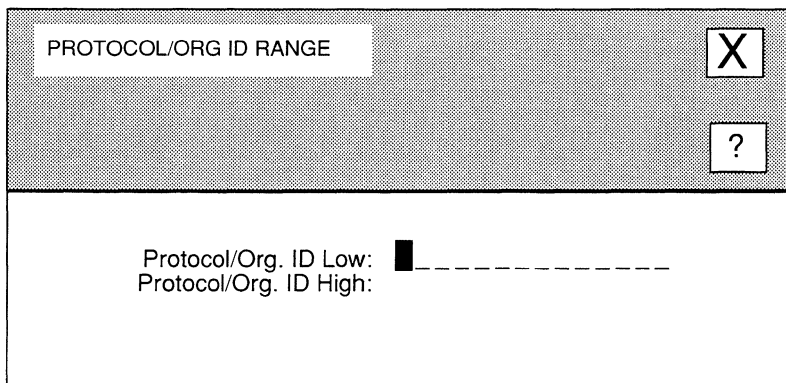


Figure 6-37. PROTOCOL/ORG ID RANGE Window

8. Select and then .

NCU returns to the **PROTOCOL/ORG ID LIST** window, which now displays the ranges in the filter list you just constructed:

- to add additional ranges to this protocol ID/organization code filter list, repeat steps 5 through 8
- to delete a range from the protocol ID/organization code filter list, simply select the range under **ID Low** and select .

6.4.1.1.7.5 Modifying Filter Lists

You modify filter lists from the **BRIDGE FILTER LISTS** window. Simply select the filter list you wish to modify under **List Name**; then select and . Depending on the type of filter list you selected, NCU displays windows, as follows:

- if you selected a MAC address filter list
NCU displays the **MAC ADDRESS LIST** window, which displays the current ranges in that filter list. You may now add and delete ranges (see steps 5 through 8 of *Section 6.4.1.1.7.1, Constructing MAC Address Filter Lists*, for instructions).
- if you selected an Ethernet type filter list
NCU displays the **ETHERNET TYPE LIST** window, which displays the current ranges in that filter list. You may now add and delete ranges (see steps 5 through 8 of *Section 6.4.1.1.7.2, Constructing Ethernet Type Filter Lists*, for instructions).
- if you selected a SAP filter list
NCU displays the **SAP LIST** window, which displays the current ranges in that filter list. You may now add and delete ranges (see steps 5 through 8 of *Section 6.4.1.1.7.3, Constructing SAP Filter Lists*, for instructions).
- if you selected a protocol ID/organization code filter list
NCU displays **PROTOCOL/ORG ID LIST** window, which displays the current ranges in that filter list. You may now add and delete ranges (see steps 5 through 8 of *Section 6.4.1.1.7.4, Constructing Protocol ID/Organization Code Filter Lists*, for instructions).

6.4.1.1.7.6 Deleting Filter Lists

You delete filter lists from the **BRIDGE FILTER LISTS** window. Simply select the filter list you wish to delete under **List Name**; then select and .

6.4.1.1.8 Forwarding Filtered Traffic

Generally, the bridge treats traffic that is forwarded as the result of filtering the same as any other traffic. The bridge directs frames for known destinations to a bridge port “in the direction” of the destination, and floods frame for unknown destinations to all interfaces. However, you can configure the bridge to direct (or *forward*) filtered traffic to specific interfaces.

You configure the bridge to forward filtered traffic from the **BRIDGE INTERFACE FILTERS** window. Simply select the filter you wish to forward under **Filter Name**, and then select .

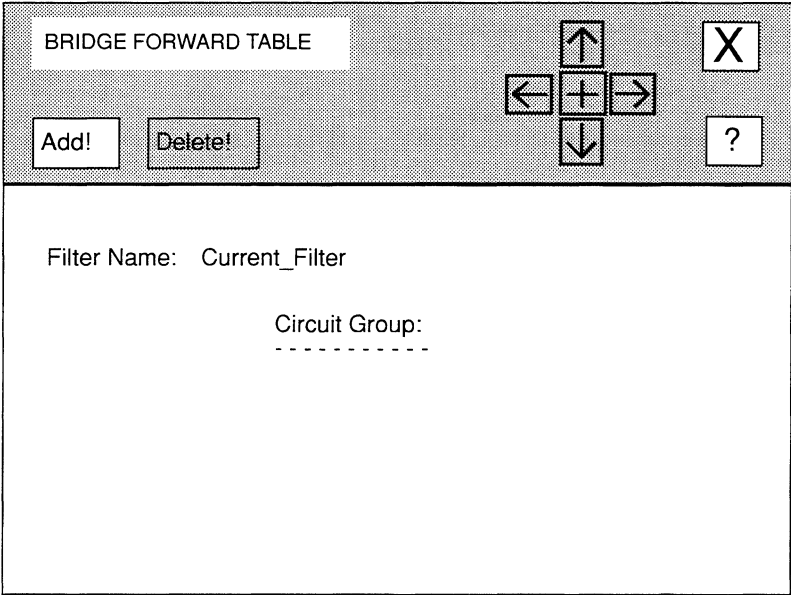


Figure 6-38. BRIDGE FORWARD TABLE Window

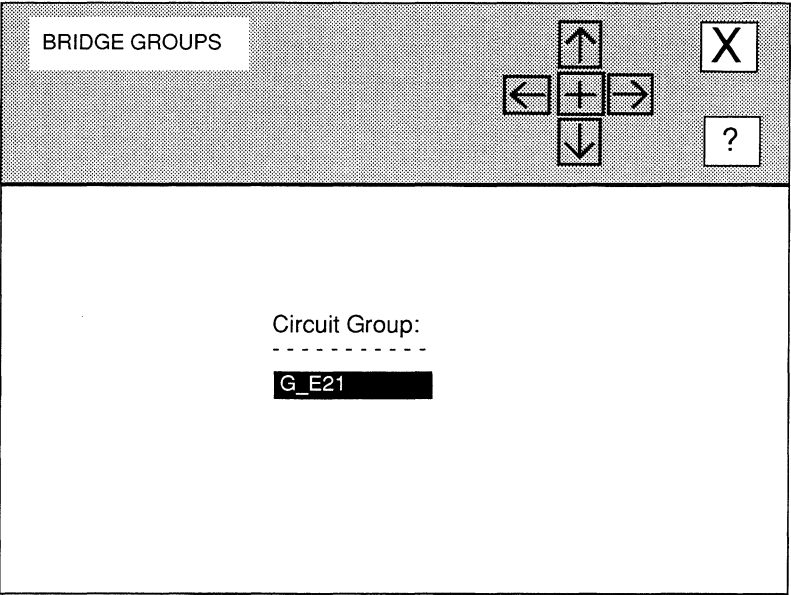


Figure 6-39. BRIDGE GROUPS Window

NCU displays the **BRIDGE FORWARD TABLE** window (see Figure 6-38). Select to display the **BRIDGE GROUPS** window (see Figure 6-39). Select the circuit group, under **Circuit Group**, over which you wish to forward the filtered traffic. Then select and . NCU returns to the **BRIDGE FORWARD TABLE** window, which now displays the circuit group you selected. Select and to return to the **BRIDGE INTERFACE FILTERS** window.

6.4.1.1.9 Deleting Filters

You delete filters from the **BRIDGE INTERFACE FILTERS** window. Simply select the filter you wish to delete under **Filter Name**, and then select .

6.4.1.2 Configuring the Load-Balance Option

The load-balance option allows you to direct specified traffic (identified by a designated Ethernet type field value) to a specified circuit. For example, you can use the load-balance option to configure all CHAOSNet frames (type field = 0804) to travel over a specific circuit.

You can only configure one load-balancing option per interface. However, you can assign multiple protocol types and circuits to the load-balancing option.

BRIDGE LOAD BALANCE

Add! Delete!

↑
← + →
↓

X

?

Node: BOS

Circuit Group: G_E21

Circuit: ----- Protocol: -----

Figure 6-40. BRIDGE LOAD BALANCE Window

You configure the load-balancing option from the **BRIDGE CIRCUIT GROUPS** window (see Figure 6-16), which allows you to configure interface-specific parameters. To display this window for an interface, first select the circuit group over which you wish to balance traffic, under **Circuit Group**, in the **BRIDGE** window. Next, select and then . NCU displays the **BRIDGE CIRCUIT GROUPS** window for that circuit group. Select to display the **BRIDGE LOAD BALANCE OPTION** window (see Figure 6-40). You may then configure the load-balance option, as follows:

1. Select to display the **SELECT CIRCUIT** window (see Figure 6-41).
 2. Select the specific circuit, under **Circuit**, which will carry protocol-type traffic.
 3. Select and .
- NCU displays the **SELECT PROTOCOL** window (see Figure 6-42).
4. At **Protocol**, enter the 12-digit hexadecimal protocol-type value that identifies the protocol you wish to relay to the circuit you selected in step 2.

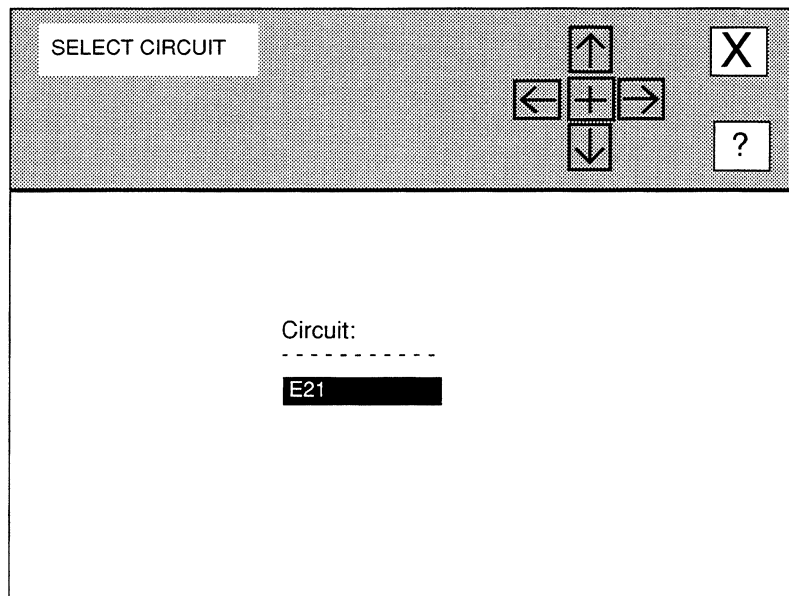


Figure 6-41. SELECT CIRCUIT Window

5. Select and .

NCU returns to the **BRIDGE LOAD BALANCE** window, which now displays the load-balancing selection you just configured. Repeat this procedure for each additional load-balancing selection you wish to configure for this option.

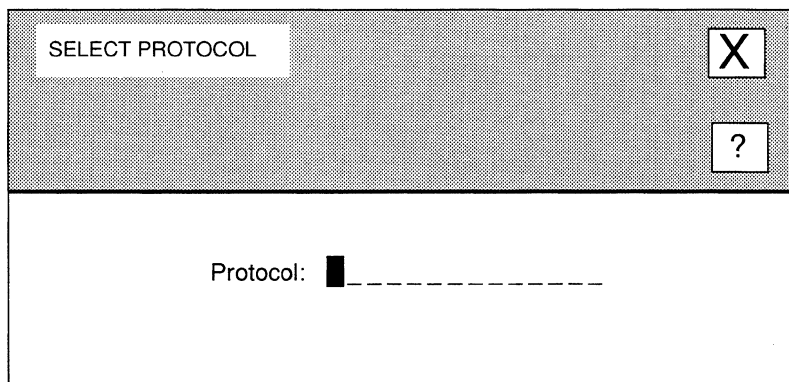


Figure 6-42. SELECT PROTOCOL Window

6.4.2 Deleting Bridge Interfaces

You delete bridge interfaces from the **BRIDGE** window. First, select the interface under **Circuit Group**. Next, select and then . NCU deletes the bridge interface.

7 Editing DECnet Parameters

The Wellfleet node supports Phase IV DECnet. DECnet parameters consist of:

- ❑ Basic parameters
Basic parameters apply to the entire DECnet router software module.
- ❑ Interface parameters
Interface parameters apply to individual DECnet interfaces.

This chapter describes how to access and edit these parameters. The first section provides an overview of DECnet.

7.1 DECnet Overview

A DECnet network is partitioned into distinct non-overlapping areas. You assign each area a unique ID number, from 1 to 63. You also assign each node within an area a unique ID number, from 1 to 1023.

Routing between nodes is hierarchical and intended to support large high-speed networks. Level 1 routing is routing *within* an area; Level 2 routing is routing *between* areas. The Wellfleet node supports both routing levels.

Routers with the highest priority are *designated routers*. There is one designated router per area on an Ethernet (the DECnet architecture supports multiple areas on a single Ethernet).

The DECnet router functions, as follows:

- ❑ Determines locations of other routers and hosts on the extended network.
- ❑ Builds the network's topology based on this information.
- ❑ Decides on the best route for a particular packet to follow.

DECnet routers periodically transmits *hello* and *topology* messages to determine the locations and characteristics of hosts and other routers on the extended network. These messages contain information that enables the router to develop the network's topology and determine the best route to each end-node on the network.

DECnet routers use a connectivity algorithm to maintain path lengths and a traffic-assignment algorithm to maintain path costs. The router updates its database with the information derived from these algorithms.

When a DECnet router receives a packet, it checks the destination address in the packet's header, as follows:

- ❑ If the packet is intended for a local node, the router returns the packet to the local network.
- ❑ If the packet is intended for a remote node, the router selects a path and then forwards the packet over it.
- ❑ If the packet's destination is not known, or is for some reason unreachable, the router checks the packet's header information to determine whether the packet should be returned to its source address or simply discarded.

DECnet routers provide traffic-control mechanisms, as follows:

Mechanisms	Function
Congestion Control	Limits the number of packets queued for transmission across a circuit. In addition, a router regulates the ratio of the number of packets received directly from the end-communication (application) layer to the number of routed-through packets.
Packet-Lifetime	Limits the number of intermediate nodes that a packet can pass through.
Loop Detection	Limits the number of times a looping packet can arrive.
Node Listener	Ensures that a minimal traffic level is passing through an adjacent node. If there is not a minimum traffic level, the router declares the adjacent node down and reroutes packets around it.
Node Talker	Generates the minimum traffic level.

7.2 Accessing DECnet Parameters

In order to access DECnet parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 7-1 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default parameters in NCU; any changes the network operator makes will affect every node configured thence on.

To access the **DECNET REDIRECTOR** window, select and then . NCU displays the **DECNET REDIRECTOR** window which allows you to edit DECnet parameters (see Figure 7-2).

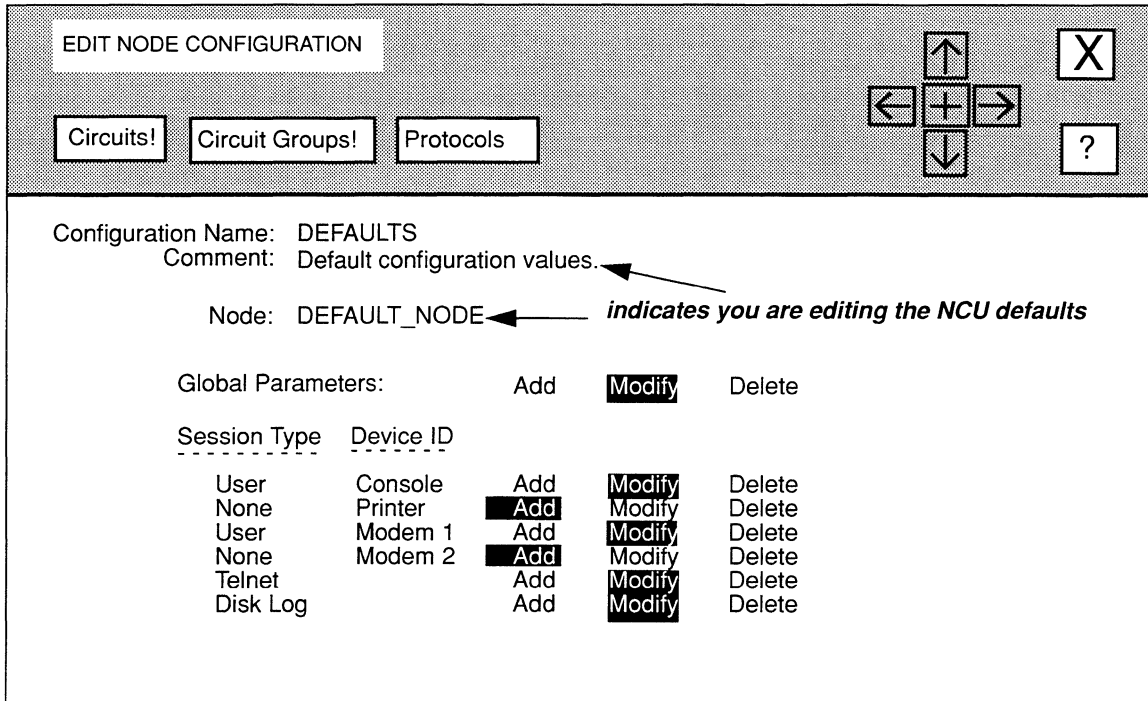


Figure 7-1. EDIT NODE CONFIGURATION Window for Default Settings

7.3 Editing DECnet Basic Parameters

DECnet basic parameters apply to the entire DECnet router, rather than to individual DECnet interfaces on the node. When you connect a node to a network segment that runs DECnet, NCU automatically sets the DECnet basic parameters for the node. This section describes how to modify and delete DECnet basic parameters.

7.3.1 Modifying DECnet Basic Parameters

You modify DECnet basic parameters from the **DECNET REDIRECTOR** window, as follows:

1. **At Auto Enable, specify the state of the DECnet router software when the node boots.**

This DECnet-route-specific **Auto Enable** works in conjunction with the global **Auto Enable** parameter to enable or disable the DECnet-router software module when the node boots, as follows:

- ❑ When global **Auto Enable** is set to **NO**, the DECnet router (and every other application software module) is unconditionally disabled.

You will subsequently need to enable the DECnet router manually with the NCL Interpreter after the node boots.

- ❑ When global **Auto Enable** is set to **YES**, the DECnet router (and every other application software module) is conditionally enabled.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the DECnet router.
- Select **NO** to disable the DECnet router (you will subsequently need to enable the DECnet router manually with the NCL Interpreter after the node boots)

2. At Max. Nodes, select the highest node number residing in the DECnet.

NCU provides responses that range from **31** to **1023**.

Note

You must configure all Wellfleet nodes in the DECnet to have the same **Max Nodes** value.

3. At Max. Area, select the highest area ID number in the DECnet.

NCU provides responses that range from **1** to **63**.

Note

You must configure all Wellfleet nodes in the DECnet with the same **Max. Area** value.

4. At Area, enter the local area's DECnet ID number.

5. At Node, enter the node's DECnet ID number.

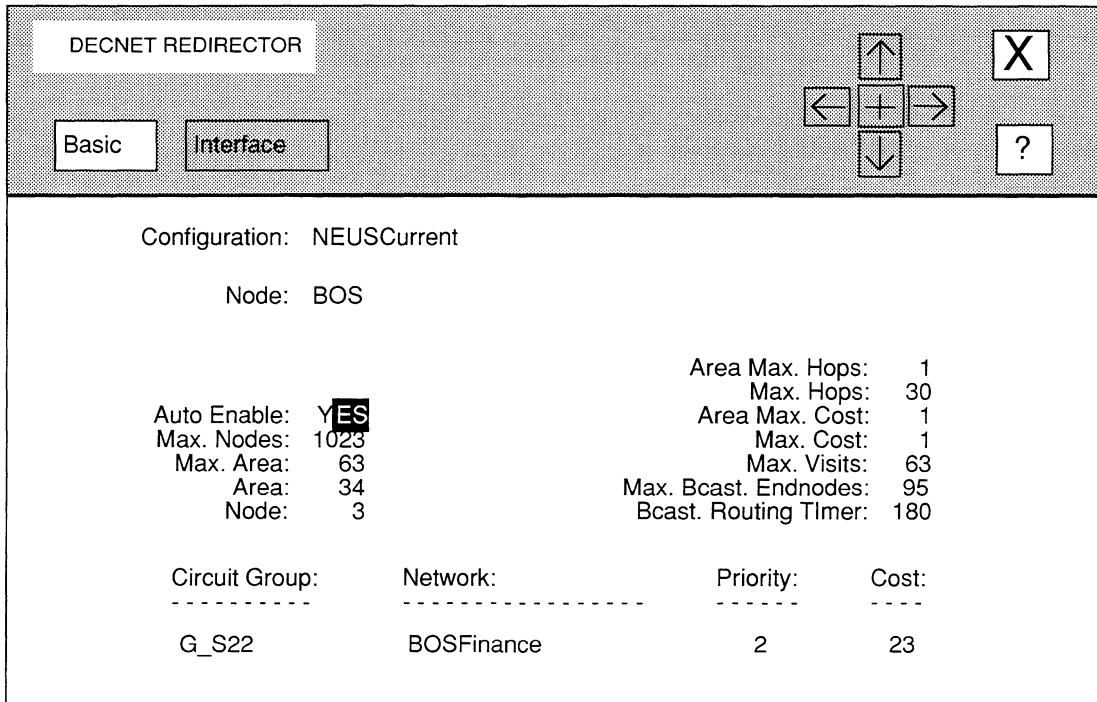


Figure 7-2. DECNET REDIRECTOR Window

- At **Area Max. Hops**, enter the maximum number (from 1 to 30) of DECnet areas a packet can pass through before it reaches the area containing its destination end station.

In a large network, there are frequently multiple paths to a single destination area; therefore, the number of routers a packet has to pass through to reach a particular destination may vary depending on the path the packet follows. Longer paths mean more routers for the packet to transit, potentially increasing the packet's travel time. Based on your network's topology, determine the longest acceptable path for a packet to follow, then count the number of DECnet routers in this path and enter the result at **Area Max. Hops**.

- At **Max. Hops**, enter the maximum cost (a number from 1 to 1008) of a path to any node on the network.

Note

Ensure that **Max. Hops** is equal to or greater than **Area Max. Cost**.

8. At Area Max. Cost, enter the maximum path cost (a number from 1 to 1008) to any area in the network.

DECnet determines path costs based on the sum of individual circuit costs. Typically, you assign circuit costs to reflect the speed of the transmission media: low costs reflect high-speed media; high costs reflect slower media. Table 7-1 lists suggested DECnet circuit costs for transmission media of various clock speeds.

For example, the recommended cost for an Ethernet circuit is 3, and the recommended cost of a full T1 circuit is 7. If a path traverses a T1 circuit followed by an Ethernet circuit, the path cost is 10. Figure 7-3 depicts another example. In the figure, the cost from **A** to **C** by way of **B** equals **10**, while the cost from **A** to **C** by way of **D** equals **22**.

Table 7-1. Suggested DECnet Circuit Costs

Transmission Speed	Circuit Cost
100Mb/sec	1
16Mb/sec	2
10Mb/sec	3
4Mb/sec	5
1.54Mb/sec	7
1.25Mb/sec	8
833Kb/sec	9
625Kb/sec	10
420Kb/sec	11
230.4Kb/sec	12
125Kb/sec	13
64Kb/sec	14
56Kb/sec	15
38.4Kb/sec	16
32Kb/sec	17
19.2Kb/sec	18
9.6Kb/sec	19
7.2Kb/sec	20
4.8Kb/sec	21
2.4Kb/sec	22
1.2Kb/sec	25

The DECnet router always selects the circuit(s) with the lowest cost when defining a path, so assigning each circuit a cost is, in effect, a way of assigning it a priority. If you do not want this circuit to be used on a regular basis, assign it a high cost.

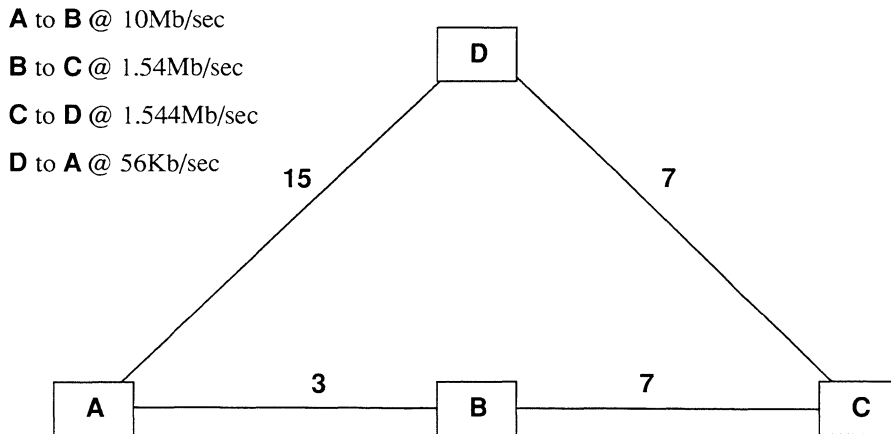


Figure 7-3. Sample DECnet Circuit Costs

9. At **Max. Cost**, enter the maximum cost (a number from 1 to 1008) of a path to any node on the network.

Note

Ensure that **Max. Cost** is equal to or greater than **Area Max. Cost**.

10. At **Max. Visits**, enter the number of times (from 1 to 63) a packet can pass through the DECnet router.

Note

Ensure that **Max. Visits** is greater than or equal to **Max. Hops**.

Max. Visits is a packet-lifetime control mechanism — by limiting the number of times a packet can pass through the DECnet router, you prevent a corrupted packet, or a packet whose destination stations has somehow become unreachable, from continuously traveling through the network

11. **At Max. Bcast Endnodes, select the response that equals the maximum number of nodes (based on your network topology) adjacent to a DECnet router on an Ethernet.**

NCU provides values that range from 1 to 1023.

Note

The higher the value you set for **Max. Bcast Endnodes**, the more you impact network performance and memory utilization.

12. **At Bcast. Routing Timer, select the response that equals the maximum number of seconds between routing topology messages issued by the router.**

NCU provides values that range from 15 to 180, in increments of 15 seconds.

13. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

DECnet Basic Parameters Stored.

7.3.2 Deleting DECnet Basic Parameters

You delete DECnet basic parameters from the **DECNET REDIRECTOR** window. Simply select and then . NCU displays this window, press **[RETURN]** to clear it from the console:

Press return when done.

DECnet redirector deleted.

7.4 Configuring DECnet Interfaces

You configure each DECnet interface individually. The following sections describe how to change and delete DECnet interfaces.

7.4.1 Modifying DECnet Interfaces

You modify DECnet interfaces, as follows:

1. **Select the interface you wish to modify under Circuit Group in the DECNET REDIRECTOR window.**

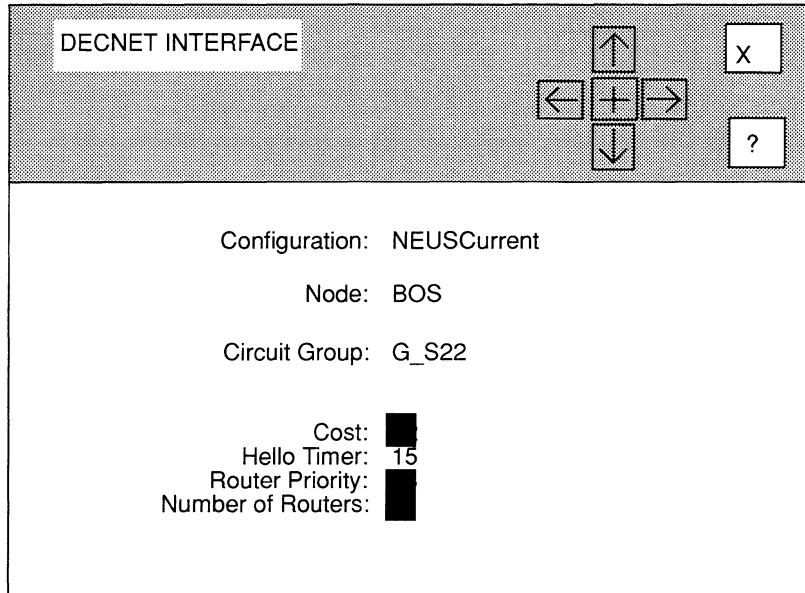


Figure 7-4. DECNET INTERFACE Window

2. Select and then .

NCU displays the **DECNET INTERFACE** window (see Figure 7-4) for that interface, which allows you to configure interface-specific parameters.

3. **At Cost, enter the relative cost (a number from 1 to 25) of the circuit group.**

DECnet determines path costs based on the sum of individual circuit costs. Typically, you assign circuit costs to reflect the speed of the transmission media: low costs reflect high-speed media; high costs reflect slower media. Table 7-1 lists suggested DECnet circuit costs for transmission media of various clock speeds.

The DECnet router always selects the circuit(s) with the lowest cost when defining a path, so assigning each circuit a cost is, in effect, a way of assigning it a priority. If you do not want this circuit to be used on a regular basis, assign it a high cost.

Note

Ensure you set **Cost** in line with **Area Max. Cost** and **Max. Cost**.

4. **At Hello Timer, select the response that equals the number of seconds between DECnet hello messages transmitted across the circuit group.**

NCU provides the following values: 15, 30, 45, 60, 600, 1800, 2400, and 3600.

5. **At Router Priority, enter the priority (a number from 1 to 127) you wish to assign to the DECnet router.**

On any Ethernet segment, the router with the highest **Router Priority** is the designated router. If two routers on a segment share the same priority, the router with the highest node number is the designated router.

6. **At Number of Routers, enter the number of DECnet routers (from 1 to 33) associated with this circuit group.**

Base your response on your network topology.

7. Select and then Save .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

DECnet Interface Parameters Stored.

NCU returns to the DECNET REDIRECTOR window; if you changed **Router Priority** or **Cost** for the interface, the **DECNET REDIRECTOR** window displays the new information.

Repeat this procedure for each additional DECnet interface you wish to modify. When you have modified all DECnet interfaces for the node, select and then Confirm in the **DECNET REDIRECTOR** window to return to the **EDIT NODE CONFIGURATION** window for the node.

7.4.2 Deleting DECnet Interfaces

You delete DECnet interfaces from the **DECNET REDIRECTOR** window. First, select the interface under **Circuit Group**. Next, select Interface and then Delete. NCU deletes the DECnet interface.

8 Editing XNS Parameters

XNS parameters consist of:

- ❑ Basic parameters
Basic parameters apply to the entire XNS router software module,
- ❑ Interface parameters
Interface parameters apply to individual XNS router interfaces.
- ❑ Static-route parameters
Static-route parameters apply to user-specified transmission paths. You configure static routes when you want to restrict the paths that packets can follow to paths you specifically define. Static routes cannot be overwritten.

This chapter describes how to access and edit these parameters. The first section provides an overview of XNS.

8.1 XNS Overview

The Xerox Network Systems Internet Transport Protocols (XNS) are a suite of internet protocols developed by the Xerox Corporation. Originally developed for internets of connected Ethernets, XNS traffic can be delivered over Ethernets and routed across HDLC or LAPB point-to-point lines.

XNS is similar to the internet TCP/IP protocol suite in basic functionality, but differs in internet packet format. The Routing Information Protocol defined for TCP/IP networking in Request for Comments (RFC) 1058, is based upon the XNS routing protocol, RIP.

XNS is based on a five-level protocol architecture (see Figure 8-1). XNS Level 1, which corresponds to the network layer of the International Organization for Standardization (ISO) reference model, concerns itself with internet packet format, internet addressing, and internet routing. XNS Level 2, which corresponds to the ISO transport layer, concerns itself with interprocess communication primitives and provides a “best-effort” connectionless delivery service.

Note

The Wellfleet node fully supports XNS Level 1, the Internet Datagram Protocol (IDP), and XNS Level 2 Echo, Error, and Routing Information Protocols. XNS routers do not participate in Level 2 Sequenced Packet and Packet Exchange Protocols, but do route such traffic through an XNS internet.

For specific technical information on XNS refer to *Internet Transport Protocols* (Xerox Corporation; Xerox System Integration Standard; Stamford, Connecticut; December, 1981; XSIS-028112).

When you configure the XNS router, you provide information that it uses to route packets through an XNS internet. The XNS internet packet header (see Figure 8-2) consists of 30 bytes of address and delivery data, as follows:

Bytes	Contain
1 & 2	Checksum of the entire XNS packet. A value of FFFF (hexadecimal) in bytes 1 and 2 of the packet header indicate that checksumming has been disabled. Later, as you proceed through the configuration process, you will enable or disable checksumming on a per interface basis.
3 & 4	Internet packet length. The packet originator supplies the packet length (which varies from 30 to 576 bytes).
5	Transport-control mechanism that the XNS router uses. The least significant four bits (bits 0 to 3) are unused and are always set to 0. The most significant four bits (bits 4 to 7) contain a hop count that enumerates the number of routers encountered in a packet's transit from source to destination. Each router that forwards an internet packet to another intervening router increments the hop count by one. A packet that reaches its 16th router is discarded.
6	Coded value that identifies the protocol carried in the packet's data section. Typically, the packet originator supplies the protocol type. Table 8-1 lists common XNS protocol type values.

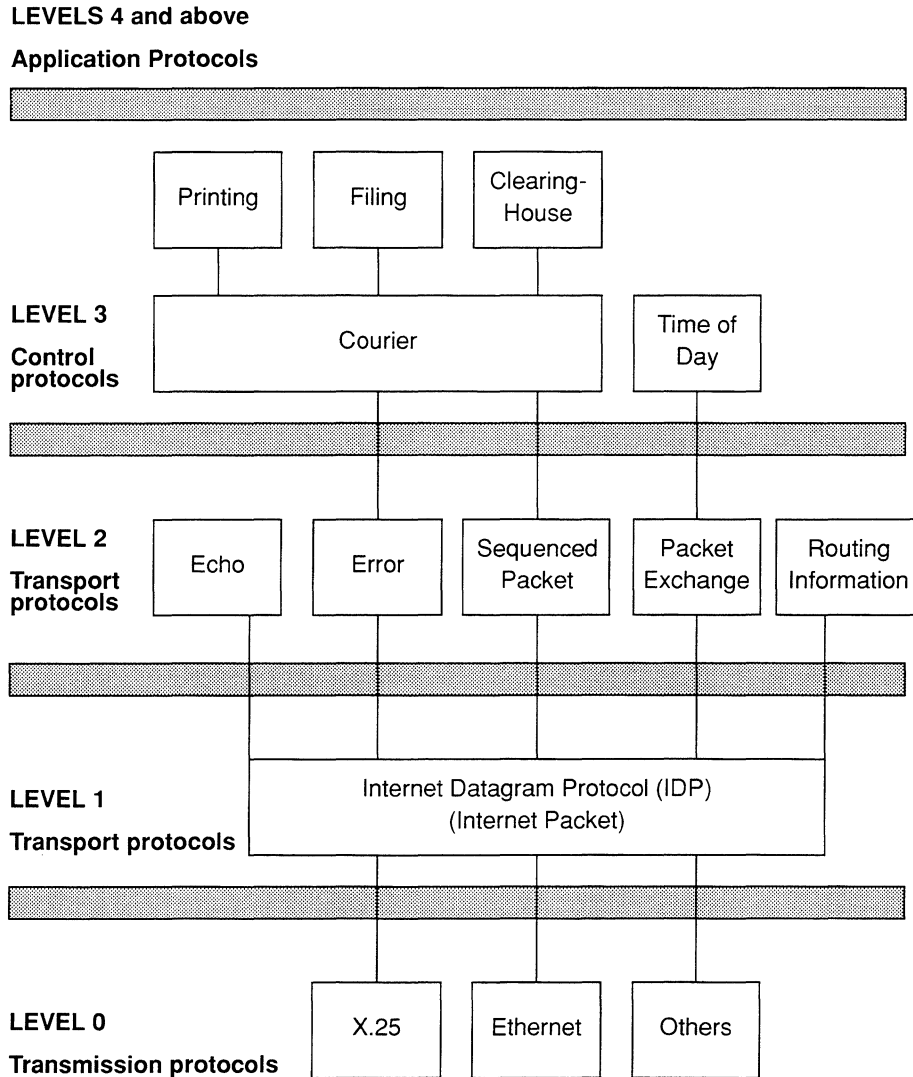


Figure 8-1. XNS Protocol Architecture

Bytes (continued)	Contain
7, 8, 9, & 10	Destination network number. The packet originator supplies the destination network number. Under XNS, a local administrator assigns all networks a network number. If the network number is less than 32 bits in length, the network number is written in the least-significant portion of the destination-network field, and the unused most-significant bits are set to 0.
11 through 16	Destination host's physical address (also called host number). The packet originator supplies the destination physical address. Under XNS, all hosts are assigned a unique physical address. This address can be the universally-administered 48-bit Ethernet address, or it can be a locally-assigned address of arbitrary length (but less than 48 bits). If the host address is less than 48 bits in length, it is written in the least-significant portion of the destination-host field and the unused most-significant bits are set to 0.
17 & 18	Destination socket (a local address identified with a special software service). The packet originator supplies the destination socket. XNS reserves well-known sockets for use by certain protocols. Table 8-2 lists these well-known socket numbers.
19, 20, 21, & 22	Source-network number (supplied by the packet originator). If the network-number length is less than 32 bits, it is written in the least-significant portion of the source-network field, and the unused most-significant bits are set to 0.
23 through 28	Source-host's physical address (supplied by the packet originator). If the physical-address length is less than 48 bits, it is written in the least-significant portion of the source-host field and the unused most-significant bits are set to 0.
29 & 30	Source socket. The packet originator supplies the source socket.

Table 8-1. XNS Packet-Type Values

Packet Type Value	Protocol
0	Unknown
1	Routing-Information Protocol
2	Echo Protocol
3	Error Protocol
4	Sequenced-Packet Protocol

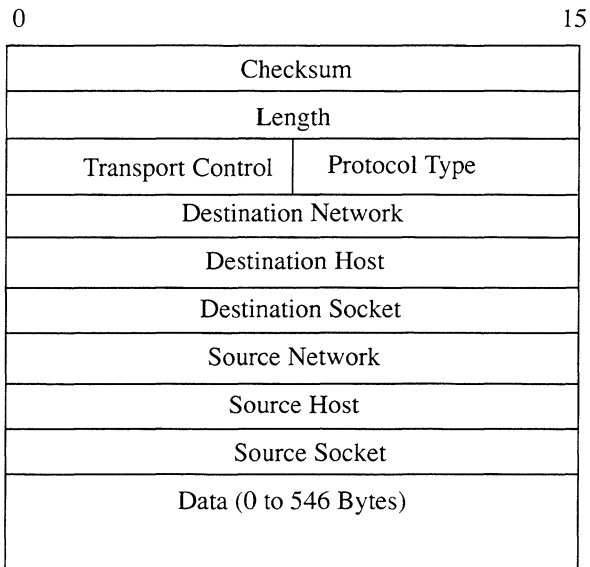


Figure 8-2. XNS Internet Packet Format

Table 8-2. XNS Well-Known Sockets

Packet Type Value	Protocol
0	Unknown
1	Routing-Information Protocol
2	Echo Protocol
3	Error Protocol

8.2 Accessing XNS Parameters

In order to access XNS parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 8-1 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default parameters in NCU; any changes the network operator makes will affect every node configured thence on.

To access the **XNS REDIRECTOR** window, select **Protocols** and then **XNS**. NCU displays the **XNS REDIRECTOR** window which allows you to edit XNS parameters (see Figure 8-4).

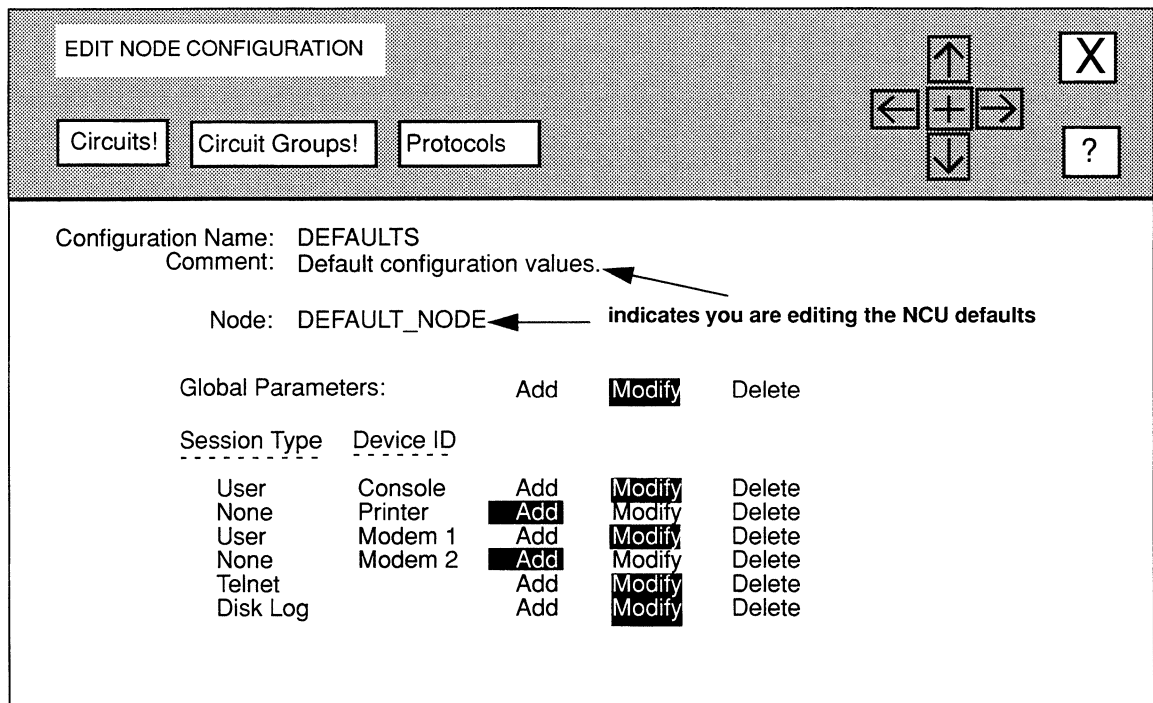


Figure 8-3. EDIT NODE CONFIGURATION Window for DEFAULT_NODE

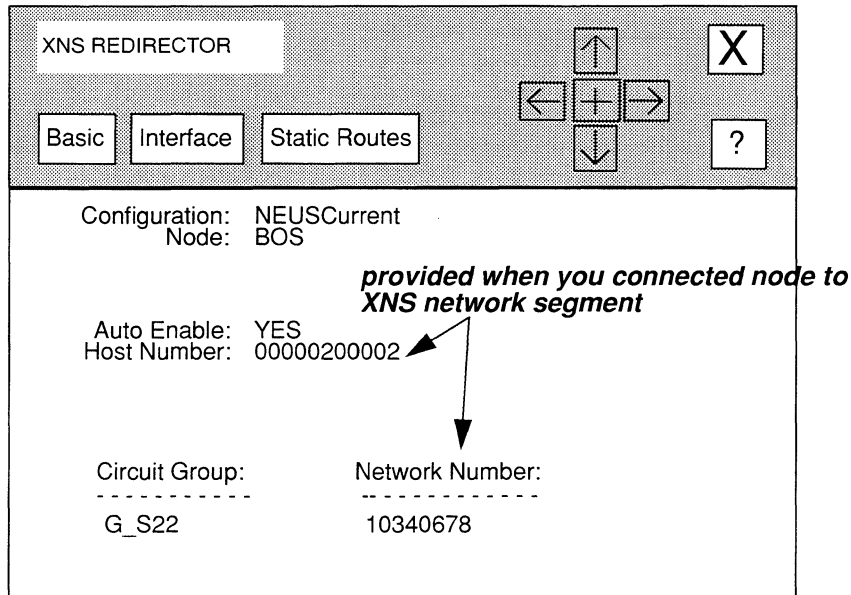


Figure 8-4. XNS REDIRECTOR Window

8.3 Editing XNS Basic Parameters

XNS basic parameters apply to the entire XNS router, rather than to individual XNS interfaces on the node. When you connect a node to a network segment that runs XNS, NCU automatically sets the XNS basic parameters for the node. This section describes how to modify and delete XNS basic parameters.

8.3.1 Modifying XNS Basic Parameters

You modify XNS basic parameters from the **XNS REDIRECTOR** window, as follows:

1. **At Auto Enable, select the state of the XNS router software when the node boots.**

This XNS-router-specific **Auto Enable** works in conjunction with the global **Auto Enable** parameter to enable or disable the XNS-router software module when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the XNS router (and every other application software module) is unconditionally disabled.

You will subsequently need to enable the XNS router manually with the NCL Interpreter after the node boots.

- ❑ When global **Auto Enable** is set to **YES**, the XNS router (and every other application software module) is conditionally enabled.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the XNS router.
- Select **NO** to disable the XNS router (you will subsequently need to enable the XNS router manually with the NCL Interpreter after the node boots).

2. At **Host Number**, enter the XNS host address in 12-digit hexadecimal format (pad with leading zeros if necessary).

Note

You set **Host Number** when you connected the node to the XNS network segment.

3. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

XNS Basic Parameters Stored.

8.3.2 Deleting XNS Basic Parameters

You delete XNS basic parameters from the **XNS REDIRECTOR** window. Simply select and then . NCU displays this window, press **[RETURN]** to clear it from the console:

Press return when done.

XNS redirector deleted.

8.4 Configuring XNS Interfaces

You configure each XNS interface individually. The following sections describe how to modify and delete XNS interfaces.

8.4.1 Modifying XNS Interfaces

You modify XNS interfaces, as follows:

1. Select the interface you wish to modify under **Circuit Group in the XNS REDIRECTOR window**.

The screenshot shows a window titled "XNS INTERFACE" with a close button (X) and a help button (?). The configuration details are as follows:

```

Configuration:  NEUSCurrent
Node:          BOS
Circuit Group: G_S22

Network Number: 10340678 ← provided when you connected node
RIP Supply:    YES      to rXNS network segment
RIP Cost:      1
Checksum:     YES
  
```

Figure 8-5. XNS INTERFACE Window

2. Select and then .

NCU displays the **XNS INTERFACE** window for that interface (see Figure 8-5), which allows you to configure interface-specific parameters.

3. At **Network Number**, enter the locally-assigned network number in 8-digit hexadecimal value (pad with leading zeros, if necessary).

Note

You set **Network Number** when you connected the node to the XNS network segment.

4. At **RIP Supply**, enable or disable the **RIP Supply** function.

YES Enables the RIP Supply function so that the XNS router transmits periodic RIP update across the circuit group.

NO Disables the RIP Supply function.

5. At **RIP Cost**, enter the cost (a number from 1 to 15) for each router hop.

Standard XNS RIP implementations assign a cost of 1 to each hop. Keep in mind, that if you increase the cost, you will reach the value of 16 (at which XNS declares a destination unreachable) more quickly.

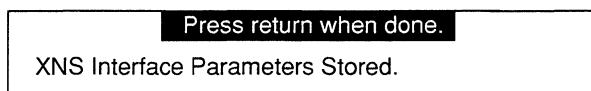
6. **At Checksum, enable or disable the checksum processing.**

YES Enables checksumming. With checksumming enabled, XNS checksums the entire internet packet; the router verifies the internet packet checksum on packet arrival, and generates a new checksum value upon packer relay.

NO Disables checksumming.

7. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.



NCU returns to the **XNS REDIRECTOR** window. Repeat this procedure for each additional XNS interface you wish to modify.

8.4.2 Deleting XNS Interfaces

You delete XNS interfaces from the **XNS REDIRECTOR** window. First, select the interface under **Circuit Group**. Next, select and then . NCU deletes the XNS interface.

8.5 Configuring Static Routes

Static routes are user-specified transmissions paths. You configure static routes when you want to restrict the paths that packets can follow to paths you specifically define. Like routes learned through RIP, static routes are listed in the XNS routing table. Unlike routes learned through RIP, static routes cannot be overwritten.

You configure static routes from the **XNS STATIC ROUTES** window. To display this window, select in the **XNS REDIRECTOR** window (see Figure 8-6). You may now add, update, and delete static routes.

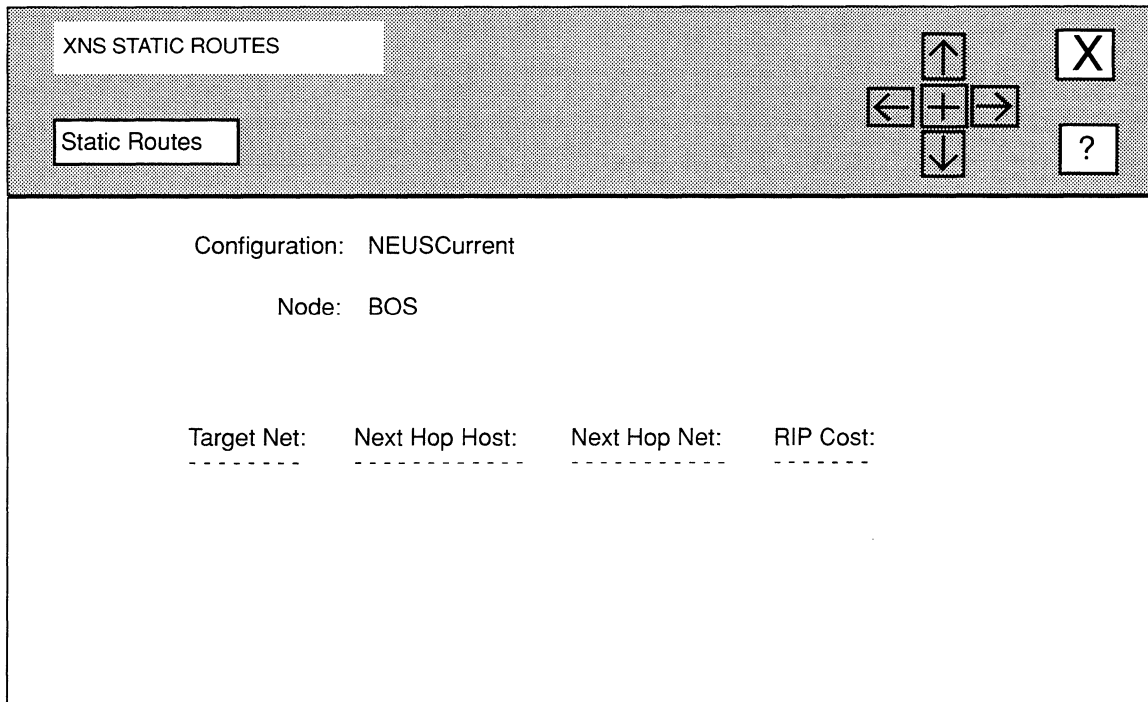


Figure 8-6. XNS STATIC ROUTES Window

8.5.1 Adding Static Routes

You add static routes from the **XNS STATIC ROUTES** window (see Figure 8-6), as follows:

1. Select **Static Routes** and then **Add**.
2. At **Target Net**, enter the network number of the destination network.
3. At **Next Hop Host**, enter the host address of the next-hop router used to reach **Target Net**.
4. At **Next Hop Net**, enter the network address of the next-hop router.

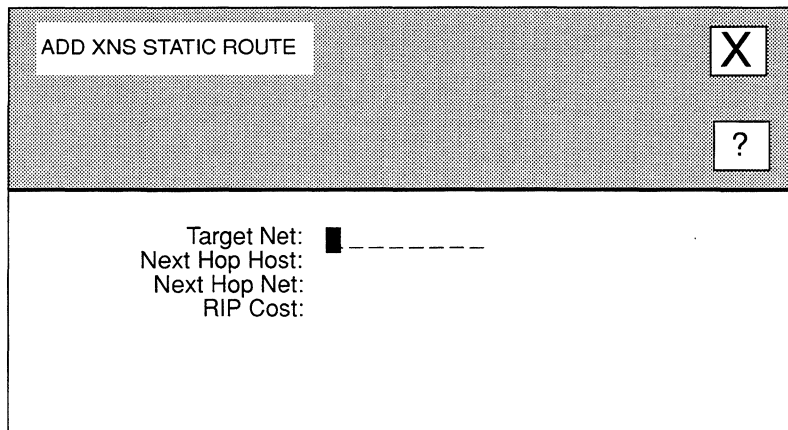


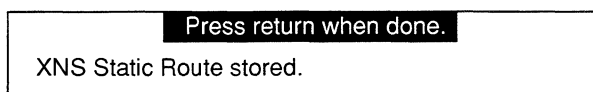
Figure 8-7. ADD XNS STATIC ROUTE Window

5. At **RIP Cost**, enter the cost (a number from 1 to 15) for the relay to **Target Net**.

If you set the interface-specific **RIP COST** to 1, simply enter the number of hops to **Target Net**. Otherwise, enter the value you obtain when you total the individual **RIP costs** of each hop to **Target Net** — the value you enter can be no greater than 15 (XNS declares a destination unreachable that has a cost of 16).

6. Select **X** and then **Save**.

NCU displays this window; press **[RETURN]** to clear it from the console.



NCU returns to the **XNS STATIC ROUTES** window, which now displays the static route you just added. Repeat this procedure for each additional XNS static route you wish to add.

8.5.2 Updating Static Routes

You update static routes from the **XNS STATIC ROUTES** window. First, select the static route you wish to update under **Target Net**. Next, select **Static Routes** and **Update**. NCU displays the **ADD XNS STATIC ROUTE** window displaying the current parameter settings for that static route. From this point on, updating a static route is the same as adding a static route, see *Section 8.5.1, Adding Static Routes* for information on parameter settings.

8.5.3 Deleting Static Routes

You delete static routes from the **XNS STATIC ROUTES** window. First, select the static route you wish to delete under **Target Net**. Next, select and . NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

XNS Static Route deleted.

9 Editing IPX Parameters

IPX parameters consist of:

- ❑ Basic parameters
Basic parameters apply to the entire IPX router software module.
- ❑ Interface parameters
Interface parameters (for example, *Service Advertising Protocol*, *SAP*, *filters* and *NetBIOS static routes*) apply to individual IPX interfaces.
- ❑ Static-route parameters
Static-route parameters apply to user-specified transmission paths. You configure static routes when you want to restrict the paths that packets can follow to paths you specifically define. Static routes cannot be overwritten.

This chapter describes how to access and edit these parameters. The first section provides an overview of IPX.

9.1 IPX Overview

The Internet Packet Exchange Protocol (IPX) is the Novell, Inc. implementation of XNS. Generally found in PC and/or workstation environments, IPX supports a wide variety of LAN topologies and media.

IPX uses the XNS internet-packet format. Like XNS, IPX makes a “best effort” to deliver internet packets, but does not guarantee delivery. IPX requires that specific applications build upon the basic IPX-packet protocol to implement reliable-delivery and other higher-level interconnection protocols.

The Wellfleet node supports the *Service Advertising Protocol* (SAP), which enables network-service programs to promulgate their services throughout an IPX-based internet. SAP requires that such services periodically broadcast their identity and service type. The IPX router uses the information in these periodic broadcasts to construct and maintain a table of available services and their locations within the IPX-internet. The IPX router refers to this table in order to reply to host queries about the availability of specific services. *Section 9.1.1, Service Advertising Protocol* provides additional information about SAP.

For specific technical information on IPX refer to *Advanced Netware, V2.0 Internet Packet Exchange Protocol (IPX) with Asynchronous Event Scheduler* (Novell, Inc., Specifications as of March 19, 1986).

When you configure the IPX router, you provide information that it uses to route packets through an IPX internet. The IPX internet packet header (see Figure 9-1) consists of 30 bytes of address and delivery data, as follows:

Bytes	Contain
1 & 2	Checksum of the 30-byte IPX header. A value of FFFF (hexadecimal) in bytes 1 and 2 of the packet header indicate that checksumming has been disabled.
3 & 4	Internet packet length. The packet originator supplies the packet length (which varies from 30 to 576 bytes).
5	Transport-control mechanism that the IPX router uses. The least significant four bits (bits 0 to 3) are unused and are always set to 0. The most significant four bits (bits 4 to 7) contain a hop count that enumerates the number of routers encountered in a packet's transit from source to destination. Each router that forwards an internet packet to another intervening router increments the hop count by one. A packet that reaches its 16th router is discarded.
6	Coded value that identifies the protocol carried in the packet's data section. Typically, the packet originator supplies the protocol type. Table 9-1 lists common IPX protocol type values.
7, 8, 9, & 10	Destination network number. The packet originator supplies the destination network number. Under IPX, a local administrator assigns all networks a network number. If the network-number length is less than 32 bits, the network number is written in the least-significant portion of the destination-network field, and the unused most-significant bits are set to 0.

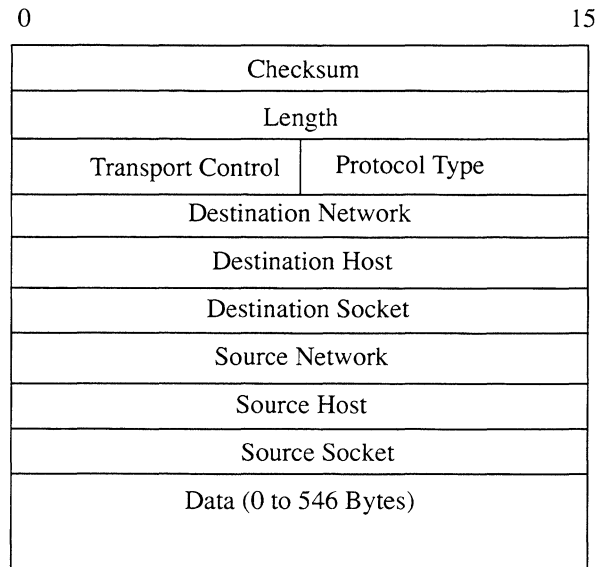


Figure 9-1. IPX Internet Packet Format

Bytes(continued)	Contain
11 through 16	<p>Destination host's physical address (also called host number).</p> <p>The packet originator supplies the destination physical address. Under IPX, all hosts are assigned a unique physical address. This address can be the universally-administered 48-bit Ethernet address, or it can be a locally-assigned address of arbitrary length (but less than 48 bits). If the host-address length is less than 48 bits, it is written in the least-significant portion of the destination-host field and the unused most-significant bits are set to 0.</p>
17 & 18	<p>Destination socket (a local address identified with a special software service).</p> <p>The packet originator supplies the destination socket. IPX reserves well-known sockets for use by certain protocols. Table 9-2 lists these well-known socket numbers.</p>

Bytes(continued)	Contain
19, 20, 21, & 22	<p>Source-network number (supplied by the packet originator).</p> <p>If the source-network number length is less than 32 bits, it is written in the least-significant portion of the source-network field, and the unused most-significant bits are set to 0.</p>
23 through 28	<p>Source-host's physical address (supplied by the packet originator).</p> <p>If the physical-address length is less than 48 bits, it is written in the least-significant portion of the source-host field and the unused most-significant bits are set to 0.</p>
29 and 30	<p>Source socket.</p> <p>The packet originator supplies the source socket.</p>

Table 9-1. IPX Packet-Type Values

Packet Type Value	Protocol
0	Unknown
4	Sequenced-Packet Protocol

9.1.1 Service Advertising Protocol

Service Advertising Protocol (SAP) enables network-resident value-added servers to inform clients of their presence. SAP provides identification-broadcasting services that the server uses to make itself known to clients by name, type, and IPX network address. After initialization, servers broadcast service-advertising packets every 60 seconds. All routers within the IPX network receive service-advertising packets.

These packets identify the server by:

- Name
A network-unique character string up to 48 bytes in length.
- Server type
A 16-bit service type identifier administered by Novell.
- Network address
Consists of network, host, and socket identifiers.

Table 9-2. IPX Well-Known Sockets

Packet Type Value	Protocol
0	Unknown
0x451	File Server
0x452	SAP
0x453	RIP
0x455	NetBIOS (see <i>Section 9.1.2, NetBios</i>)

IPX routers maintain a small database (called a bindery) that contains the following server-specific information: name, type, address, hop count, the interface to the server, a timer value to age bindery entries. Each time a router receives a service-advertising packet, it compares the packet's contents with its bindery:

- ❑ If the bindery contains information on this service:
The router simply refreshes the age timer.
- ❑ If the packet contents advertise a previously-unknown service:
The router adds a new entry to its bindery and triggers an advertisement of the new service to all connected networks.

IPX routers also issue regularly scheduled advertisements of their bindery. These advertisements, issued at 60-second intervals, propagate server tables throughout the IPX network.

Clients can use the IPX broadcast facility to obtain information on network servers. Client information requests take two forms: general-service queries seek information from all network servers; nearest-service queries seek information on the closest service on a specified type.

9.1.2 NetBIOS

NetBIOS (Network Basic Input/Output System) is a session-layer protocol developed by Sytek, Inc. for IBM PC networks. NetBIOS has been widely implemented among other vendors, including Novell.

A session is a logical connection between two devices (often workstations) that must be established prior to any communication. NetBIOS establishes that prerequisite logical connection.

9.2 Accessing IPX Parameters

In order to access IPX parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 9-2 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default parameters in NCU; any changes the network operator makes will affect every node configured thence on.

To access the **IPX REDIRECTOR** window, select **Protocols** and then **IPX**. NCU displays the **IPX REDIRECTOR** window which allows you to edit IPX parameters (see Figure 9-3).

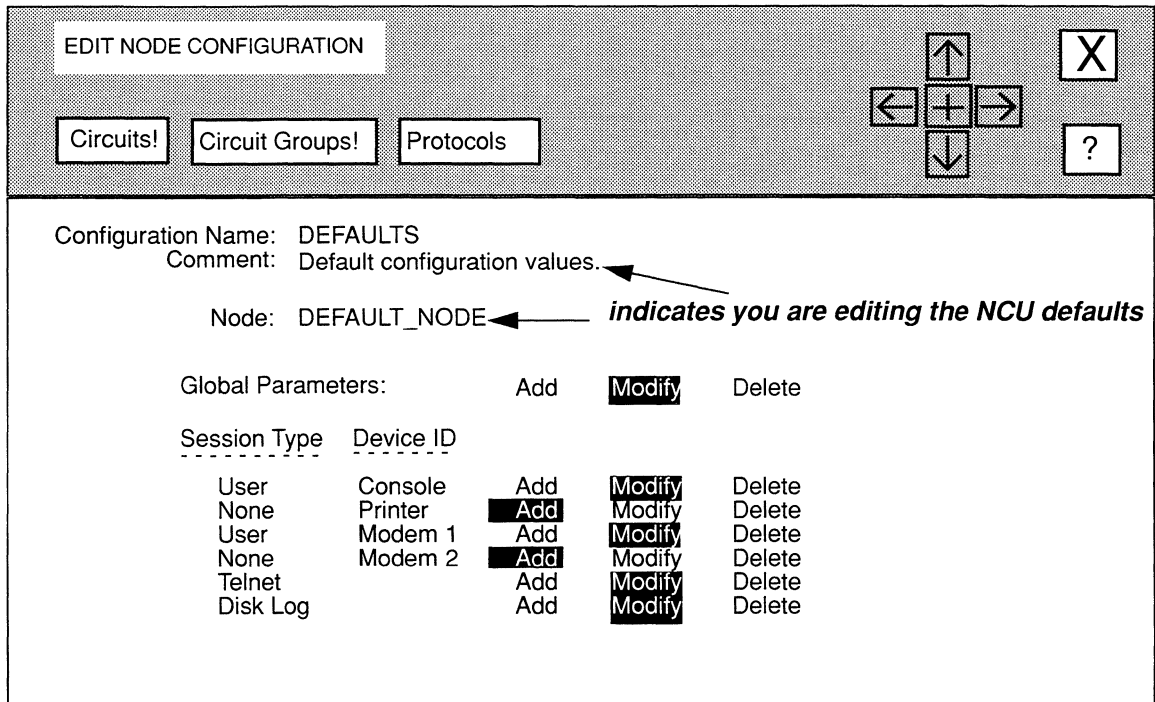


Figure 9-2. EDIT NODE CONFIGURATION Window for DEFAULT_NODE

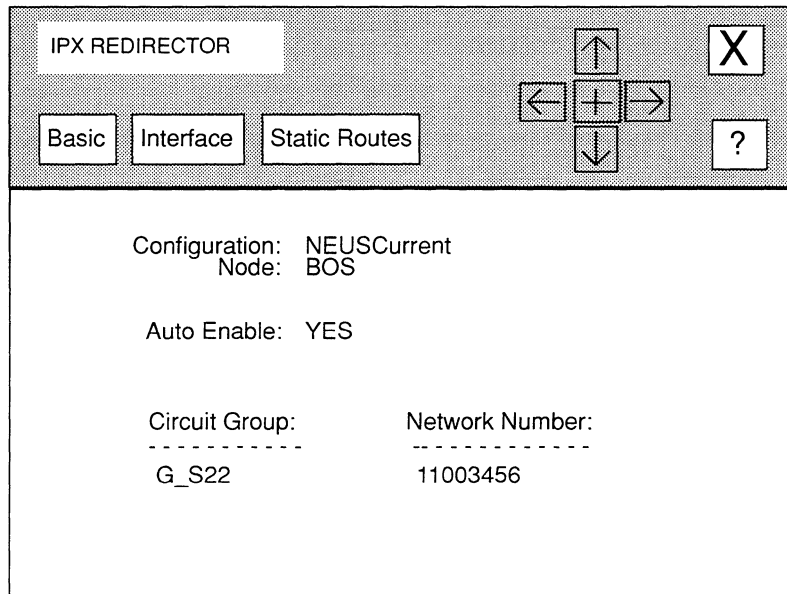


Figure 9-3. IPX REDIRECTOR Window

9.3 Editing IPX Basic Parameters

IPX basic parameters apply to the entire IPX router, rather than to individual IPX interfaces on the node. When you connect a node to a network segment that runs IPX, and activate the default settings, NCU automatically sets the IPX basic parameters for the node. This section describes how to modify and delete IPX basic parameters.

9.3.1 Modifying IPX Basic Parameters

Modifying IPX basic parameters consists of resetting the **Auto Enable** parameter in the **IPX REDIRECTOR** window. This IPX-route-specific **Auto Enable** works in conjunction with the global **Auto Enable** parameter to enable or disable the IPX-router software module when the node boots, as follows:

- When global **Auto Enable** is set to **NO**, the IPX router (and every other application software module) is unconditionally disabled.

You will subsequently need to enable the IPX router manually with the NCL Interpreter after the node boots.

- ❑ When global **Auto Enable** is set to **YES**, the IPX router (and every other application software module) is conditionally enabled.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the IPX router.
- select **NO** to disable the IPX router (you will subsequently need to enable the IPX router manually with the NCL Interpreter after the node boots).

Select and then . NCU displays this window; press **[RETURN]** to clear it from the console.

Press return when done.

XNS Basic Parameters Stored.

9.3.2 Deleting IPX Basic Parameters

You delete IPX basic parameters from the **IPX REDIRECTOR** window. Simply select and then . NCU displays this window, press **[RETURN]** to clear it from the console:

Press return when done.

IPX redirector deleted.

9.4 Configuring IPX Interfaces

You configure each IPX interface individually. The following sections describe how to modify and delete IPX interfaces.

9.4.1 Modifying IPX Interfaces

You modify IPX interfaces, as follows:

1. **Select the interface you wish to modify under Circuit Group in the IPX REDIRECTOR window.**
2. Select and then .

NCU displays the **IPX INTERFACE** window for that interface (see Figure 9-4), which allows you to configure interface-specific parameters.

IPX INTERFACE

Filters

Configuration: NEUSCurrent
Node: BOS
Circuit Group: G_S21

Network Number: 11003456

CSMA/CD Encapsulation: ETHERNET
WAN SAP Period (mins): 1
Accept NETBIOS Broadcasts: YES
Deliver NETBIOS Broadcasts: YES

Supply RIP Updates: YES
RIP Interface Cost: 1

Figure 9-4. IPX INTERFACE Window

3. At **Network Number**, enter the IPX network number (as an 8-digit hexadecimal value padded with leading zeros, if necessary) of the network to which the interface connects.

Note

You set **Network Number** when you connected the node to the IPX network segment.

4. At **CSMA/CD Encapsulation**, select the encapsulation method for IEEE 802.3 media (this parameter has no effect on other media types).

Note

You set **CSMA/CD Encapsulation** when you connected the node to the IPX network segment.

NCU provides three responses:

- ETHERNET**..... Sets Ethernet 2.0 encapsulation. Ethernet encapsulation (see Figure 9-5) prefixes an 8-octet preamble, 6 octets of destination-address information, 6 octets of source-address information, and 2 octets of protocol type information (hexadecimal 8137) to the IPX packet. It appends a 4-octet frame check sequence to the packet.

- NOVELL**..... Sets Novell proprietary encapsulation. Novell encapsulation (see Figure 9-6) prefixes an 8-octet preamble, 6 octets of destination-address information, 6 octets of source-address information, and 2 octets of packet-length information to the unchecksummed IPX packet. It appends a 4-octet frame check sequence to the packet.

- 802.2**..... Sets IEEE 802.2 logical-link control encapsulation. 802.2 encapsulation (see Figure 9-7) prefixes 1 octet of destination-service access-point identification, 1 octet of source-service access-point identification, and 1 octet of control information to the IPX packet. The 802.2 packet, in turn, will be encapsulated within a media-specific packet.

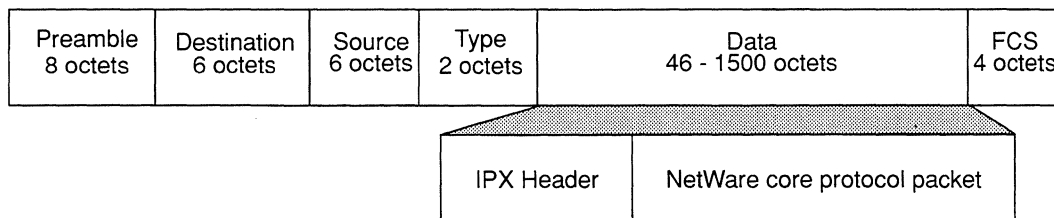


Figure 9-5. Ethernet Encapsulation

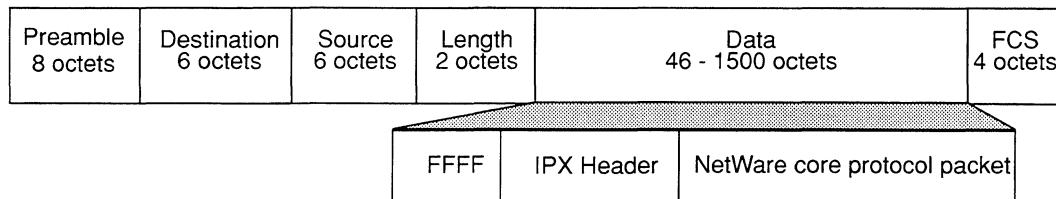


Figure 9-6. Novell Proprietary Encapsulation

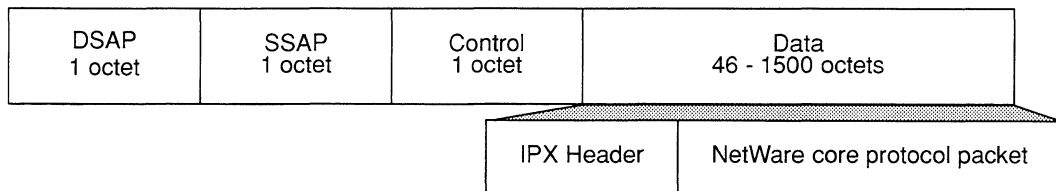


Figure 9-7. 802.2 Encapsulation

- At **WAN SAP Period (mins)**, enter the interval (from 1 to 99 minutes) at which the IPX router transmits General Server Responses (GSRs) across a point-to-point link (V.35, T1, E1, etc).

Note

Enter **0** to disable GSR transmission. You should disable GSR transmission with great care; the loss of a single SAP advertisement can result in unsynchronized binderies at both ends of the link.

GSRs are unsolicited SAP advertisements across point-to-point links. You can reduce the bandwidth that the SAP protocol consumes by setting **WAN SAP Period (mins)** to a low GSR-transmission frequency.

If the interface provides a LAN connection (Ethernet, IEEE 802.3, token ring, or FDDI) enter **1** at **WAN SAP Period (mins)**; the IPX router always transmits GSRs to such media at one minute intervals.

Note

WAN SAP Period (mins) has no effect on triggered SAP advertisements generated in response to bindery changes, or on advertisements generated in response to client requests.

6. **At Accept NETBIOS Broadcasts, enable or disable “local” client access to remote NetBIOS servers.**

YES Enables “local” client access to remote NetBIOS servers.

NO Disables “local” client access to remote NetBIOS servers.

AcceptNETBIOS Broadcasts works with **Deliver NETBIOS Broadcasts** to configure, on a per-interface basis, how the IPX router responds to NetBIOS broadcast packets. For example, Figure 9-8 depicts a Wellfleet router which serves four IPX networks (**NET_1**, **NET_2**, **NET_3**, and **NET_4**). In actuality, these networks could be a single Novell network or an internet of Novell networks and routers.

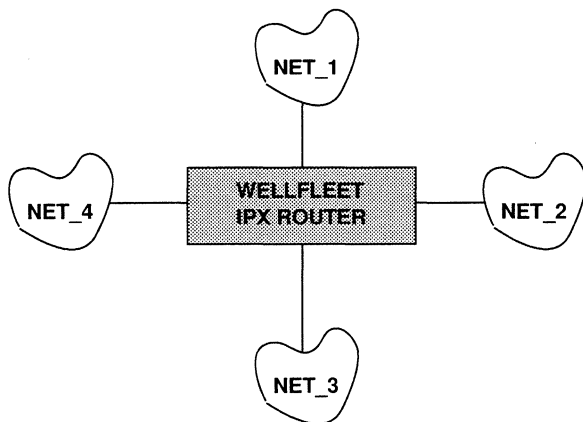


Figure 9-8. Sample IPX Internet

If you set:

- Accept NETBIOS Broadcasts** and **Deliver NETBIOS Broadcasts** to **YES** on all interfaces:

The IPX Router will broadcast a NetBIOS broadcast (originated by a client on **NET_1**) on all of its other interfaces (those to **NET_2**, **NET_3**, and **NET_4**).

- AcceptNETBIOS Broadcast to NO and Deliver NETBIOS Broadcasts to YES** for the **NET_1** interface, and **Accept NETBIOS Broadcasts and Deliver NETBIOS Broadcasts to YES** on all other interfaces

The IPX Router will *not* broadcast a NetBIOS broadcast (originated by a client on **NET_1**) on any of its other interfaces. This configuration prevents NetBIOS client applications on **NET_1** from initiating and establishing sessions with NetBIOS server applications on any network other than **NET_1**. However, client applications on **NET_2**, **NET_3**, and **NET_4** can still initiate and establish sessions with server applications on **NET_1**.

- AcceptNETBIOS Broadcast to YES and Deliver NETBIOS Broadcasts to NO** for the **NET_1** interface, and **Accept NETBIOS Broadcasts and Deliver NETBIOS Broadcasts to YES** on all other interfaces

The IPX Router will *not* broadcast a NetBIOS broadcast (originated by a client on **NET_2**, **NET_3**, or **NET_4**) to **NET_1**. This configuration prevents NetBIOS client applications on **NET_2**, **NET_3**, and **NET_4** from initiating and establishing sessions with NetBIOS server applications on **NET_1**. However, client applications on **NET_1** can still initiate and establish sessions with server applications on **NET_2**, **NET_3**, and **NET_4**.

7. At Deliver NETBIOS Broadcast, enable or disable remote access to “local” servers.

YES Enables remote access to “local” servers.

NO Disables remote access to “local” servers.

Note

AcceptNETBIOS Broadcasts works in conjunction with **Deliver NETBIOS Broadcasts** to configure, on a per-interface basis, how the IPX router responds to NetBIOS broadcast packets. Refer to the example provided for the **AcceptNETBIOS Broadcasts** parameter.

8. At Supply RIP Updates, enable or disable the RIP supply function.

YES Enables the RIP supply function so that the IPX router transmits periodic RIP updates across the circuit group.

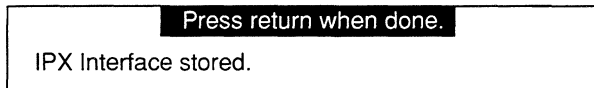
NO Disables the RIP supply functions,

9. At RIP Interface Cost, enter a cost (a number from 1 to 15) for each router hop.

Standard IPX RIP implementations assign a cost of 1 to each hop. Keep in mind, that if you increase the cost, you will reach the value of 16 (at which IPX declares a destination unreachable) more quickly.

10. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.



NCU returns to the **IPX REDIRECTOR** window. Repeat this procedure for each additional IPX interface you wish to modify.

9.4.2 Configuring Interface-Specific Filters and NetBIOS Static Routes

You can configure *Service Advertising Protocol (SAP) filters* and *NetBIOS static routes* on a per-interface basis. The following sections describe how to configure SAP filters and NetBios Static Routes.

9.4.2.1 Configuring SAP Filters

The node can filter SAP transmissions on an interface basis. By controlling the advertisement of servers, SAP filters enable the logical partitioning (for security and/or management purposes) of an IPX internet. If a server is filtered and not advertised on a given IPX network, workstations on that network cannot access that server. In effect, SAP filters erect user-configured logical partitions between network workstations and network servers.

You can configure SAP filters at both the network and individual-server level. Both network-level filters and server-level filters consist of a pattern (containing a network number and a server type) and an action (advertise or ignore):

- If the action is **ADVERTISE**, the interface advertises any server matching the pattern.
- If the action is **IGNORE**, the interface does not advertise any server matching the pattern.

You can specify up to 50 filters for each level. Within the network or server level, the filter's position in the network-level or server-level virtual queue determines filter precedence. Filters are moved into the queue in the order they are created:

- The first-in filter is the one with the lowest precedence
- The last-in filter is the one with the highest precedence.

Conflicts between server-level and network-level filters are resolved in favor of the server-level filter. With no configured filters, the IP router advertises all servers listed in its bindery.

You configure SAP filters for an interface from the **IPX INTERFACE** window for the interface. To display this window, first select the interface under **Circuit Group** in the **IPX REDIRECTOR** window; next, select **Interface** and then **Modify**. NCU displays the **IPX INTERFACE** window for that interface. You may now configure SAP network-level filters and SAP server-level filters.

9.4.2.1.1 Configuring SAP Network-Level Filters

You configure SAP network-level filters for an interface from the **IPX INTERFACE** window for an interface. Select **Filters** and then **SAP Network Level**. NCU displays the **IPX SAP NETWORK FILTERS** window (see Figure 9-9). You may now add, update, and delete SAP network-level filters.

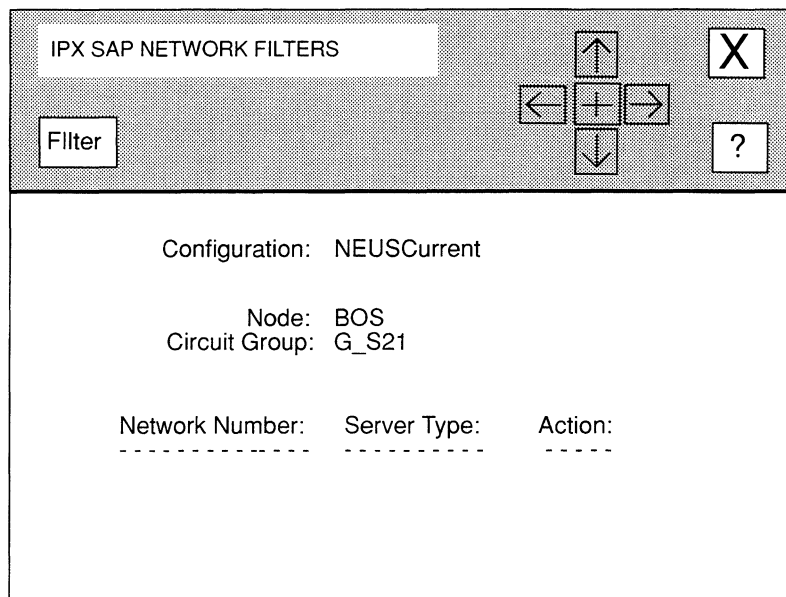


Figure 9-9. IPX SAP NETWORK FILTERS Window

9.4.2.1.1.1 Adding SAP Network-Level Filters

You add SAP network-level filters from the **IPX SAP NETWORK FILTERS** window, as follows:

1. Select and then .

NCU displays the **ADD SAP NETWORK FILTER** window (see Figure 9-10).

2. At **Network Number**, enter the server network-address portion (in 8-digit hexadecimal format — be certain, if you include leading zeros) of the filter pattern.

Enter **FFFFFFFF** to indicate “all networks”.

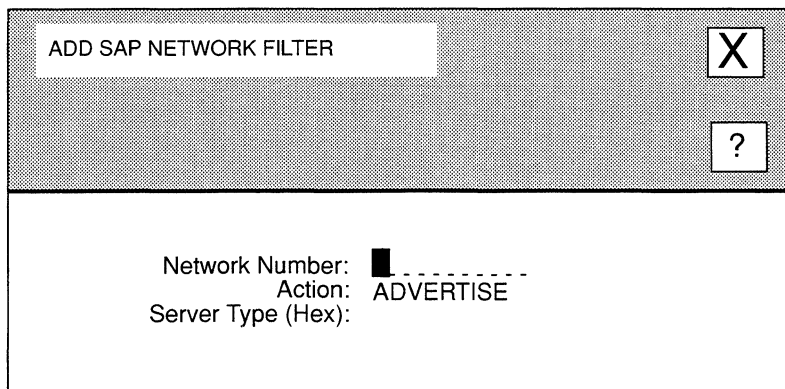


Figure 9-10. ADD SAP NETWORK FILTER Window

3. At **Action**, select how you want the IPX router to handle a SAP advertisement which contains a server that matches the filter pattern you specify.

ADVERTISE..... Specifies that the IPX router transmits SAP advertisements containing servers which match the pattern.

IGNORE Specifies that the IPX router drops such servers from the SAP advertisement.

4. At **Server Type (Hex)**, enter the server-type portion (in 4-digit hexadecimal format — be certain, if you include leading zeros) of the filter pattern.

Enter **FFFF** to indicate “all types”.

5. Select and then .

NCU returns to the **IPX SAP NETWORK FILTERS** window, which displays the SAP network-level filter you just added.

9.4.2.1.1.2 Updating SAP Network-Level Filters

You update SAP network-level filters from the **IPX SAP NETWORK FILTERS** window. First, select the filter under **Network Number**. Next, select and then . NCU displays the **ADD SAP NETWORK FILTER** window displaying the current settings for the SAP network-level filter. See *Section 9.4.2.1.1.1, Adding SAP Network-Level Filters* for information on how to set these parameters.

9.4.2.1.1.3 Deleting SAP Network-Level Filters

You delete SAP network-level filters from the **IPX SAP NETWORK FILTERS** window. First, select the filter under **Network Number**. Next, select and then . NCU deletes the SAP network-level filter.

9.4.2.1.2 Configuring SAP Server-Level Filters

You configure SAP server-level filters for an interface from the **IPX INTERFACE** window for an interface. Select and then . NCU displays the **SAP SERVER LEVEL FILTERS** window (see Figure 9-11). You may now add, update, and delete SAP server-level filters.

9.4.2.1.2.1 Adding SAP Server-Level Filters

You add SAP server-level filters from the **SAP SERVER LEVEL FILTERS** window, as follows:

1. Select and then .

NCU displays the **ADD SAP SERVER FILTER** window (see Figure 9-12).

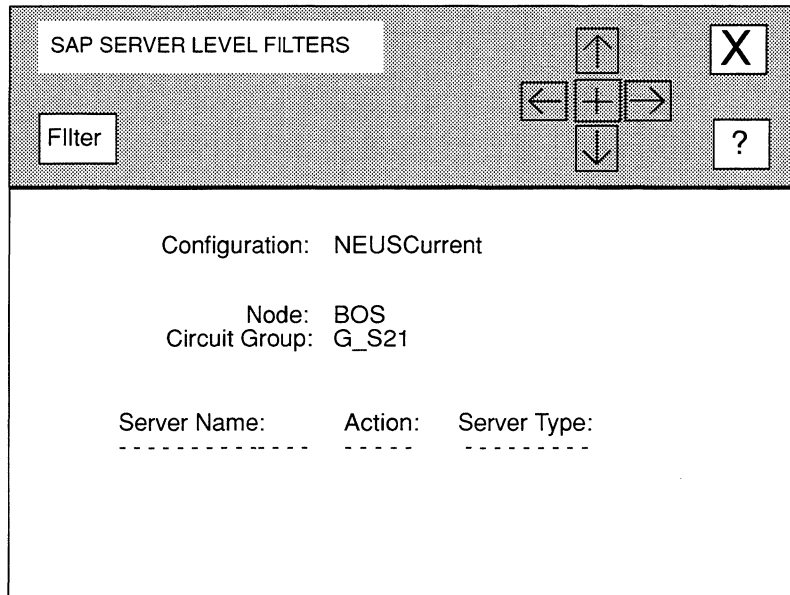


Figure 9-11. SAP SERVER LEVEL FILTERS Window

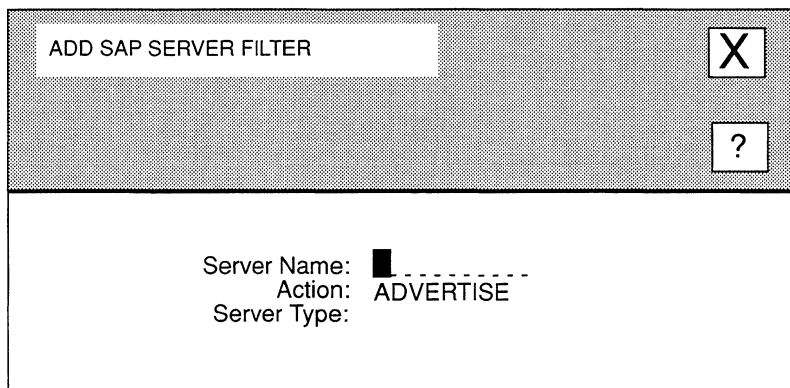


Figure 9-12. ADD SAP SERVER FILTER Window

2. **At Server Name, enter the server-name portion (any valid Novell server name of up to 48 characters) of the filter pattern.**

You may use any keyboard character except the tilde (~) character. If **Server Name** contains 48 characters, the node sets the final character to NULL (hexadecimal 00) when matching against actual server names. If **Server Name** contains less than 48 characters, the node sets the name left-justified and the remaining characters are NULL-filled. Name matching is performed up to the first NULL character.

Note

Matching is case sensitive. **Ken's.IPX.Router** does not equal **ken's.ipx.router**.

3. **At Action, select how you want the IPX router to handle a SAP advertisement which contains a server that matches the filter pattern you specify.**

ADVERTISE..... Specifies that the IPX router transmits SAP advertisements containing servers which match the pattern.

IGNORE Specifies that the IPX router drops such servers from the SAP advertisement.

4. **At Server Type, enter the server-type portion (in 4-digit hexadecimal format — be certain, if you include leading zeros) of the filter pattern.**

You cannot enter **FFFF** to indicate “all types”.

5. Select and then .

NCU returns to the **SAP SERVER LEVEL FILTERS** window, which displays the SAP network-level filter you just added.

9.4.2.1.2.2 Updating SAP Server-Level Filters

You update SAP network-level filters from the **SAP SERVER LEVEL FILTERS** window. First, select the filter under **Server Name**. Next, select and then . NCU displays the **ADD SAP SERVER FILTER** window displaying the current settings for the SAP server-level filter. See *Section 9.4.2.1.2.1, Adding SAP Server-Level Filters* for information on how to set these parameters.

9.4.2.1.2.3 Deleting SAP Server-Level Filters

You delete SAP server-level filters from the **SAP SERVER LEVEL FILTERS** window. First, select the filter under **Server Name**. Next, select and then . NCU deletes the SAP server-level filter.

9.4.2.2 Configuring NetBIOS Static Routes

Wellfleet's IPX router provides a non-Novell-standard "static routing" mechanism that converts "all nets" IPX NetBIOS packets to "directed" broadcast (a network-specific broadcast). Thus, you can logically partition an IPX NetBIOS network and minimize the bandwidth that the IPX "all nets" broadcast facility uses.

Each IPX router interface supports up to 50 NetBIOS static routes arranged as a table. Each NetBIOS static route specifies a NetBIOS resource name and a destination network (where the resource resides). With configured NetBIOS static routes, the IPX router compares all IPX NetBIOS broadcast packets received on an interface with interface-specific NetBIOS static routes:

- ❑ If the NetBIOS destination name in the packet matches a table entry:
The IPX router transmits the NetBIOS packet to the associated destination network
- ❑ If the NetBIOS destination name in the packet does not match a table entry:
The IPX router treats the packet as specified by the **AcceptNETBIOS Broadcast** and **Deliver NETBIOS Broadcasts** parameters

Note

NetBIOS static routes take precedence over the **AcceptNETBIOS Broadcast** and **Deliver NETBIOS Broadcasts** parameters. For example, a statically routed NetBIOS broadcast packet will be delivered to the destination network regardless of how **Deliver NETBIOS Broadcasts** is set at the receiving end.

Typically, you use NetBIOS static routing to enable a NetBIOS client on one network to establish a session with a remote NetBIOS server. To facilitate session establishment and use minimum bandwidth, the IPX router interface connected to the client network requires a NetBIOS static route that specifies the server's network and the server's name. Judicious configuration of IPX NetBIOS static routes, in conjunction with the **AcceptNETBIOS Broadcast** and **Deliver NETBIOS Broadcasts** parameters, enables session-establishment control; thus, facilitating internet security and management.

Note

IPX NetBIOS static routing is not a Novell "standard"; this feature may not interoperate with non-Wellfleet routers.

You configure NetBIOS static routes for an interface from the **IPX INTERFACE** window for an interface. Select and then . NCU displays the **NETBIOS BROADCAST STATIC ROUTES** window (see Figure 9-13). You may now add, update, and delete NetBIOS static routes.

9.4.2.2.1 Adding NetBIOS Static Routes

You add NetBIOS static routes from the **NETBIOS BROADCAST STATIC ROUTES** window, as follows:

1. Select and then .

NCU displays the **ADD NETBIOS STATIC ROUTES** window (see Figure 9-14).

2. At **Destination Network**, enter the network number (in 8-digit hexadecimal format — be certain, if you include leading zeros) of the network where the NetBIOS target resides.
3. At **Resource Name**, enter the name (up to 16 characters) of the NetBIOS target as it appears in *IPX NetBIOS Name Find Packets*.

You may use any keyboard character except the tilde (~) character and the backslash (\) character. If you cannot enter a character from the keyboard, you can enter it in 2-digit hexadecimal format for as “\xx” (where xx is a 2-digit hexadecimal value) — you enter the backslash character as “\.”.

NETBIOS BROADCAST STATIC ROUTES

Routes

Configuration: NEUSCurrent

Node: BOS

Circuit Group: G_S21

Destination Net: _____

Resource Name: _____

Figure 9-13. NETBIOS BROADCAST STATIC ROUTES Window

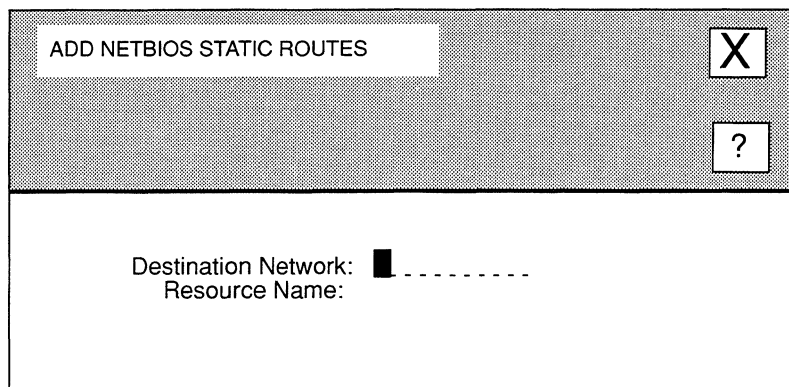


Figure 9-14. ADD NETBIOS STATIC ROUTES Window

For example, the name **JOE's Server** \\03\01 results in the following byte sequence:

```

4A 4F 45 27 73 20 53 65 72 76 65 72 5C 03 01
J O E ' s       S e r v e r \ 03 01
    
```

If **Resource Name** is less than 16-bytes, the byte string is left-justified and NULL filled. The name match is performed on all 16 bytes.

Note

Matching is case sensitive. **JOE's Server** does not equal **joe's server**.

4. Select and then .

NCU returns to the **NETBIOS BROADCAST STATIC ROUTES** window, which displays the NetBIOS static route you just added.

9.4.2.3 Updating NetBIOS Static Routes

You update NetBIOS static routes from the **NETBIOS BROADCAST STATIC ROUTES** window. First, select the filter under **Destination Net**. Next, select and then . NCU displays the **NETBIOS BROADCAST STATIC ROUTES** window displaying the current settings for the NetBIOS static route. See *Section 9.4.2.2.1, Adding NetBIOS Static Routes* for information on how to set these parameters.

9.4.2.4 Deleting NetBIOS Static Routes

You delete NetBIOS static routes from the **NETBIOS BROADCAST STATIC ROUTES** window. First, select the filter under **Destination Net**. Next, select and then . NCU deletes the NetBIOS static route.

9.4.3 Deleting IPX Interfaces

You delete IPX interfaces from the **IPX REDIRECTOR** window. First, select the interface under **Circuit Group**. Next, select and then . NCU deletes the IPX interface.

9.5 Configuring Static Routes

Static routes are user-specified transmissions paths. You configure static routes when you want to restrict the paths that packets can follow to paths you specifically define. Like routes learned through RIP, static routes are listed in the XNS routing table. Unlike routes learned through RIP, static routes cannot be overwritten.

In addition, the Wellfleet IPX router provides a non-Novell-standard “static routing” mechanism that enables a logical partitioning of an IPX NetBIOS network, and minimizes the bandwidth used by the IPX “all nets” broadcast facility. See *Section 9.4.2, Configuring Interface-Specific Filters and NetBIOS Static Routes* for information.

You configure static routes from the **IPX STATIC ROUTES** window (see Figure 9-15). To display this window, select in the **IPX REDIRECTOR** window. You may now add, update, and delete static routes.

9.5.1 Adding Static Routes

You add static routes from the **IPX STATIC ROUTES** window (see Figure 9-15), as follows:

1. Select and then .

NCU displays the **ADD IPX STATIC ROUTE** window (see Figure 9-16).

2. At **Target Net**, enter the network number of the destination network.
3. At **Next Hop Host**, enter the host address of the next-hop router used to reach **Target Net**.
4. At **Next Hop Net**, enter the network address of the next-hop router.

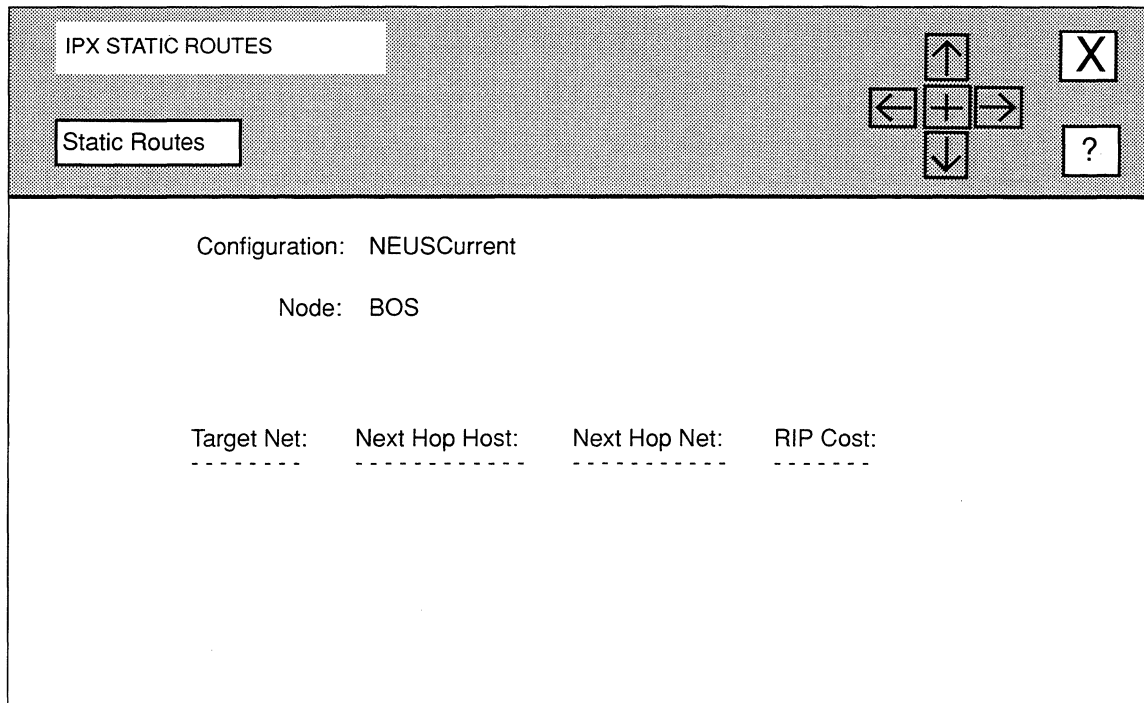


Figure 9-15. IPX STATIC ROUTES Window

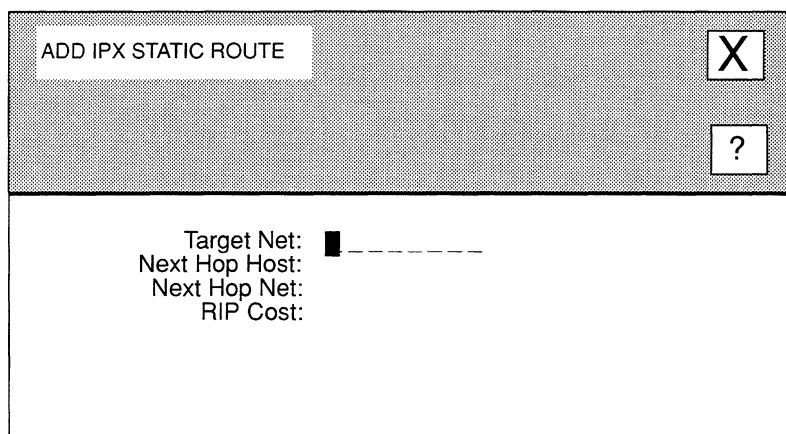


Figure 9-16. ADD IPX STATIC ROUTE Window

5. At **RIP Cost**, enter the cost (a number from 1 to 15) for the relay to **Target Net**.

If you set the interface-specific **RIP COST** to 1, simply enter the number of hops to **Target Net**. Otherwise, enter the value you obtain when you total the individual RIP costs of each hop to **Target Net** — the value you enter can be no greater than 15 (XNS declares a destination unreachable that has a cost of 16).

6. Select and then **Save** .

NCU displays this window; press **[RETURN]** to clear it from the console.

```
Press return when done.
IPX Static Route stored.
```

NCU returns to the **IPX STATIC ROUTES** window, which now displays the static route you just added. Repeat this procedure for each additional IPX static route you wish to add.

9.5.2 Updating Static Routes

You update static routes from the **IPX STATIC ROUTES** window. First, select the static route you wish to update under **Target Net**. Next, select **Static Routes** and **Update** . NCU displays the **ADD IPX STATIC ROUTE** window displaying the current parameter settings for that static route. From this point on, updating a static route is the same as adding a static route, see *Section 9.5.1, Adding Static Routes* for information on parameter settings.

9.5.3 Deleting Static Routes

You delete static routes from the **IPX STATIC ROUTES** window. First, select the static route you wish to delete under **Target Net**. Next, select **Static Routes** and **Delete** . NCU displays this window; press **[RETURN]** to clear it from the console.

```
Press return when done.
IPX Static Route deleted.
```

10 Editing AppleTalk Parameters

AppleTalk parameters consist of:

- Basic parameters
Basic parameters apply to the entire AppleTalk router software module.
- Interface parameters
Interface parameters apply to individual AppleTalk interfaces.

This chapter describes how to access and edit these parameters. The first section provides an overview of AppleTalk.

10.1 AppleTalk Overview

Apple Computer, Inc. developed AppleTalk (a protocol suite) to provide connectivity between members of the Macintosh family of personal computers. The Wellfleet AppleTalk Router implements the AppleTalk Phase 2 protocol which extends the original AppleTalk suite and offers enhanced routing and naming services.

Note

The Wellfleet AppleTalk Router does not support AppleTalk Phase 1. You cannot route Phase 1 traffic through the AppleTalk Router; however, you can relay such traffic through the Bridge.

AppleTalk uses routers to construct large, and, possibly, geographically disperse, network systems. A group of local AppleTalk networks connected with routers is an AppleTalk internet. AppleTalk allows you to logically associate end stations within a network or across network (and cable) boundaries; this logical association is a zone.

Each node within an AppleTalk internet has a unique address. An AppleTalk address consists of a 16-bit network number and an 8-bit node identifier. The AppleTalk router provides each node with a range of values. The node randomly generates a network number from this range: a Macintosh or an Apple router randomly generates a node identifier that fall within the range 1 to 253; the Wellfleet AppleTalk router allows you to explicitly assign a node identifier.

AppleTalk distinguishes between three types of routers:

❑ Local router

Connects geographically-proximate AppleTalk networks (see Figure 10-1). Each connection between the router and a proximate network is a local port.

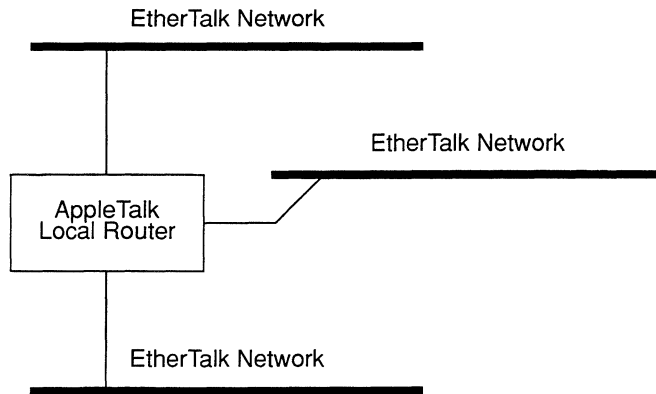


Figure 10-1. AppleTalk Local Router

❑ Half router

Connects remote AppleTalk networks by means of a wide-area connection terminated by another half router (see Figure 10-2).

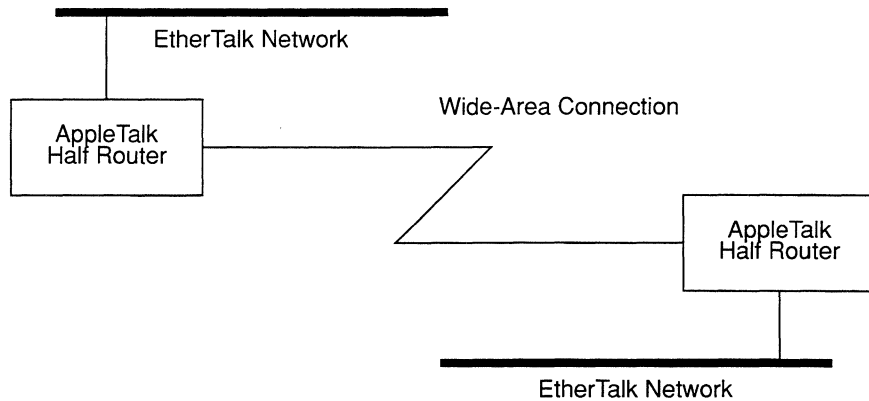


Figure 10-2. AppleTalk Half Router

- ❑ Backbone router
Connects AppleTalk networks through a non-AppleTalk backbone network (for example, a Token Ring, or a wide-area packet-switched network. See Figure 10-3.

Note

The Wellfleet AppleTalk router functions as a local router, a half router, or a backbone router.

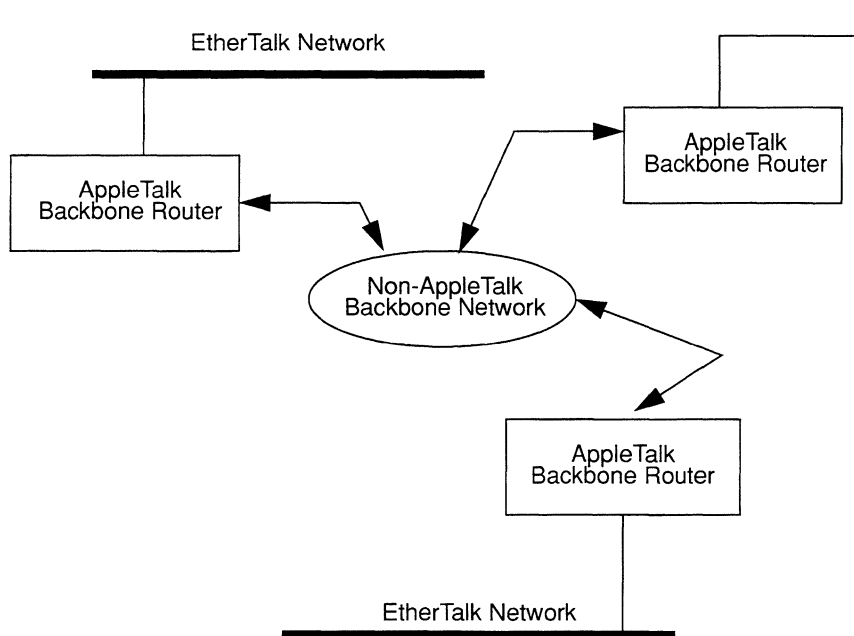


Figure 10-3. AppleTalk Backbone Router

AppleTalk follows the layered-model for network protocols. Higher-level protocols draw on the services of one-or-more lower-level protocols. Figure 10-4 depicts the hierarchical structure of a portion of the AppleTalk protocol suite — namely those protocols in the data-link, network, transport, and session layers that the AppleTalk Router implements. Table 10-1 lists the function of the implemented protocols.

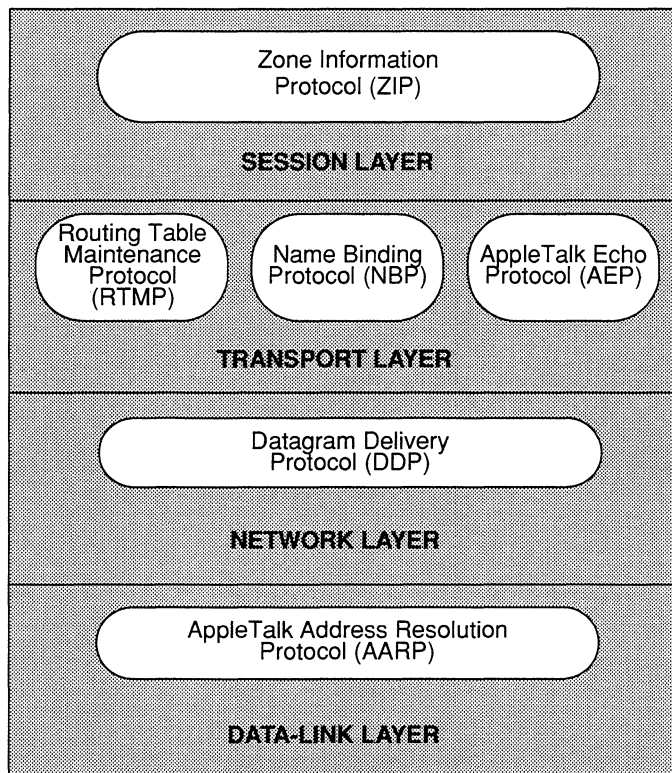


Figure 10-4. Layered Model of AppleTalk Routing Protocols

Table 10-1. Functions of AppleTalk Protocols

Protocol	Function
<i>AppleTalk Address Resolution Protocol (AARP)</i>	<p>Data-link layer protocol that translates AppleTalk node addresses (also called protocol addresses) to their equivalent data-link or hardware address.</p> <p>AARP maintains an Address Mapping Table (AMT) which lists protocol/hardware address equivalents and indicates the circuit group/port on which each address resolution is in effect. AARP transmits AARP Request packets and receives AARP Response packets in order to maintain the AMT. AARP also generates AARP Probe messages in order to ensure network-node/node-identifier pair integrity.</p> <p>An extension of the standard AARP functionality, AARP Probe prevents duplicate addresses in an internet. In the absence of an explicitly-assigned node identifier, AARP generates a tentative identifier, combines it with a randomly-generated network number, and then scans its AMT. If AARP located the tentative address in the AMT (which means another nodes uses that address) AARP generates another tentative identifier and scans the AMT again. Once AARP generates a tentative address that does not exist in the AMT, AARP transmits a series of Probe packets to the tentative address. If there is not positive response to the Probe packets, indicating that the address is unused, AARP validates the tentative address.</p>
<i>Datagram Delivery Protocol (DDP)</i>	<p>Network layer protocol that provides a “best-effort” socket-to-socket delivery mechanism over an AppleTalk internet.</p>
<i>Routing Table Maintenance Protocol (RTMP)</i>	<p>Transport layer protocol that creates and maintains the routing information the AppleTalk Router needs to transmit packets across an internet from a source to a destination socket.</p> <p>The AppleTalk Routing Table contains this routing information. Each table entry contains: a destination network range; the AppleTalk protocol address (network number and node identifier) through which the destination is reached; the number of router hops to the destination; and the route status.</p>

Table 10-1. (Continued) Functions of AppleTalk Protocols

Protocol	Function
<i>Name Binding Protocol (NBP)</i>	Transport layer protocol that translates a character string (a network node name) to the equivalent AppleTalk protocol address.
<i>AppleTalk Echo Protocol (AEP)</i>	Transport layer protocol that tests node reachability. AEP enables a node to send a packet to another internet node and to receive an identical (echoed) packet in response.
<i>Zone Information Protocol (ZIP)</i>	Session layer protocol that maintains an internet-wide mapping of zone names and network numbers.

10.2 Accessing AppleTalk Parameters

In order to access AppleTalk parameters, you must first display the **EDIT NODE CONFIGURATION** window for either the **DEFAULT_NODE** or a node on your network.

Note

Use the proper access mechanism to edit either the configuration-default parameters or the configuration parameters of a single node. See Chapter 1.

Figure 10-5 displays the **EDIT NODE CONFIGURATION** window for **DEFAULT_NODE**. In the figure, the network operator is changing the configuration-default parameters in NCU; any changes the network operator makes will affect every node configured thence on.

To access the **APPLETALK REDIRECTOR** window, select and then . NCU displays the **APPLETALK** window which allows you to edit AppleTalk parameters (see Figure 10-6).

10.3 Editing AppleTalk Basic Parameters

AppleTalk basic parameters apply to the entire AppleTalk router, rather than to individual AppleTalk interfaces on the node. When you connect a node to a network segment that runs AppleTalk, and activate the default settings, NCU automatically sets the AppleTalk basic parameters for the node. This section describes how to modify and delete AppleTalk basic parameters.

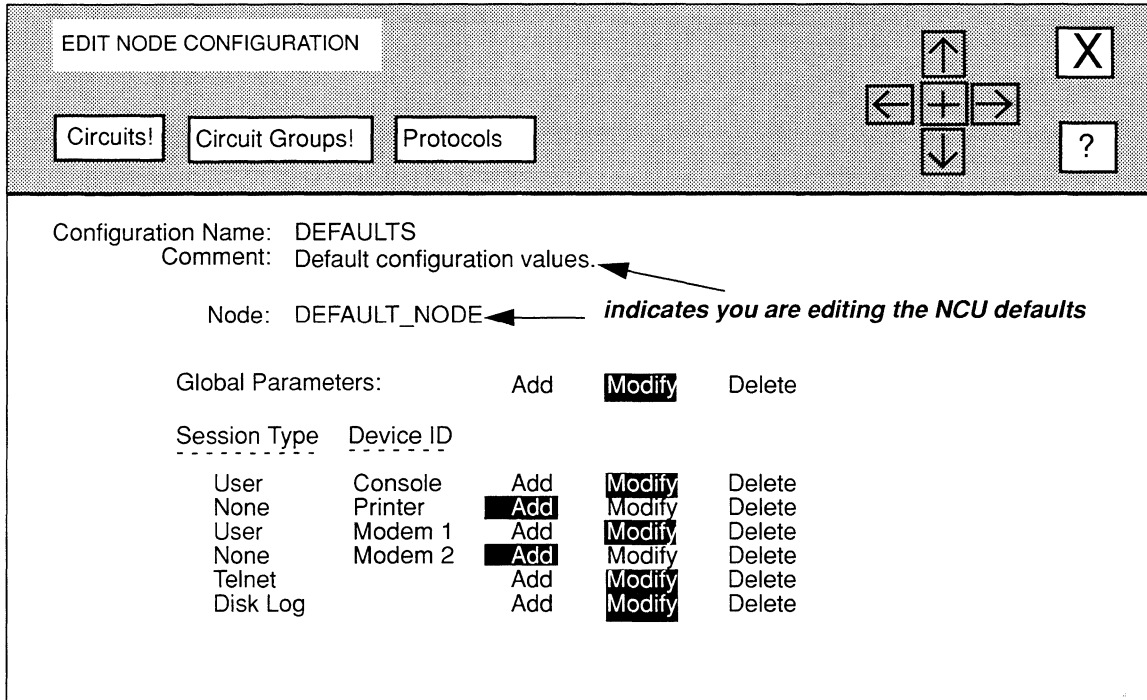


Figure 10-5. EDIT NODE CONFIGURATION Window for DEFAULT_NODE

10.3.1 Modifying AppleTalk Basic Parameters

You modify AppleTalk basic parameters from the **APPLETALK** window, as follows:

1. **At Auto Enable, specify the state of the AppleTalk router software when the node boots.**

This AppleTalk-router-specific **Auto Enable** works in conjunction with the global **Auto Enable** parameter to enable or disable the AppleTalk-router software module when the node boots, as follows:

- ❑ When global **Auto Enable** is set to **NO**, the AppleTalk router (and every other application software module) is unconditionally disabled.

You will subsequently need to enable the AppleTalk router manually with the NCL Interpreter after the node boots.

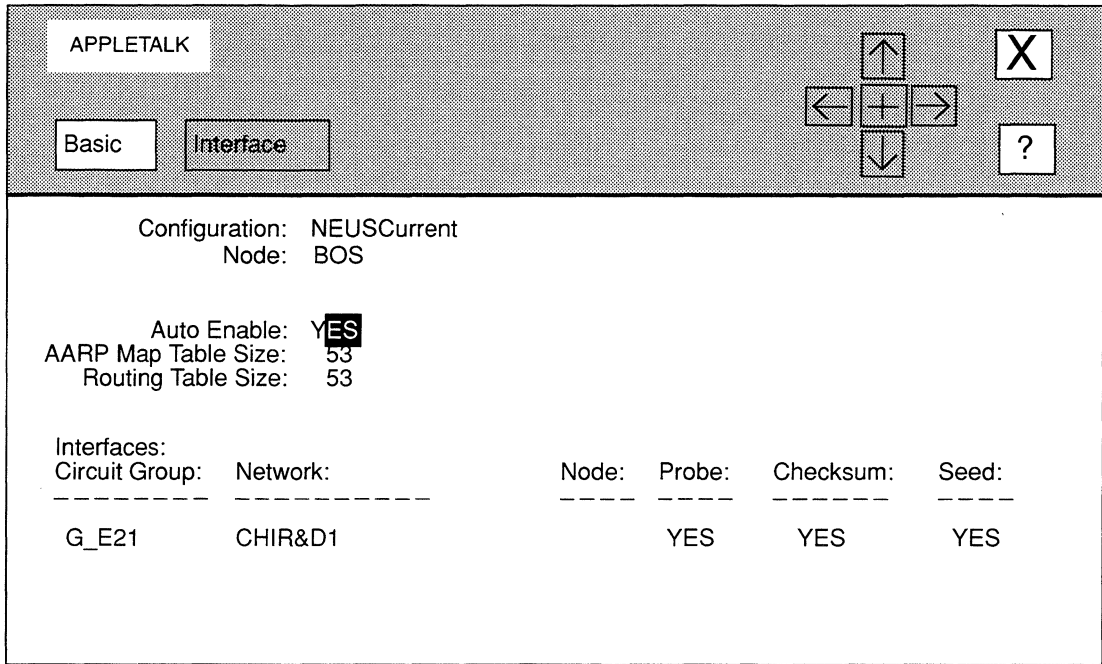


Figure 10-6. APPLE TALK Window

- ❑ When global **Auto Enable** is set to **YES**, the AppleTalk router (and every other application software module) is conditionally enabled.

If global **Auto Enable** is set to **YES**, do one of the following:

- Select **YES** to enable the AppleTalk router.
- Select **NO** to disable the AppleTalk router (you will subsequently need to enable the AppleTalk router manually with the NCL Interpreter after the node boots).

2. **At AARP Map Table Size, select the number of entries in the AppleTalk Router's Address Mapping Table (AMT).**

NCU provides responses that range from **53** to **9551**.

To calculate the number of AMT entries, estimate the number of end-nodes potentially reachable through the AppleTalk Router and select the next-highest available response that NCU provides.

3. At **Routing Table Size**, select the number of entries in the AppleTalk Router's routing table.

NCU provides responses that range from **53** to **9551**.

To calculate the number of routing-table entries, estimate the number of networks potentially reachable through the AppleTalk Router and select the next-highest available response that NCU provides.

4. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.

```
Press return when done.
AppleTalk Basic parameters stored.
```

10.3.2 Deleting AppleTalk Basic Parameters

You delete AppleTalk basic parameters from the **APPLETALK** window. Simply select and then . NCU displays this window, press **[RETURN]** to clear it from the console:

```
Press return when done.
Appletalk Redirector deleted.
```

10.4 Configuring AppleTalk Interfaces

You configure each AppleTalk interface individually. The following sections describe how to modify and delete AppleTalk interfaces.

10.4.1 Modifying AppleTalk Interfaces

You modify AppleTalk interfaces, as follows:

1. Select the interface you wish to modify under **Circuit Group** in the **APPLETALK** window.
2. Select and then .

NCU displays the **APPLETALK INTERFACE** window for that interface, which allows you to configure interface-specific parameters. Figure 10-7 depicts the **APPLETALK INTERFACE** window for a seed router; Figure 10-8 depicts the **APPLETALK INTERFACE** window for a non-seed router.

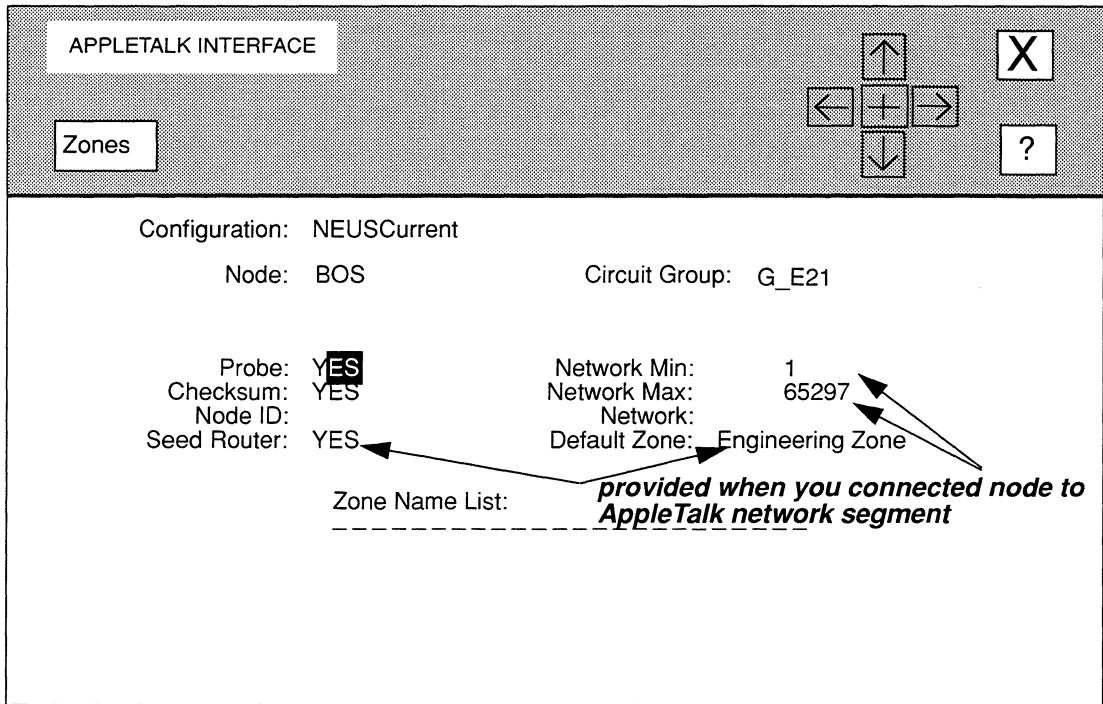


Figure 10-7. APPLE TALK INTERFACE Window with Seed Router Set to YES

3. At Probe, enable or disable AARP Probe packets generation and transmission across the circuit group.

Note

Probe works with Node ID (or, if you set Seed Router to YES, Probe works with Node ID and Network) to enable or disable the generation of AARP Probe packets and their subsequent transmission across the interface.

NCU provides two responses:

- YES Enables the node to generate and transmit AARP Probe packets. You should set Probe to YES, even if you assign an explicit node identifier at Node ID. Enabling Probe prevents duplicate AppleTalk addresses within an internet.

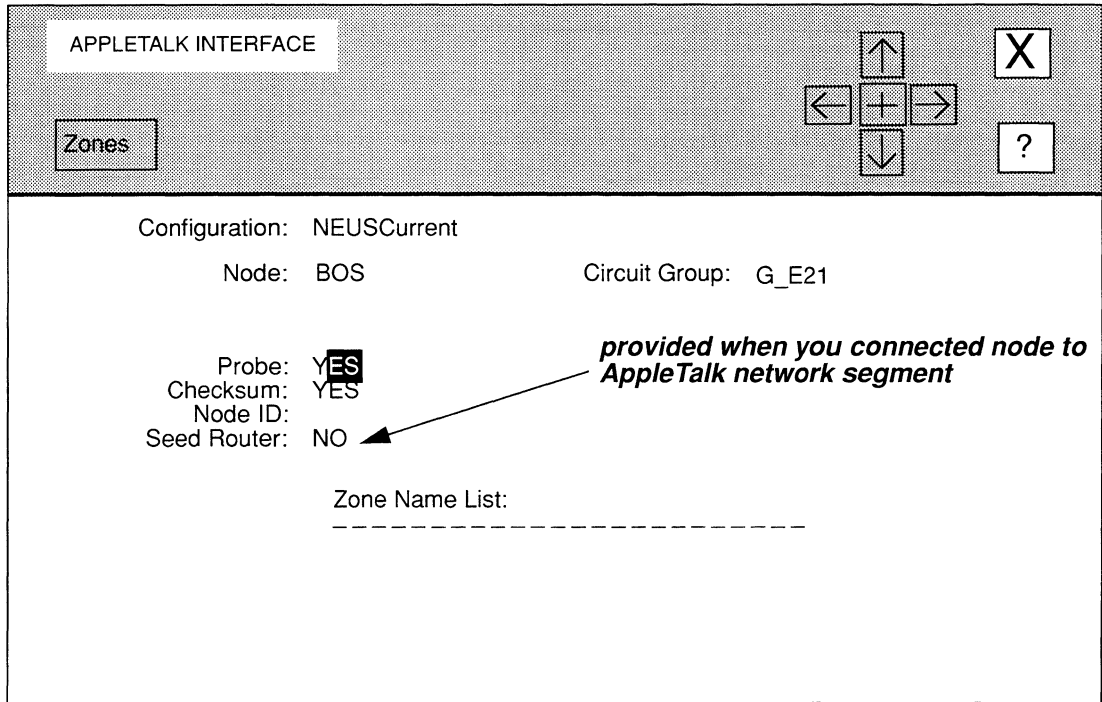


Figure 10-8. APPLE TALK INTERFACE Window with Seed Router Set to NO

NO Disables AARP *Probe* packet generation and transmission. You may set **Probe** to **NO**, as follows:

- If the interface supports a non-seed router (**Seed Router** set to **NO** as in Figure 10-8), you can set **Probe** to **NO**, but *only* if you explicitly assign a node identifier at **Node ID**. If you do not assign a node identifier, the *Probe* protocol is forced regardless of how you set **Probe** (see Table 10-2).
- If the interface supports a seed router (**Seed Router** set to **YES** as in Figure 10-7), you can set **Probe** to **NO**, but only if you *only* explicitly assign a node identifier at **Node ID** and a network number at **Network**. If you do not assign a node identifier and a network number at **Network**, the *Probe* protocol is forced regardless of how you set **Probe** (see Table 10-3).

Table 10-2. Probe Implementation for Non-Seed Routers

Node ID	Probe	Probe Implemented
assigned	disabled	No
unassigned	disabled	Yes
-----	enabled	Yes

Table 10-3. Probe Implementation for Seed Routers

Node ID	Network	Probe	Probe Implemented
assigned	assigned	disabled	No
assigned	unassigned	disabled	Yes
unassigned	assigned	disabled	Yes
unassigned	unassigned	disabled	Yes
-----	-----	enabled	Yes

4. At Checksum, enable or disable DDP checksumming for packets that the AppleTalk router constructs and transmits.

YES Enables DDP checksumming so that the AppleTalk router calculates and writes a 16-bit checksum in the header of any DDP packet that the router originates.

NO Disables DDP checksumming so that the AppleTalk router does not calculate a 16-bit checksum and write a value of 0 in the DDP packet header.

Note

Checksum has no effect on incoming packets. If the AppleTalk router receives a packet containing a checksum, it verifies the checksum.

5. At Node ID, either enter the circuit group/port-specific node identifier portion of the AppleTalk address (a number from 1 to 253) or leave the field empty to enable the AppleTalk router to assign its own node identifier.

The AppleTalk router uses multiple AppleTalk addresses (one address for each network to which the router connects directly). Each node within the AppleTalk internet must have a unique AppleTalk address (network number/node identifier pair). Whether or not you explicitly assign a node identifier, you should set **Probe** to **YES** to ensure a unique node identifier.

6. At **Seed Router**, select whether the router is a seed or non-seed router for the network to which it is attached.

Note

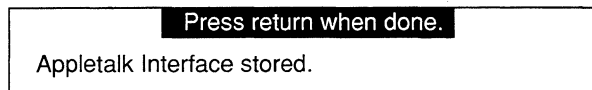
You set **Seed Router** when you connected the node to the AppleTalk network segment.

YES Specifies the router is a seed router. If you set **Seed Router** to **YES**, NCU displays additional parameters. Go directly to *Section 10.4.1.1, Configuring Seed Routers* for instructions on how to set these parameters.

NO Specifies the router is not a seed router. Go directly to step 7.

7. Select and then Save .

NCU displays this window; press [RETURN] to clear it from the console.



NCU returns to the **APPLETALK** window. Repeat this procedure for each additional AppleTalk interface you wish to modify.

10.4.1.1 Configuring Seed Routers

Once you set **Seed Router** to **YES** in the **APPLETALK INTERFACE** window, you can configure the seed router, as follows:

1. At **Network Min**, enter the lowest network number (from 1 to 65297) for the directly-connected network segment.

Note

You set **Network Min**. when you connected the node to the AppleTalk network segment.

Network Min works with **Network Max** to specify the range of network numbers available to nodes on the directly-connected network segment. In order to increase the number of nodes that can reside on a LAN medium, AppleTalk Phase 2 requires the seed router to provide a range of network numbers and to make the range available to network nodes. Network nodes can then randomly generate a network number (from within the provided range) the same as they randomly

generate a node identifier. **Network Min** specifies the lowest number in the range, **Network Max** specifies the highest number.

2. **At Network Max, enter the highest network number (from 1 to 65297) for the directly-connected network segment.**

Note

You set **Network Max** when you connected the node to the AppleTalk network segment.

3. **At Network, either enter the circuit group/port-specific network number portion of the AppleTalk address (a value equal to or greater than Network Min and equal to or less than Network Max) or leave the field empty to enable the AppleTalk router to assign a random network number (also, a value equal to or greater than Network Min and equal to or less than Network Max).**

Network works with **Node ID** and **Probe** to enable or disable the generation of AARP Probe packets and their subsequent transmission across the interface (see Table 10-3).

4. **At Default Zone, enter the default zone name (a maximum of 32 characters — you may use characters from the AppleTalk character set* and any keyboard-generated character, except the tilde “~”).**

Note

You set **Default Zone** when you connected the node to the AppleTalk network segment.

The 32-character limit applies to the zone name as it is entered. *When you compute the zone name length, count each AppleTalk character as 3 characters and each escaped backspace character as 2 characters.*

A zone is a logical grouping of network devices. A zone can be confined to a single network or can span multiple networks within the AppleTalk internet. You identify a zone by its zone name.

You represent the AppleTalk character set, as follows:

`\xx`

where:

`\` is there backslash character

`xx` is the 2-digit hexadecimal value that identifies the special AppleTalk character. For example, to use the zone name $\Delta\Sigma\Pi$ enter the following string:

$\backslash c6\backslash b7\backslash b9$

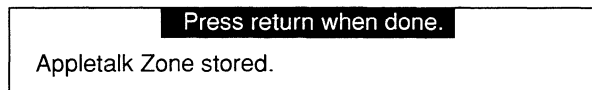
where **c6**, **b7**, and **b9** are the hexadecimal values of the AppleTalk characters Δ , Σ , and Π , respectively.

You can use the backslash character (\backslash) in the zone name if you precede with it another backslash. For example, to use the zone name **Bldg1\1st-floor**, enter the following string:

Bldg1\\1st-floor

5. To add additional zone names (up to a maximum of 10), select **Zones** and then **Add** .

NCU displays the **APPLETALK ZONE** window (see Figure 10-9). Simply enter the zone name and then select **X** and **Save** . NCU displays this window; press **[RETURN]** to clear it from the console.



NCU returns to the **APPLETALK INTERFACE** window, which now displays the zone you just added. Repeat this procedure until you have added all desired zones (up to a maximum of 10).

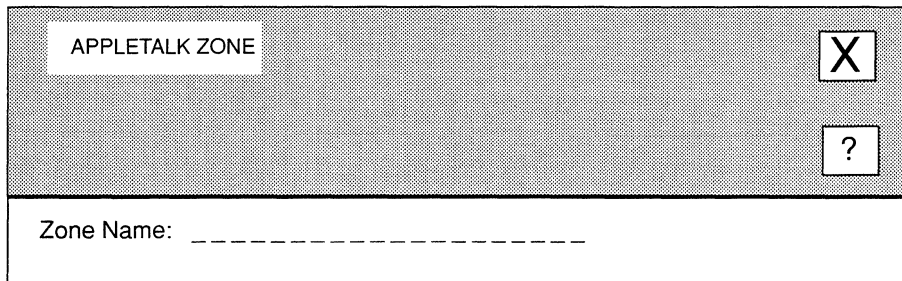
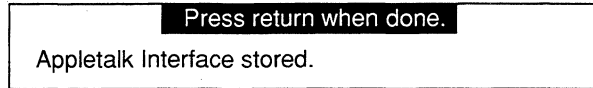


Figure 10-9. APPLETALK ZONE Window

To delete a zone, simply select the zone name under **Zone Name List** in the **APPLETALK INTERFACE** window, and then select and .

6. Select and then .

NCU displays this window; press **[RETURN]** to clear it from the console.



NCU returns to the **APPLETALK** window. Repeat this procedure for each additional AppleTalk interface you wish to modify.



15 Crosby Drive, Bedford, MA 01730-2204 Tel: (617) 275-2400 Fax: (617) 275-8421