

HUBwatch for Windows

User Information

Order Number: EK-487AA-UI. A01

Revision/Update Information: This is a new manual.

First edition, April 1993

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

© Digital Equipment Corporation 1993.

DEC, DECbridge, DECconnect, DECdirect, DEChub, DEcmcc, DECnet, DECserver, Digital, EtherWORKS, HUBwatch, LAT, MicroVAX, PATHWORKS, ThinWire, UNIBUS, VAX, VMS, and the DIGITAL logo are trademarks of Digital Equipment Corporation.

IBM is a registered trademark of International Business Machines Corporation.
Microsoft and MS-DOS are registered trademarks and Windows is a trademark of Microsoft Corporation.

NetWare and Novell are registered trademarks of Novell, Inc.

3Com is a registered trademark of 3Com Corporation.

VINES is a registered trademark of Banyan Systems, Inc.

Adobe and PostScript are registered trademarks of Adobe Systems, Inc.

All other trademarks and registered trademarks are the property of their respective holders.

This document was prepared using VAX DOCUMENT, Version 2.1.

Contents

Preface	xiii
1 Overview	
Introduction	1-1
HUBwatch for Windows Features	1-1
Product Description	1-1
2 Installing HUBwatch for Windows	
Introduction	2-1
Hardware Requirements	2-1
Software Requirements	2-2
HUBwatch Kit Contents	2-2
Installing HUBwatch	2-3
Configuring Your Network Management Station for HUBwatch	2-3
Modifying AUTOEXEC.BAT	2-4
Modifying CONFIG.SYS	2-5
Configuring NETDEV.SYS	2-5
3 HUBwatch Tools for Mapping Networks	
Introduction	3-1
Icons	3-1
Alarm Icons	3-2
Device Icons	3-2
Site Icon	3-3
Bridge Icon	3-3
Router Icon	3-3
Host Icon	3-4
Hub Icon	3-4
Repeater Icon	3-4

Terminal Server Icon	3-4
Generic Device Icon	3-5
Specialized Network Icons	3-5
Pointers	3-5
Network Connections	3-6
Background Maps	3-7
Notes	3-8

4 Using HUBwatch for Windows

Introduction	4-1
Starting a HUBwatch Session	4-1
HUBwatch Views	4-1
HUBwatch Menus	4-2
Buttons	4-3
Navigating within HUBwatch	4-4
The <u>F</u> ind Option	4-5
Find <u>M</u> e Option	4-7
Zoom <u>I</u> n Option	4-8
Zoom <u>O</u> ut Option	4-8
Zoom <u>T</u> op Option	4-8
Printer Setup	4-8

5 Configuring Networks

Introduction	5-1
Creating a New Network	5-1
Adding Objects at Network or Site View	5-3
Adding Sites	5-3
Adding Devices	5-4
Adding Connections	5-13
Adding Specialized Network Icons	5-14
Using Auto Discovery	5-15
Adding Objects at the Hub View	5-18
Deleting Objects at Network or Site View	5-18
Deleting Devices	5-19
Deleting Connections	5-19
Deleting Other Icons (Specialized Network Icons)	5-20
Deleting Objects at Hub View	5-20
Moving Objects	5-21
Moving Objects within the Same Network View	5-21
Moving Objects to Another Network View	5-21

Modifying Views	5-22
Modifying Sites	5-22
Modifying Devices or Hubs	5-22
Opening an Established Network	5-26
Backing Up a Network	5-27
Restoring a Network	5-29
Deleting a Network	5-30
Modifying a Network	5-30

6 Managing Networks

Introduction	6-1
Managing Network Performance	6-1
Statistics Option	6-1
Graph Option	6-3
Ping Option	6-5
Managing Network Faults	6-7
Viewing Error Statistics	6-8
Viewing Alarms	6-9
Setting Audible Alarms	6-10
Setting Thresholds	6-11
Viewing the Alarm Log	6-14
Alarm Log Report Menu	6-16
Managing Network Security	6-16
Setting Passwords	6-17
Viewing Security Reports	6-18
Security Log Report Menu	6-21
Accessing the Management Information Base	6-21
Common Features of MIB Devices	6-22
Managing MIB Information	6-23
Accessing the <u>MIBII</u> Groups	6-23
System Group	6-24
ITF Group	6-25
AT Group	6-26
IP Group	6-27
ICMP Group	6-32
TCP Group	6-33
UDP Group	6-34
EGP Group	6-35
SNMP Group	6-36

Accessing Bridge Groups	6-37
Dot1dBase Group	6-38
Dot1dStp Group	6-39
Dot1dSr Group	6-40
Dot1dTp Group	6-41
Dot1dStatic Group	6-42
Accessing Char-like Groups	6-43
General Group (char-like)	6-43
Session Group	6-44
Accessing RS232-like Groups	6-45
General Groups (RS232-like)	6-45
Async Group	6-46
Sync Group	6-47
Viewing the MIB Variable	6-48
Viewing a Specific MIB Variable	6-49
Walking the MIB	6-50
Accessing Network Reports	6-50
Local Configuration Report	6-51
Brief Configuration Report	6-52
Full Configuration Report	6-53
Change Log	6-54
Change Log Report Menu	6-56

7 Managing Generic Devices

Introduction	7-1
Accessing Generic Devices	7-1

8 Managing DECagent 90 Modules

Introduction	8-1
Selecting the DECagent 90 Module	8-1
Accessing DECagent Information	8-3
Managing the DECagent 90 Configuration	8-3
DECagent Configuration Reports	8-4
Managing the DECagent 90 Performance	8-8
DECagent Performance Reports	8-8
Managing the DECagent 90 Faults	8-9
Viewing DECagent Error Report	8-11
Setting Thresholds - DECagent	8-12
Accessing the MIB	8-13
Printing DECagent Reports	8-13

A Menus

Introduction	A-1
Network View Menu	A-2
Site View Menu	A-3
Hub View Menu	A-4
Module View Menu	A-5
Device View Menu	A-6

B NOS with HUBwatch

Introduction	B-1
Using a Networked Printer	B-1
Switching Between HUBwatch and Your NOS	B-2
Two Versions on Your Hard Disk	B-2
Booting Your PC from a Diskette	B-3

C Clarkson Packet Drivers

Introduction	C-1
Overview	C-1
Installation with a Digital EtherWORKS Adapter and NETDEV.SYS	C-2
Customization of the Installed Clarkson Packet Driver	C-3

D Background Maps

Introduction	D-1
------------------------	-----

E MIB Descriptions

Introduction	E-1
MIBII Groups	E-1
System Group	E-1
Interface (ITF) Group	E-2
Address Translation (AT) Group	E-5
Internet Protocol (IP) Group	E-5
IP Address Table	E-7
IP Routing Table	E-8
IP Address Translation Table (Net-to-Media Table)	E-10
Internet Control Message Protocol (ICMP) Group	E-10
Transmission Control Protocol (TCP) Group	E-12
TCP Connection Table	E-14

User Datagram Protocol (UDP) Group	E-15
UDP Listener Table	E-16
Exterior Gateway Protocol (EGP) Group	E-16
EGP Neighbor Table	E-16
Simple Network Management Protocol (SNMP) Group	E-18
Bridge MIB	E-22
dot1dBase Group	E-22
Generic Bridge Port Table	E-22
dot1dStp Group	E-23
Spanning Tree Port Table	E-25
dot1dSr Group	E-27
dot1dTp Group	E-29
Forwarding Database for Transparent Bridges	E-30
Port Table for Transparent Bridges	E-31
Static (Destination-Address Filtering) Database:	E-32
Char-like MIB	E-34
Generic Character Group	E-34
Character Port Table	E-34
Character Session Table	E-38
RS232-like MIB	E-40
RS-232-like Asynchronous Port Group	E-41
RS-232-like Synchronous Port Group	E-42
Input Signal Table	E-43
Output Signal Table	E-44
DH_90 MIB	E-45
dh90 Group	E-45
da90 Group	E-49
ds90L Group	E-51
ds90LPortTable	E-55
ds90LSessionTable	E-56
drpt90 Group	E-57
drpt90PortTable	E-57
dbSysChar - System Characteristics Group	E-59
dbSysStatus - System Status Group	E-60
dbIfTable - Extended Interface Module Group	E-60
db90Char - Bridge Characteristics Group	E-61
db90Stat - Bridge Status Group	E-62
db90Coun - Bridge Counters Group	E-62

db90Span - Bridge Spanning Tree Group	E-63
db90IfTable - Extended Bridge Port Table	E-64
db90IfEtherTable - Extended Bridge Ethernet Port Table	E-65
db90IfSpanTable - Extended Bridge Spanning Tree Port Table	E-65
DECbridge 90 Protocol Database	E-66

F Documentation and Ordering

Introduction	F-1
Related Documentation	F-1
Ordering Information	F-2

Glossary

Introduction	Glossary-1
------------------------	------------

Index

Figures

3-1	Example of Network Connections	3-7
3-2	Example of Background Map	3-8
4-1	Find Window	4-5
4-2	Selections Dialog Box	4-6
4-3	Find Me Window	4-7
5-1	New Network Dialog Box	5-2
5-2	Add Site Dialog Box	5-4
5-3	Add Device Window	5-6
5-4	Add DEChub 90 Window	5-7
5-5	Add DECagent 90 Window	5-8
5-6	Community Window	5-11
5-7	Device System Group Window	5-12
5-8	Connection Information Window	5-14
5-9	Other Icons	5-15
5-10	Add Window	5-16
5-11	Modify Site Dialog Box	5-23
5-12	Modify Device Window	5-24
5-13	Control Panel Window	5-26

5-14	Open Network Window	5-27
5-15	Network Backup	5-28
5-16	Restore Window	5-29
5-17	Delete Network Window	5-31
5-18	Modify Network Window	5-32
6-1	DECserver Performance Window	6-2
6-2	MIB Object Selection Window	6-3
6-3	MIB Object Graph	6-4
6-4	Ping Window	6-6
6-5	DECagent Error Window	6-8
6-6	Current Alarms Network Wide Window	6-9
6-7	The Audible Alarm Dialog Box	6-11
6-8	Set Thresholds Window	6-13
6-9	Alarm Log Window	6-14
6-10	Alarm Log Report	6-15
6-11	Change Network Password Window	6-17
6-12	Security Log Window	6-19
6-13	Security Log Report	6-20
6-14	Device System Group Table	6-24
6-15	Interfaces Group Table	6-25
6-16	Address Translation Table	6-26
6-17	IP Group Table	6-28
6-18	IP Address Table	6-29
6-19	IP Routing Table	6-30
6-20	IP Net-To-Media Table	6-31
6-21	ICMP Group Table	6-32
6-22	TCP Group Table	6-33
6-23	UDP Group Table	6-34
6-24	EGP Group Table	6-35
6-25	SNMP Group Table	6-36
6-26	Bridge - dot1dBase Group Table	6-38
6-27	Bridge - dot1dStp Group Table	6-39
6-28	Bridge - dot1dSr Group Table	6-40
6-29	Bridge - dot1dTp Group Table	6-41
6-30	Bridge - Static (Destination - Address Filtering) Group Table	6-42
6-31	Character Port Table	6-43

6-32	Character Session Table	6-44
6-33	RS232 General Port Table	6-45
6-34	RS232 Asynchronous Port Group Table	6-46
6-35	RS232 Synchronous Port Group Table	6-47
6-36	MIB Extensions Window	6-48
6-37	Local Configuration Report	6-51
6-38	Brief Configuration Report	6-52
6-39	Full Configuration Report	6-53
6-40	Change Log Dialog Box	6-54
6-41	Change Log Report	6-55
8-1	Selecting a DECagent Module	8-2
8-2	Agent Module Report	8-5
8-3	Agent Edit Strings Report	8-7
8-4	DECagent Performance Report	8-9
8-5	DECagent Error Report	8-11
8-6	Set Thresholds Window	8-13
8-7	Output Dialog Box	8-14
A-1	Network View Menus	A-2
A-2	Site View Menus	A-3
A-3	HUB View Menus	A-4
A-4	Module View Menus	A-5
A-5	Device View Menus	A-6

Preface

Introduction

This manual describes how to use HUBwatch for Windows to manage the DEChub ONE family of products. It describes how to install HUBwatch, it provides an overview of the tools used for mapping your network, and it describes how to manage the network and the modules in the hub.

Organization

This manual contains eight chapters, six appendixes, a glossary, and an index.

- Chapter 1, Overview, provides an overview of the HUBwatch application, including the features and a product description.
- Chapter 2, Installing HUBwatch for Windows, lists the hardware and software required to run HUBwatch, describes the HUBwatch kit contents, and provides procedures for installing HUBwatch.
- Chapter 3, HUBwatch Tools for Mapping Networks, provides a description of all the graphic tools HUBwatch uses to assist you with mapping and managing your network within the graphic user interface (GUI). These tools include icons, pointers, and backgrounds.
- Chapter 4, Using HUBwatch for Windows, provides basic information for starting HUBwatch, understanding the layout of the menus, and navigating within the HUBwatch application.
- Chapter 5, Configuring Networks, provides information on how to use HUBwatch to build a configuration from which you can manage your network.
- Chapter 6, Managing Networks, explains how to use the HUBwatch application to manage network performance, alarms, security, management information bases (MIBs), and reports.
- Chapter 7, Managing Generic Devices, explains how to manage generic devices on your network.

- Chapter 8, Managing DECagent 90 Modules, describes how to use HUBwatch to manage a DECagent 90.
- Appendix A, Menus, provides a HUBwatch map for the menu items at different levels of the application.
- Appendix B, NOS with HUBwatch, provides information about using HUBwatch with NetWare or other network operating systems (NOSs), and creating a boot diskette.
- Appendix C, Clarkson Packet Drivers, explains how to use HUBwatch on a PC that is running a network operating system.
- Appendix D, Background Maps, provides a list of a sample products that may be used to generate background geographic maps to use with HUBwatch.
- Appendix E, MIB Descriptions, provides a description for each object within MIBII, the Bridge MIB, the Char-like MIB, the RS232-like MIB and the DH90 MIB.
- Appendix F, Documentation and Ordering, contains information on how to order related documentation.
- Glossary, defines terms used in this manual.

Conventions

The following table lists the conventions used in this manual.

Convention	Meaning
Note	Contains important information.
Return	A key name enclosed in a box indicates that you press that key. In this example, you would press the Return key.
<i>Italic type</i>	Emphasizes important information, indicates variables, and indicates complete titles of documents.
Boldface type	Boldface type in examples indicates user input. Boldface type in text indicates the first instance of terms defined either in the text, in the glossary, or both.
Monospaced type	Text that the system displays on the screen.
Bold monospaced type	Text you enter is shown in bold monospaced type.
Click on	To press and release a mouse button when the pointer is positioned on an active object.

Convention	Meaning								
Drag	To press and hold a mouse button, move the mouse, and then release the button.								
MB	Indicates a mouse button.								
	<table border="1"> <thead> <tr> <th>Mouse Button</th> <th>Position</th> </tr> </thead> <tbody> <tr> <td>MB1</td> <td>Left mouse button</td> </tr> <tr> <td>MB2</td> <td>Right mouse button (middle button on 3-button mouse)</td> </tr> </tbody> </table>	Mouse Button	Position	MB1	Left mouse button	MB2	Right mouse button (middle button on 3-button mouse)		
Mouse Button	Position								
MB1	Left mouse button								
MB2	Right mouse button (middle button on 3-button mouse)								
<u>Underline</u>	Indicates the underlined letter on the screen menu item, option, or button. These are designed for use if you do not have a mouse or do not want to use your mouse for accessing menu items. To access items without using a mouse, do the following:								
	<table border="1"> <thead> <tr> <th>To ...</th> <th>Press ...</th> </tr> </thead> <tbody> <tr> <td>Access menu items</td> <td>[Alt] and the underlined letter</td> </tr> <tr> <td>Access options</td> <td>[Shift] and the underlined letter</td> </tr> <tr> <td>Activate buttons</td> <td>[Alt] and the underlined letter</td> </tr> </tbody> </table>	To ...	Press ...	Access menu items	[Alt] and the underlined letter	Access options	[Shift] and the underlined letter	Activate buttons	[Alt] and the underlined letter
To ...	Press ...								
Access menu items	[Alt] and the underlined letter								
Access options	[Shift] and the underlined letter								
Activate buttons	[Alt] and the underlined letter								

1

Overview

Introduction

This chapter provides a basic overview of HUBwatch for Windows. The following topics are included in this chapter:

- HUBwatch for Windows Features
- Product Description

HUBwatch for Windows Features

The following list summarizes the features of HUBwatch for Windows:

- Runs on a PC with DOS and Microsoft Windows Version 3.1
- Uses the Simple Network Management Protocol (SNMP)
- Provides a Microsoft Windows Graphic User Interface (GUI)
- Provides a graphic network map
- Supports the Digital enterprise-specific MIB definitions (DEChub 90 MIB as of September, 9, 1992) for the DEChub 90 family of products
- Provides near real-time information on hub configuration, and module and port status
- Provides color-coded graphics that show module and port status

Product Description

HUBwatch for Windows is a tool for managing Digital's DEChub ONE family of products. It enables you to manage elements which support Management Information Base II (MIBII) as defined by the Internet standard RFC 1213. It also includes support for network elements which implement the following proposed MIBII groups:

- RFC 1286—Bridge MIB

- RFC 1316—Character Stream Device MIB
- RFC 1317—RS-232 Interface Type MIB

Note

For detailed information about Simple Network Management Protocol (SNMP), refer to *The Simple Book* by Marshall T. Rose.

HUBwatch for Windows provides assistance in the following areas:

Provides Assistance In...	By...
Network configuration	Providing you with the capability to add, delete, modify, and move network elements on the map. You also have the option to enter information about the devices that are attached to each port of a DECRepeater 90T or DECRepeater 90C. Free-form ASCII text can also be associated with each network element.
Isolating network faults	Providing the capability of displaying error counters about each SNMP-manageable device. HUBwatch supports alarms based on SNMP traps and threshold violations. Logs are for alarm records and configuration changes.
Monitoring performance	Assisting the user by displaying traffic-related counters for each SNMP-manageable device. Preconfigured reports can be generated for configuration, fault, and error counters.

You can double click on any of the modules in the hub view to access several windows that display the following module information:

Module	Information
DECbridge 90	Displays operating status, learned forwarding entries, Spanning Tree Protocol parameters, counter information, individual port data. You can also control the bridge's filtering operations.
DECserver 90L and DECserver 90L+	Displays general module information, collective port status, services information, and individual port setup.
DECRepeater 90C and DECRepeater 90T Ethernet repeaters	Displays information about the devices attached to a port. You can control and monitor ports one at a time or all at once.

Module	Information
DECagent 90	Displays general module information and the DECagent 90 community tables. You can also directly manage the DECagent 90 module.

In addition, you can add selected module types to a new hub. A new community can be either a standalone DECbridge 90 Ethernet bridge or a DECserver 90L or DECserver 90L+ terminal server, or it can contain a DEChub 90 backplane configured either as an 8- or 16-slot (2 hubs daisy-chained) hub. The DECserver 90TL and the DECwanrouter 90 can be added as standalone devices.

2

Installing HUBwatch for Windows

Introduction

This chapter provides information for installing HUBwatch for Windows. The following topics are included in this chapter:

- Hardware Requirements
- Software Requirements
- HUBwatch Kit Contents
- Installing HUBwatch

Hardware Requirements

The following hardware is required to install HUBwatch for Windows:

- A 386 or 486 processor running at a minimum of 16 megahertz. Performance on a 16-megahertz processor is slow; a processor with 25 megahertz or more is strongly recommended.
- A minimum of 4 megabytes of random-access memory (RAM). Additional memory improves performance.
- A 3½-inch 1.44-megabyte diskette drive.
- A minimum of 20 megabytes of available disk space.
- A printer supported by Windows — HUBwatch supports all the devices listed in the Windows Printer Setup. If you use a driver that is not part of the Windows package, then you may need to install it as an unlisted device.
- A mouse that is compatible with Windows — It is recommended that you use a mouse with HUBwatch. If you do not use a mouse, then you won't have the point-and-click control over network elements.

- A network interface card (NIC) for inband communications — This card is dedicated to the link between the workstation and the network. If you plan to use other networking applications, then refer to Appendix B for more information.
- An Ethernet port.
- A color VGA monitor. (A super VGA color monitor is optional.)
- One or more DECagent 90 modules (optional).

HUBwatch should be able to accommodate any configuration that meets the hardware requirements. For details on specific devices and software packages recommended for Microsoft Windows, refer to the *Windows Version 3.0 Application Reference List and Hardware Compatibility List*.

Software Requirements

You need the following software to install HUBwatch for Windows:

- MS-DOS Version 3.3 or higher.
- Microsoft Windows Version 3.0 or 3.1.
- Clarkson packet driver for the Ethernet adapter board. The Clarkson packet driver for many common adapter boards is included with HUBwatch.

HUBwatch Kit Contents

The HUBwatch kit should contain the following items:

- Product registration card — Complete this card and return it to the address provided. This registers your copy of HUBwatch and ensures that you receive all relevant software upgrades.
- HUBwatch management station software on two 3½-inch 1.44-megabyte diskettes
A README.TXT file provides information about product features, installation, and packet drivers. The DRIVER.TXT file contains information about the packet driver interfaces supported by HUBwatch.
- *HUBwatch for Windows User Information* manual.

Installing HUBwatch

Follow this procedure to install HUBwatch for Windows:

1. Turn on your personal computer (PC) and run Windows.
2. Put the HUBwatch diskette in drive A (or drive B, as appropriate).
3. Choose the Run option from the Program Manager's File menu.
4. Enter the following in the Command Line field of the Run dialog box:

```
A:SETUP
```

5. Choose the OK button.

HUBwatch opens a dialog box that displays the HUBwatch application name and prompts you for the destination drive and directory.

6. Enter the drive name and directory path where you want to store the HUBwatch network management software.

If you choose the default, then the installation utility does the following:

- Creates a subdirectory on drive C called C:\HUBWATCH>.
- Decompresses the application files from the distribution diskette to the new directory.
- Creates a new application group called HUBwatch and adds the HUBwatch icon to the group.
- Informs you when installation is complete and offers you the choice of restarting Windows or returning to DOS.

Before you restart Windows to run HUBwatch you must configure the PC for the packet driver.

The Windows initialization file is modified during installation and the changes take place after you restart Windows.

Configuring Your Network Management Station for HUBwatch

You must configure your network management station before HUBwatch can communicate on the network. Follow this procedure to configure your network management station:

1. Insert HUBwatch diskette 2 in drive A.
2. Copy the following files to the C:\HUBWATCH> directory:

```
CUSTOM.EXE
```

PCUDP.EXE
PING.EXE
NETDEV.SYS

3. Copy the online help file, DECCORE.HLP, to the C:\HUBWATCH\CORE> directory.
4. Copy the appropriate packet driver for your NIC to the C:\HUBWATCH> directory.
5. From your network manager, obtain an Internet Protocol (IP) address for your PC.
6. Ensure that you have a NIC installed.
7. Install the packet driver.

You must install a packet driver so your NIC operates properly. Install the appropriate Clarkson packet driver in the HUBWATCH directory. Packet drivers are particular to each type of NIC. For more information about installing the Clarkson packet driver, refer to Appendix C.

8. Modify your AUTOEXEC.BAT file to load the NIC's device driver at startup.
Because modifying your AUTOEXEC.BAT file installs a different network driver than you are currently using, you may want to have two versions of the AUTOEXEC.BAT file: one for running HUBwatch and one for running the PC. Refer to Appendix B for an explanation.
9. Modify your CONFIG.SYS file to configure your PC at bootup.

Because modifying your CONFIG.SYS will configure your computer differently than for normal operation, you may want to have two versions of the CONFIG.SYS file: one for running HUBwatch and one for running the PC. Refer to Appendix B for an explanation.

Modifying AUTOEXEC.BAT

In the AUTOEXEC.BAT file, add a line (after the line "ECHO off") to load the packet driver for your NIC. The following example applies to a DEPCA NIC:

```
C:\HUBWATCH\DEPCA 0X69 5 0X300 0XD000
```

This specifies software interrupt 69, irq 5, I/O address 300 and memory address D000.

For other NIC settings read the INSTALL.DOC on diskette 2.

Modifying CONFIG.SYS

In the CONFIG.SYS file, add a line at the bottom of the device list to load NETDEV.SYS. For example:

```
DEVICE=C:\HUBWATCH\NETDEV.SYS
```

Configuring NETDEV.SYS

During installation, NETDEV.SYS is loaded into the C:\HUBWATCH> directory. You need to customize the NETDEV.SYS file. Follow this procedure to customize NETDEV.SYS:

1. Change to the C:\HUBWATCH> directory and enter the following:

```
CUSTOM NETDEV.SYS
```

2. Press **[Return]**.

The PC Network Customizer menu appears followed by a list of defaults and menu options.

Note

When you choose a menu option use lowercase letters. Each single-letter command is obeyed immediately. You do not need to press **[Return]**. CUSTOM ignores invalid input.

3. Enter **h** at the Enter Command prompt to choose the Do hardware customizations option.

The Hardware Customization menu appears:

- a. Enter **b** to choose the Set net interface CSR base I/O address option.

The following prompt appears:

```
Enter the base I/O address for the net interface in hex
```

- b. Enter the correct I/O address and press **[Return]**.
- c. Enter **v** to choose the Set net interface interrupt vector option.

The following prompt appears:

```
Enter the interrupt number for the net interface
```

- d. Enter the interrupt number and press **[Return]**. The interrupt number should be the same number that you entered when you modified the AUTOEXEC.BAT file.

The interrupt number is reflected in the list of settings at the top of the screen.

e. Press **[Esc]** to return to the PC Network Customizer menu.

4. Enter **s** at the Enter Command prompt to choose the Do Site Customizations option.

The Site Customization menu appears with a list of defaults and menu options.

a. Enter **a** to choose the Set my internet address option.

The following prompt appears:

```
Enter my internet address
```

b. Enter the internet address and press **[Return]**.

The given internet address is reflected in the list of settings at the top of the screen.

c. Enter **s** at the Enter Command prompt to choose the Set number of subnet bits option.

The following prompt appears:

```
Enter the number of bits in the subnet field
```

d. Enter the number of subnet bits.

The number of subnet bits is reflected in the list of settings at the top of the screen.

e. If your system is connected to the Internet:

1. Enter **g** at the Enter Command prompt to choose the Set default gateway's address option, and enter the address.

2. Enter **d** at the Enter Command prompt to choose the Set my domain option, and enter the domain that the host is in.

f. Press **[Esc]** to return to the PC Network Customizer menu.

5. Enter **u** to choose the Set user's name option.

The following prompt appears:

```
Enter the user name.
```

6. Enter your user name and press **[Return]**.

The user name is listed at the top of the screen.

7. Enter **e** at the Enter Command prompt to exit and save the configuration.

Note

Make sure you modify all these fields. Do not leave out the Internet Address.

8. Reboot your network management station and monitor the screen for startup errors.

If you get a message similar to the following and the address is *not* all zeros (0s), then the Network Interface Card (NIC) is set up correctly. If the address *is* all zeros, then you probably have the wrong driver for your NIC.

Packet driver is at segment xxx

My Ethernet address is XX:XX:XX:XX:XX:XX

9. To verify the changes, go to the C:\HUBWATCH> subdirectory and run Ping. The syntax is

PING -T *target internet address*

(Q will quit Ping.)

Note

You can use a DECagent 90 Internet address as a test. You cannot Ping the PC on which HUBwatch has been installed.

If Ping works, your setup is correct. Otherwise, recheck your setup using the previous steps.

Note

Ping must work before HUBwatch will work.

10. Run Reset Database once to ensure that the database subdirectories are clean. Follow this procedure to ensure the subdirectories are clean:
 - a. Open the HUBWATCH window.
 - b. Double click on the Reset Database icon.

c. Answer yes to all the questions.

HUBwatch and the packet drivers are installed and ready to run.

11. Open the HUBwatch window and double click on the NMA icon.

Note

If HUBwatch aborts or becomes unstable, click on the Reset Data Base icon. If that does not work, click on the Rebuild Data Base icon. Be aware that these actions will destroy your network data.

3

HUBwatch Tools for Mapping Networks

Introduction

This chapter describes the graphic tools HUBwatch uses to help you map and manage your network within the GUI. The following topics are included in this chapter:

- Icons
- Pointers
- Network Connections
- Background Maps
- Notes

Icons

Each type of MIB device, including hubs, bridges, routers, hosts, terminal servers, and generic devices, is represented by a unique icon. HUBwatch uses the following icons for mapping networks:

- Alarm
- Device
- Specialized network

Note

HUBwatch uses color to convey status information about network devices. Therefore, use caution when customizing the color scheme. Some color schemes diminish the presentation of icons, help text, or alarm information. If you are unsure of a color scheme, use the Windows Version 3.0 default. Refer to the Windows documentation for information on modifying the color scheme.

Alarm Icons



Alarm icons indicate whether or not there is a problem with a device on the network. HUBwatch continually updates the alarm icon so it always reflects the severity level of the most serious alarm. The alarm icon is designed like a traffic light. The alarm color indicates whether or not there is an alarm and the severity of the alarm. The following table lists the alarm colors and what they indicate:

Alarm Color	Indicates . . .
Red	A major network alarm.
Yellow	A minor alarm.
Green	No alarms are active on the selected network.

At startup, the alarm icon is in the upper right corner of the HUBwatch display. You can move it to any location in the window.

To display alarm information, double click MB1 on the Alarm icon. This instructs HUBwatch to open the Current Alarms Network Wide window. You can scroll through the alarms or choose the Go To button to move immediately to the source of the selected alarm.

Device Icons

Device icons represent the different elements within the network. There are eight device icons:

- Site
- Bridge
- Router
- Host
- Hub
- Repeater
- Terminal server
- Generic device

The background color of a device icon indicates the alarm and the polling status of the associated device. The following table lists the device icon colors and what they indicate:

Icon Color	Indicates . . .
Green	Normal operation.
Red	The device has signaled a major alarm or the communication path is down.
Yellow	The device has signaled a minor alarm condition.
Blue	Polling is disabled on a device.
Gray	Uninitialized devices are on the network.

Site Icon



Site icons represent network sites. Sites are network elements incorporated as a group. The organization of network elements into sites is arbitrary. Do whatever is convenient to your task.

Bridge Icon



Bridge icons represent any bridges that are on your network. A bridge is a protocol-independent network device that provides for the exchange of packets between the physical networks it connects.

For example, a bridge might be used to divide large networks into smaller, interconnected segments. These can provide a measure of network traffic control by filtering the packets that pass between networks. It can also serve as a tool for isolating troublesome regions of a network. For example, collisions and corrupted packets can be blocked and faults can be isolated to one side of the bridge.

Router Icon



Router icons represent any routers that are on the network. A router is a protocol-dependent network device that provides for the exchange of packets between the physical networks it connects.

Routers are often used to provide a connection between networks with different protocols. They must be able to understand each of these protocols. Routers also receive and buffer entire packets before forwarding them.

Host Icon



Host icons represent any hosts on the network. A host is a service device such as a personal computer, workstation, or file server.

This classification of device is for the convenience of the network manager who might need to designate a device to serve some broad function within the network. For example, a host might be a device configured as a file server on a larger network. Such a device requires its own built-in or peripheral Ethernet interface with an SNMP agent designed into the interface or running as a process.

Hub Icon



Hub icons represent any hubs on the network. A hub is an enclosure that contains modular network devices.

The DEChub 90 contains eight module slots. You can connect two DEChub 90 hubs together to form a virtual 16-slot hub.

Repeater Icon



Repeater icons represent any repeater on the network. HUBwatch Version 1.0 supports the DECRepeater 90T (9-port repeater) and DECRepeater 90C (7-port repeater). The DECRepeater is designed to work either as a standalone unit or as a managed repeater in the DEChub 90 backplane.

Terminal Server Icon



Terminal server icons represent any terminal servers on the network. A terminal server is a device that allows a terminal or printer to connect with the local area network (LAN).

Generic Device Icon








Generic device icons represent any SNMP-manageable device on the network.

Use the generic device icon for all network elements of unspecified type that offer access to network management information as specified in the MIBI or MIBII specifications.

Specialized Network Icons

Specialized network icons are available through the Other option under the Add submenu on the Network pull-down menu. Specialized network icons are used to represent any kind of specialized network element and to identify the type of link between two simple network devices. The following table shows the connection icons and indicates their purpose:

Icon	Identifies . . .
	An Ethernet cable connection between network elements.
	A connection to a wide area network.
	A connection to a Public Data Network using the CCITT X.25 protocol.
	A connection to a Token Ring element in the network.
	A connection to an FDDI element in the network.

Pointers

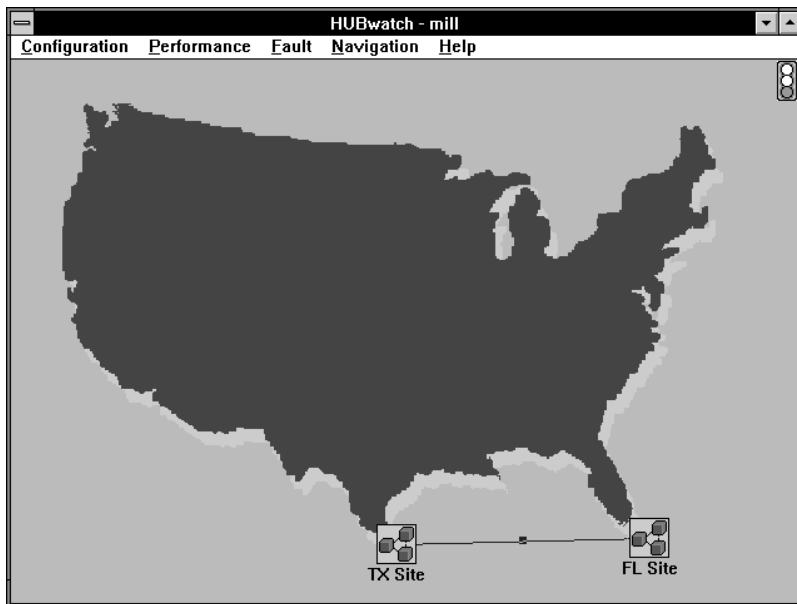
HUBwatch pointers are icons that often indicate a pending operation. Typically, you can abort pending operations by clicking on the pointer for the invalid operation. The following table describes all the pointers used within the HUBwatch application:

Pointer	Description	Purpose
Arrow	Shaped like an arrow	Used to select elements on the screen and to indicate choices in dialog boxes. This pointer changes to other pointers depending on the operation being performed.
Zoom	Shaped like a magnifying glass	Used to look at an object in more detail. It replaces the arrow pointer when you double click MB1 on the Zoom In option in the Navigation menu. To obtain a closer view of an object, position the magnifying glass on the object then press and release MB1.
Delete object	Shaped like a pair of scissors	Used to delete objects from the network. It replaces the arrow pointer when you select the Delete option from the Configuration menu. To delete an object from the Network database, position the scissors on the object then press and release MB1.
Add network connection	Shaped like a pencil with the point up	Used to make connections between objects. It replaces the arrow pointer when you select the Add Connection option from the Configuration menu. To add a connection between two network elements: <ol style="list-style-type: none"> 1. Position the pencil point on one element then press and release MB1. 2. Position the pencil point on another element then press and release MB1.
Delete network connection	Shaped like a pencil with the eraser up.	Used to delete a connection between two objects. It replaces the arrow pointer when you select Delete Connection option from the Configuration menu. To delete a connection between two network elements, click the eraser on the box (hotspot) marking the midpoint of the connection.
Activity	Shaped like an hour glass	Used to show that an activity is taking place. It replaces the arrow pointer when you make a request. It remains in the hour glass format until the request is completed.

Network Connections

Physical network connections are represented by a line drawn between two network elements. You can add and delete connections, but they are not manageable objects. Figure 3–1 shows a connection between two sites.

Figure 3–1 Example of Network Connections



LJ-02667-SIX

Background Maps

You can use background maps (.BMP files) to show geographical areas, administrative regions, building floor plans, or simple single-color backgrounds that denote different administrative sites.

You can also use geographical maps to depict sites that are scattered over a wide area. You can import any .BMP file to add a wide range of sophistication to your backgrounds.

For each network or site in the database, you can select a background map from the Background list in the Add Site dialog box (Figure 5–2).

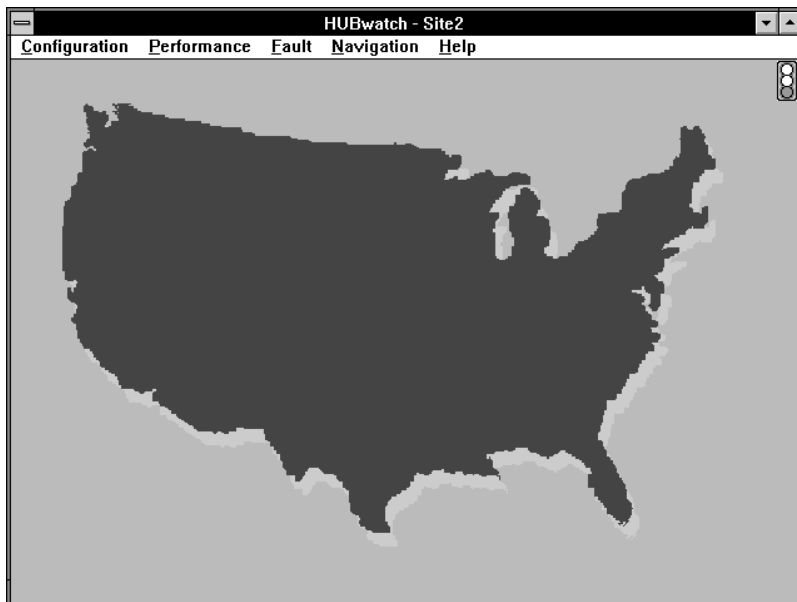
The list consists of graphics files in .BMP format. You can create these .BMP files using the Paintbrush utility in Windows or export them as .BMP files from another drawing application. Format conversion programs are also available that convert other files to .BMP format. Refer to the Windows documentation for more information. Maps are also available from other vendors. Refer to Appendix D for a list of vendors.

Figure 3–2 is a map of continental United States that was prepared with a third-party utility and saved as a .BMP file.

Note

The .BMP files are in the C:\HUBWATCH\CORE\BACKGRND> directory.

Figure 3–2 Example of Background Map



LJ-02662-SIX

Notes

The Notes option is available at every level of the HUBwatch application. This option enables you to make an annotation anywhere within the application and display a visual marker to access that information.

Follow this procedure to use the Notes option:

1. Pull down the Configuration menu and choose the Notes option.
The cursor becomes a stick pin.

2. Drag the stick pin next to the item for which you need to make a note. (The stick pin may slide behind certain objects.)
3. To enter a note, double click MB1 on the stick pin.
The HUBwatch Note Pad dialog box appears.
4. Enter whatever information you need into the dialog box and choose the Save button.
The information is saved, the dialog box is dismissed, and the stick pin remains on the screen as a marker for the message.

Note

The Cancel button at the bottom of the HUBwatch Note Pad dialog box removes the dialog box but leaves the stick pin. The Delete button removes the dialog box and the stick pin.

4

Using HUBwatch for Windows

Introduction

This chapter describes how to use the HUBwatch for Windows application. The following topics are included in this chapter:

- Starting a HUBwatch Session
- HUBwatch Views
- HUBwatch Menus
- Window Buttons
- Navigating Within HUBwatch

Starting a HUBwatch Session

How you start a HUBwatch session varies depending on whether you are opening a new network or an existing network.

If you are opening . . .	Then . . .	Result
A new network	Choose the <u>N</u> etwork <u>N</u> ew option from the <u>C</u> onfiguration menu.	The New Network dialog box appears. Refer to Adding Objects at Network or Site View for more information.
An existing network	Choose the <u>N</u> etwork <u>O</u> pen option from the <u>C</u> onfiguration menu.	The Open Network dialog box appears.

HUBwatch Views

After your network structure is fully mapped, you can move to different views to look at a particular hub or device in more detail. The views are described in the following table from the top level down.

View	Description
Network	This view is the top-level view. It is distinguishable from the site view in that contains the Security menu. The mapping of the network is the decision of the network administrator. The view may reflect devices as they exist in the physical network or it may reflect a worldwide collection of sites. The network view is the platform for conducting operations that affect the network as a whole. From this view, you can zoom into a hub view or site view.
Site	This view is one level down from the network view. For practical purposes such as network monitoring and management, the site view is equivalent to a network view. Sites are primarily used to organize networks into logical groupings often representing physical proximity such as a floor or building. Because they are administrative or logical rather than physical entities, no special configuration information is required in defining a site. Alarm information is not specifically associated with sites, but rather with the devices they contain. Also, sites can contain other sites down to four levels.
Hub	This view displays the complete hub and any modules within the hub slots. At this view, HUBwatch provides access to network management information as specified in the MIB specifications.
Module	This view is applicable only to devices within a DEChub. The DEChub modules are located at the hub view. To manage one of the modules within a DEChub, you must zoom in, or double click MB1 on that module. HUBwatch displays the module view with the device-specific menus which you use to manage the selected module.
Device	This view is similar to the module view, but it applies to devices outside the hub. The icon for the standalone devices are at the network or site view. To manage a standalone device, you must zoom in or double click MB1 on the standalone device. HUBwatch displays the device view, with the device-specific menus which you use to manage the selected device.

HUBwatch Menus

HUBwatch menus have seven basic options. The majority of the options are located at all views of the application and some are located only at specific views. Depending on the view, the pull-down menus vary. The following table lists the options, provides a description, and lists the views where the options are located. Refer to Appendix A for a map of the HUBwatch menus.

Option	Description	Level
<u>C</u> onfiguration	Provides options for changing, adding, deleting and obtaining configuration information on SNMP-manageable devices.	All
<u>P</u> erformance	Provides usage statistics, polling controls and reports of current statistics or historic usage.	All
<u>F</u> ault	Provides error statistics, alarm viewing and controls, threshold settings, diagnostics, and reports on alarms and statistics.	All
<u>N</u> avigation	Includes options for changing views and for finding a specific device or name.	All
<u>S</u> ecurity	Changes network and diagnostic passwords and obtains reports for password violations.	Network (top level)
<u>M</u> IB Access	Provides options for monitoring and setting MIB-related information for SNMP-manageable devices.	Hub and Device
<u>H</u> elp	Provides useful online information on all elements of HUBwatch.	All

Note

Menu options that are not available in the current context are dimmed in both the menu bar and the menu itself.

Buttons

Some of the HUBwatch windows have buttons. These buttons enable specific actions related to the specific screen. The following table describes the buttons in the HUBwatch application.

Button	Purpose
<u>A</u> ck	Acknowledges selected alarms on the Current Alarms Network Wide window and turns it from red to black.
<u>C</u> ancel	Clears any information entered into a dialog box and removes the dialog box.

Button	Purpose
<u>C</u> lear	Clears the selected alarms listing on the Current Alarms Network Wide window.
<u>D</u> isable All	Disables all Alarm Threshold settings on the Set Threshold windows for a selected module.
<u>E</u> nable All	Enables all Alarm Threshold settings on the Set Threshold windows for a selected module.
<u>E</u> xit	Dismisses the selected screen without accepting any changes.
<u>G</u> et	Displays information for the defined Object Identifier on the MIB Extensions window.
<u>G</u> et Next	Displays information for the object following the defined Object Identifier on the MIB Extensions window.
<u>G</u> o To	Displays the device view for a selected alarm listed on the Current Alarms Network Wide window.
<u>H</u> elp	Invokes the help utility.
<u>M</u> odify	Enables changes to the MIB.
<u>O</u> K	Accepts screen modifications and cancels the window.
<u>O</u> utput	Displays the Output dialog box enabling specifications for printing a report.
<u>P</u> ing	Counts the number of packets received and transmitted as defined on the Ping window.
<u>R</u> eset	Sets screen counters to their defaults.
<u>R</u> estore	Sets screen settings back to defaults.
<u>S</u> elections	Enables you to define the parameters for a specified selection such as alarms or devices.
<u>S</u> et Default	Sets thresholds to the default settings on the Set Thresholds window.
<u>S</u> top	Stops the display of the MIB on the MIB Extensions window.
<u>W</u> alk	Enables the system to continuously display the MIB on the MIB Extensions window.

Navigating within HUBwatch

Use the Navigation menu to move from view to view. When you select the Navigation menu, the following options appear:

- Find
- Find Me
- Zoom In

- Zoom Ot
- Zoom Top

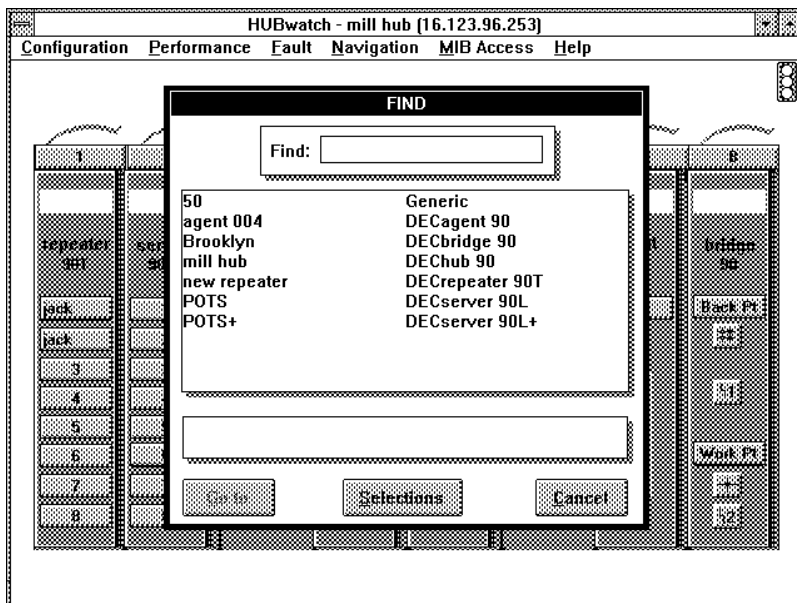
The Find Option

The Find option is a quick way to access a specific device.

1. Choose the Find option from the Navigation menu.

The Find window appears (Figure 4–1).

Figure 4–1 Find Window



LJ-02663-SIX

The Find window contains two columns. The left column alphabetically lists the names of the objects in the network. The right column lists the device type.

Use the scroll bar to search through the list, or type the first characters of a name to take you to that section of the list.

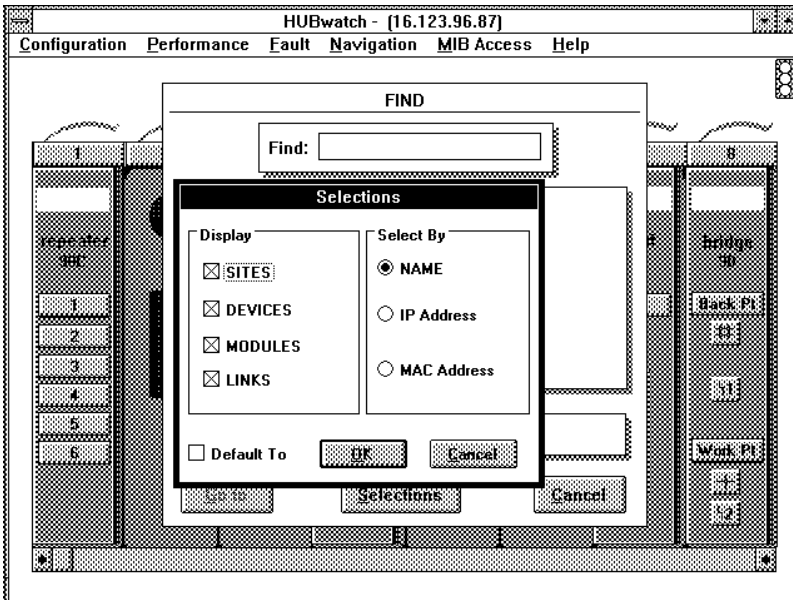
2. Select the device you need to look at.
3. To display the selected object, choose the Go To button.

or

To limit the selection of the objects listed in the Find window, follow this procedure:

- a. Choose the Selections button in the Find window.
The Selections dialog box appears (Figure 4-2).
- b. Click on one of the following to choose the object:
 - NAME
 - IP Address
 - MAC Address
- c. To list the objects as selected, choose the OK button.

Figure 4-2 Selections Dialog Box



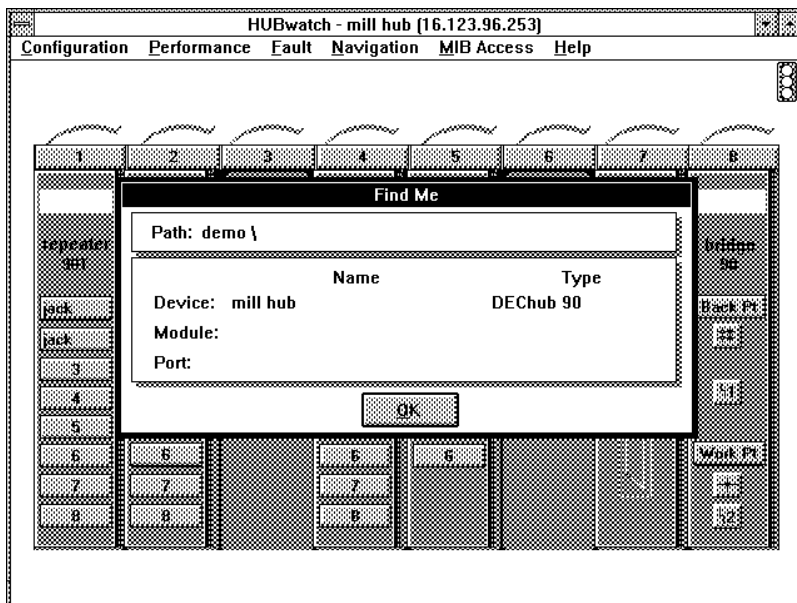
LJ-02664-SIX

Find Me Option

The Find Me option enables you to display the path to the current object. Pull down the Navigation menu and choose the Find Me option. The Find Me window appears (Figure 4-3). It displays the following information about the selected object:

- Path
- Device name and type
- Module name and type
- Port

Figure 4-3 Find Me Window



LJ-02665-SIX

Zoom In Option

The Zoom In option enables you to move to the next view for more detail.

1. Choose the Zoom In option from the Navigation menu.

The pointer changes to a magnifying glass.

2. Position the magnifying glass on the object you want to look at and click MB1.

A more detailed view of the object appears.

Note

Another way to zoom in on an object is to position the pointer on the object and double click MB1.

Zoom Out Option

The Zoom Out option enables you to move up from a more detailed view of the network to a less detailed view of the network. To move up one view, choose the Zoom Out option from the Navigation menu.

Note

Another way to zoom out is to position the pointer anywhere within the present view and double click MB2.

Zoom Top Option

The Zoom Top option enables you to quickly move from any lower view to the top view, eliminating all interim views. To move directly to the top view, choose the Zoom Top option from the Navigation menu. The top view hierarchy appears.

Printer Setup

Several of the reports, at the module view, provide a Print button that enables you to print a copy of the report. Before you can do this, you must configure a printer to the application. To fully configure a printer, you must enter information into the following windows:

- *default printer name*
- Options
- Send Header

- Advanced Options

For detailed information, refer to the *Microsoft Windows User's Guide*.

5

Configuring Networks

Introduction

Using HUBwatch, you can create a graphic user interface (GUI) through which you can monitor and manage your network. This is primarily done through the HUBwatch Configuration menu.

This chapter explains how to use the HUBwatch Configuration menu to build a configuration of your networking system within the HUBwatch application. The following topics are included in this chapter:

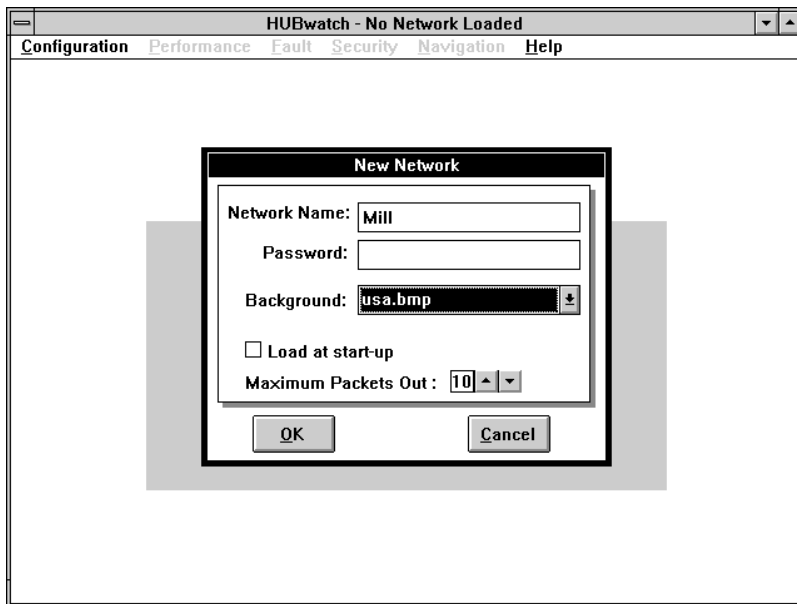
- Creating a New Network
- Opening an Established Network
- Backing Up a Network
- Restoring a Network
- Deleting a Network
- Modifying a Network

Creating a New Network

The first step in configuring a network is to create the network. Follow this procedure to create a network:

1. Choose the New option from the Configuration menu.
The Network submenu appears.
2. Choose the New option.
The New Network dialog box appears (Figure 5-1).

Figure 5–1 New Network Dialog Box



LJ-02666-SIX.PS

3. Enter the following information as needed:

Field	Description
Network Name	Name of the network you are opening.
Password	A password to access the network. This is not required, but if entered, you will always be prompted for it when opening that network.
Background	The window background for the network. For example, this may be a map of a state, country, or a building.

4. Click on the Load at start-up box, if you want this network to automatically load at startup.
5. Set the Maximum Packets Out.
The default is 10. Maximum Packets Out controls the number of SNMP packets HUBwatch will put on the network each second.
6. Choose the OK button.
The Network window appears with the name of the network on the banner.

Now that you have established a network, you are ready to configure it. To configure the network, do one or more of the following:

- Add sites, devices, or connections
- Delete devices or connections
- Move network elements
- Modify views or devices

Adding Objects at Network or Site View

You can add the following objects at the network or site view:

- Sites
- Devices
- Connections
- Other (graphic connection types)

Note

You can also add devices automatically using the Auto Discovery option under the Network menu. See *Using Auto Discovery* for more information about the Auto Discovery option.

Adding Sites

Follow this procedure to add sites to your network.

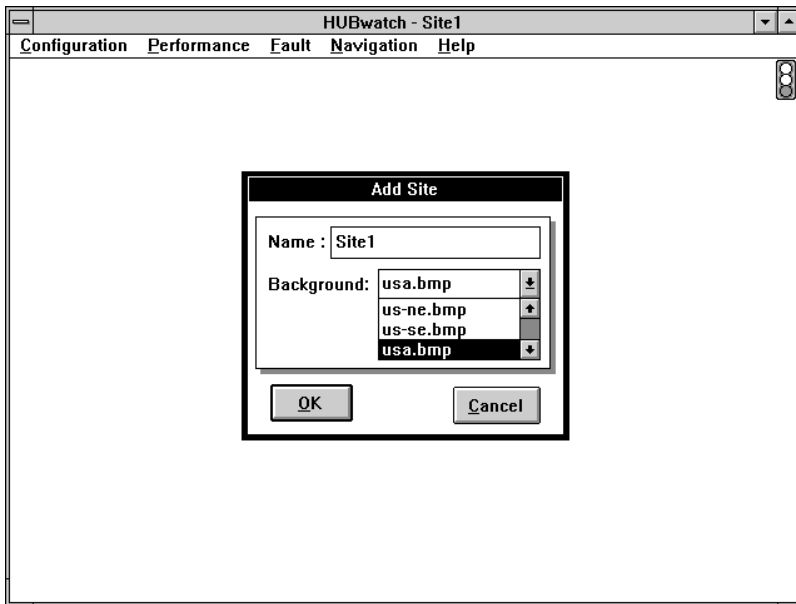
1. Choose the Add option from the Configuration menu.
The Add submenu appears.
2. Choose the Site option from the Add submenu.
The Add Site dialog box appears (Figure 5-2).
3. Enter the following information in the Add Site dialog box:

Field	Description
Site Name	The user-defined name for the site.
Background	The selected background view (.BMP) for the site.

4. Choose the OK button.
A site icon attaches itself to the cursor.

5. Drag the icon to the desired location and then click the mouse button to release the icon.
HUBwatch anchors the site icon and displays the site name.

Figure 5–2 Add Site Dialog Box



LJ-02661-SIX

Note

Sites can be nested four deep.

Adding Devices

You can add a device at the network, site, or hub view. To add a device at the network or site view, you must enter device information into the following windows:

- Add Device
- Add *device name*

- Community
- Device System Group

Note

To add a device at the Hub view, refer to Adding Objects at the Hub View.

Follow this procedure to begin the add device process:

1. Choose the Add option from the Configuration menu.
The Add submenu appears.
2. Choose the Device option from the Add submenu.
The Add Device window appears.

Add Device Window

The Add Device window (Figure 5-3) enables you to define the type of device you need to add to the network configuration. You can add any of the following Digital devices:

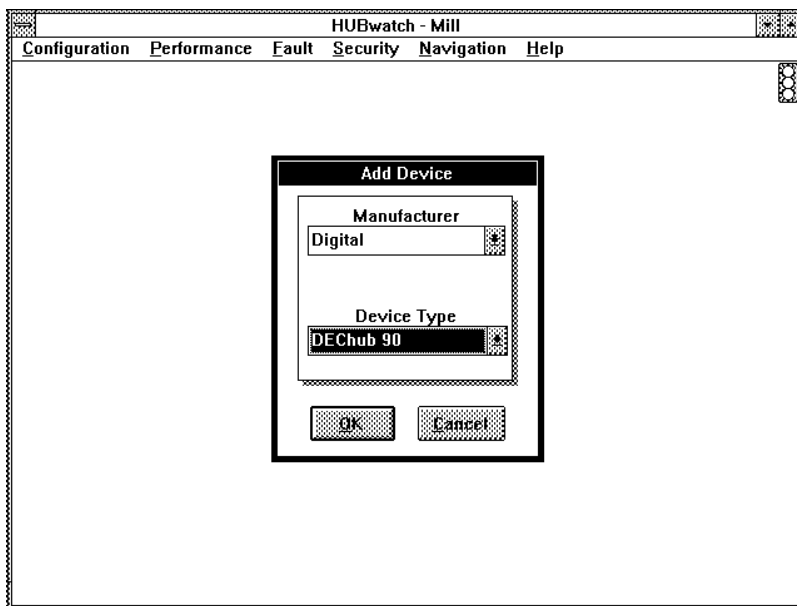
- DEChub 90
- DECbridge 90
- DEChub 90 (16)
- DECagent 90
- DECrepeater 90
- DECserver 90L
- DECrepeater 90C
- DECserver 90L+

You can also add the following generic device:

- Bridges
- Generic devices
- Routers
- Hosts

Use generic devices to add a DECserver 90TL, DECserver 90M, or DECwanrouter 90. Refer to Chapter 7 for more information about generic devices.

Figure 5-3 Add Device Window



LJ-02668-SIX

Follow this procedure to define the type of device you are adding:

1. Enter the following information in the Add Device window:

Field	Description
Manufacturer	The manufacturer of the device you are adding.
Device Type	The type of device you need to add such as repeater, server, or router.

2. Choose the OK button.

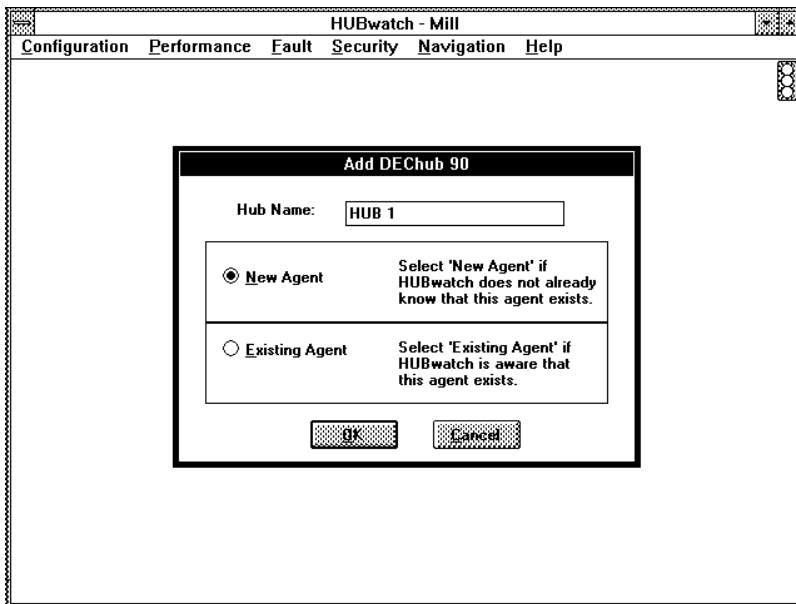
The Add *device name* window appears.

Add *device name* Window

The Add *device name* window varies depending on the type of device you select.

If you add a . . .	Then a window similar to this appears . . .
DEChub 90	Add DEChub 90 window (Figure 5-4)
Module	Add DECagent 90 window (Figure 5-5)

Figure 5-4 Add DEChub 90 Window

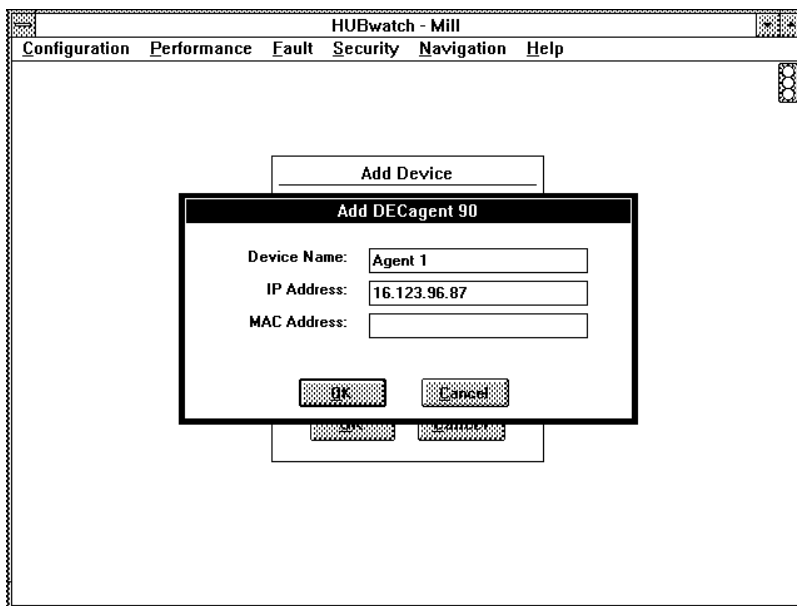


LJ-02669-SIX

Follow this procedure to define the device you are adding:

1. Enter the following, depending upon the selected device.

Figure 5-5 Add DECagent 90 Window



LJ-02670-SIX

Field	Description
New Agent	Select if HUBwatch does not know about the agent.
Existing Agent	Appears only if you have previously added an agent. Select if you want to use one of these agents.
IP Address	HUBwatch makes use of both the Internet Address and the MAC or physical address in communication with the device. Therefore, to be managed, the device must know its own Internet Address.
Device Name	This string (0-20 characters) is always displayed with the icon for the managed device. Because HUBwatch lists devices by name in the Find and Find Me lists, a unique string is recommended.
MAC Address	HUBwatch can automatically locate the MAC address if it has the IP address. If there is no IP address, then enter the MAC address.

2. Choose the OK button.

If you added a . . .	Then . . .
Generic device	The cursor becomes an icon representing the generic device you are adding. Move the icon where you need to place it and click MB1 to anchor it. The device is added. To manage it, refer to Chapter 7.
Digital device	The Community window appears.

Community Window

The Community window (Figure 5–6) requests community string information about the device you are adding. A community (in the context of a Simple Network Management Protocol) is a set of attributes that are managed as a group. Normally, there is a one-community-to-one-agent relationship. The manageable attributes are usually contained within a single hardware device, or within a single enclosure in the case of hubs. The single hardware device, or the collection of devices within a hub, is treated as one community. A specific manageable entity is uniquely identified on the network by the combination of an IP address and a community string.

A community string is a sequence of ASCII characters that is checked by the SNMP agent for access control to the manageable entity. You can think of the community string as a password. There are two strings associated with a given community:

- Read-only
- Read/write

For a GET or a GET NEXT operation, the agent accepts either the read-only or read/write string. For a SET operation, the agent accepts only the read/write string. For more information about the agent and community strings, refer to the *DECagent 90 User Information* manual.

The Community window requests the following information about the device you are adding:

Field	Description
Agent Information	
Agent Name	A name you assign to the agent.
Agent IP	The IP address of the agent.
Agent MAC	The Medium Access Control address of the agent.
Community Information	
Read only	Read-only access. Each read-only string, at a given IP address, must be unique. However, to assist the HUBwatch auto discovery feature, you may want to leave the agent's read-only community string as "public." New communities could be given strings of "public1," "public2," and so on. The read-only string is case sensitive.
Read/Write	Read/write access. Each read/write string, at a given IP address, must be unique. The read/write string is case sensitive.

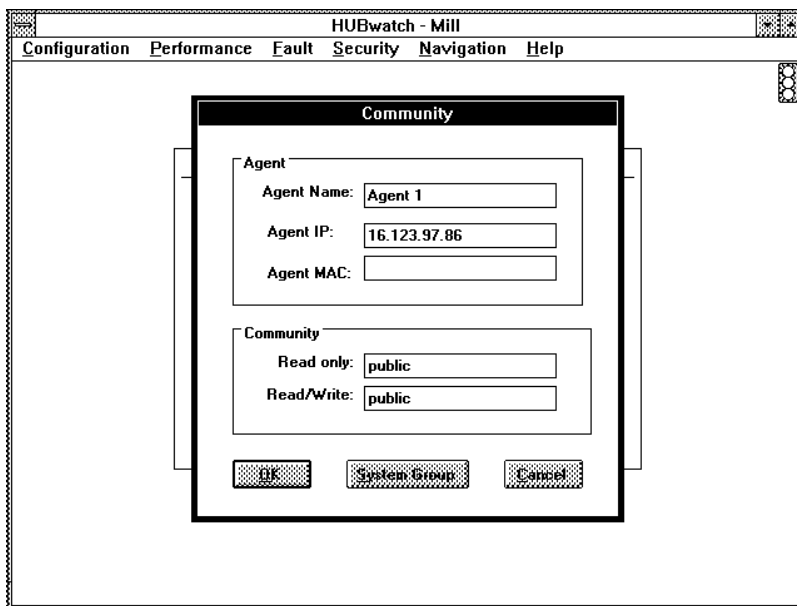
Follow this procedure to enter community information about the device you are adding:

1. Enter the appropriate information into the Community window.
2. Choose the OK button.
or
Choose the System Group button to display the Device System Group window to add detailed information about the device.

Device System Group Window

The Device System Group window (Figure 5–7) enables you to add the following detailed information about the device:

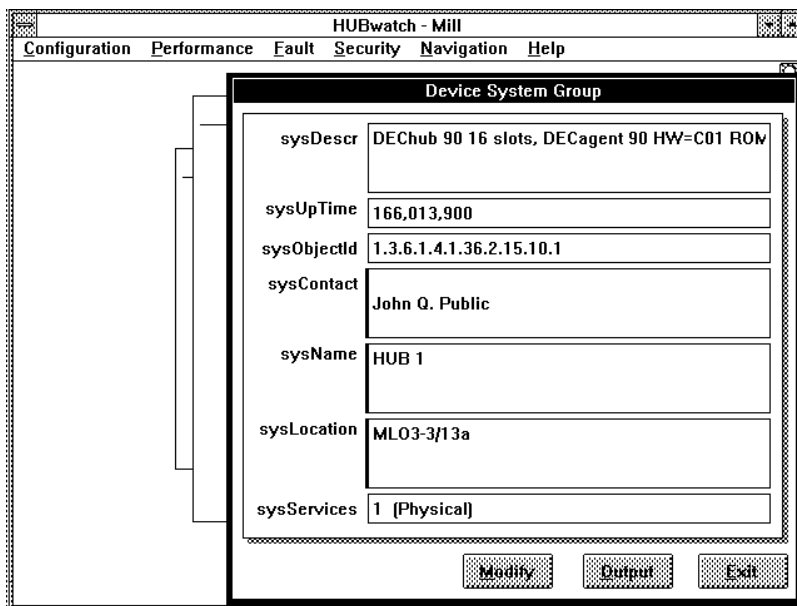
Figure 5–6 Community Window



LJ-02671-SIX

Field	Description
sysDescr	The agent or hub. Typical entries include: Software Hardware Firmware
sysUpTime	The total time that the agent has been running since it was last reset.
sysObjectId	The registered object ID (OID) for the object that is currently being managed.
sysContact	The person or group to contact for service.
sysName	The name of the agent or hub.
sysLocation	The physical location of the agent.
sysServices	An encoded description of the services provided by the agent.

Figure 5-7 Device System Group Window



LJ-02672-SIX

Follow this procedure to complete the information about the device:

1. Enter the appropriate information into the Device System Group window fields.
2. Choose the OK button.

The device is now added. To manage the device, refer to the following information:

If you added this module . . .	Then refer to . . .
DECagent 90	Chapter 8 in this document
DECbridge 90	<i>HUBwatch for Windows DECbridge 90 Management</i>
DECserver 90	<i>HUBwatch for Windows DECserver 90 Management</i>
DECrepeater 90	<i>HUBwatch for Windows DECrepeater 90 Management</i>

Adding Connections

You can show a graphic representation of actual connections between elements on the network and you can define that connection.

Follow this procedure to add a connection:

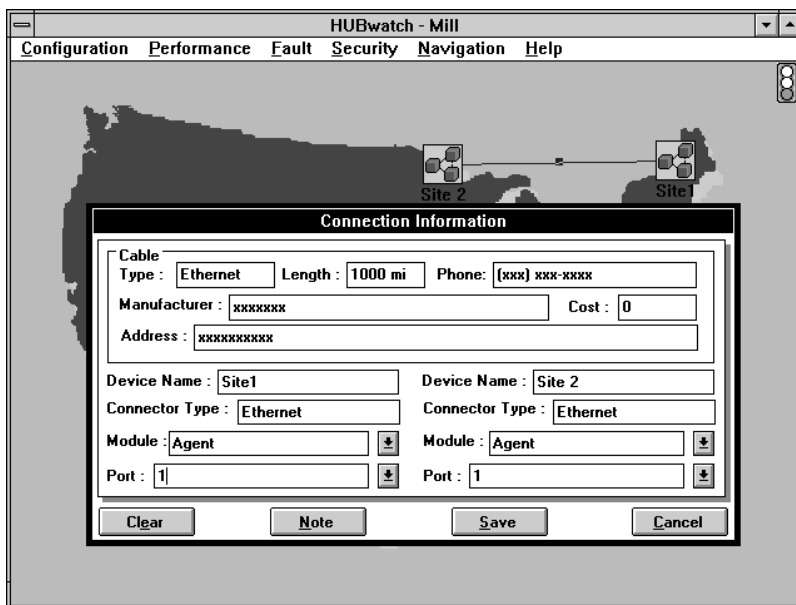
1. Choose the Add option from the Configuration menu.
The Add submenu appears.
2. Choose the Connection option from the Add submenu.
The mouse cursor changes to a pencil.
3. Position the pencil pointer on the device or site you want at one end of the connection and click MB1. Then position the pencil pointer on the device or site you want at the other end of the connection and click MB1 again.
A line appears between the two selected objects. The line has a small square in the center. This square is referred to as a **hotspot**.
4. Click on the hotspot to provide detailed information about the connection.
The Connection Information window appears (Figure 5–8).
5. Enter the following information into the Connection Information window:

Field	Description
Connection Information	
Type	The type of cable (such as Ethernet or WAN).
Length	The length of the cable (feet, miles, or meters).
Phone	The manufacturer's phone number.
Manufacturer	The name of the cable manufacturer or service provider.
Cost	The cost of the cable or leased line.
Address	The manufacturer's address.
Device Information For Each Device	
Device Name	This defaults from previously entered information.
Connector Type	The type of cable connection (such as WAN or Ethernet).
Module	The type of module (such as repeater, bridge, agent, or server).
Port	The port position of the module within the hub.

6. Choose the Save button.

The information is recorded.

Figure 5–8 Connection Information Window



LJ-02673-SIX

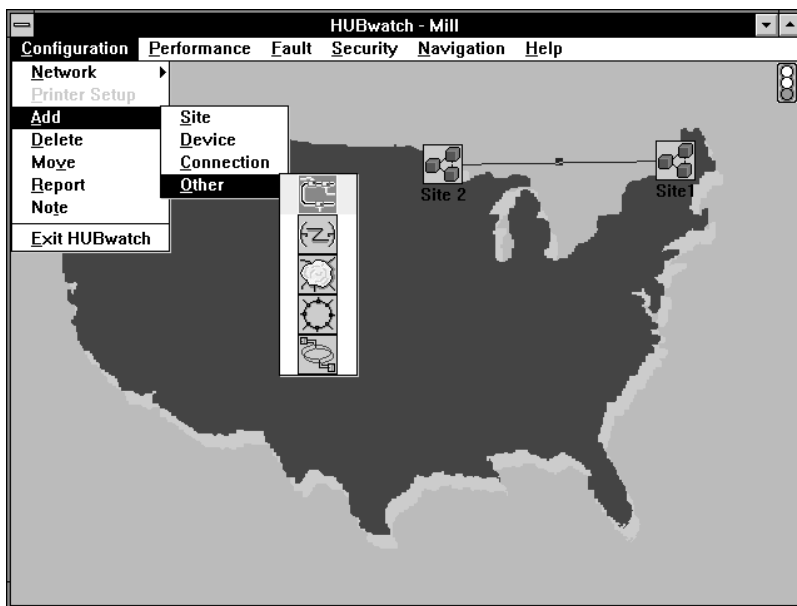
Adding Specialized Network Icons

You can use specialized network icons to define the types of connections that exist between the elements. These graphic connections are defined in the Specialized Network Icons section.

Follow this procedure to define the connections:

1. Choose the Add option from the Configuration menu.
The Add submenu appears.
2. Choose the Other option from the Add submenu.
A panel of special purpose icons appears (Figure 5–9).
3. Choose the desired icon to represent the connection.

Figure 5–9 Other Icons



LJ-02674-SIX

The Add window appears (Figure 5–10).

4. Enter the name of the connection type.
5. Choose the OK button.

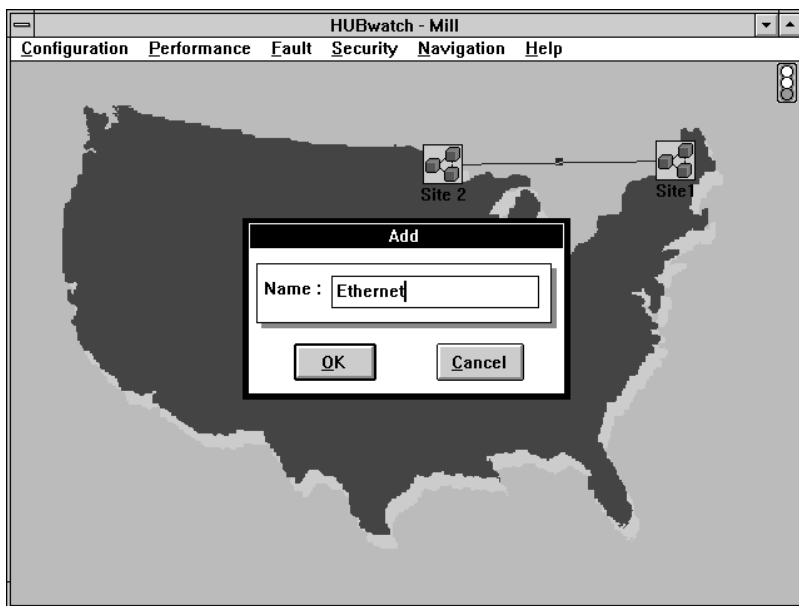
The cursor becomes an icon that represents the connection type. Drag the icon to the desired connection and click MB1. The icon is released and anchored in the defined position.

Using Auto Discovery

The Auto Discovery utility enables HUBwatch to automatically do the following:

- Find the whereabouts of all PING and SNMP-manageable devices on the network within a specified range of IP addresses.
- Record the whereabouts of the devices.
- Create an icon that represents the devices.

Figure 5–10 Add Window



LJ-02675-SIX

Follow this procedure to use the Auto Discovery utility:

1. Choose the Network option from the Configuration menu.
The Network submenu appears.
2. Choose the Auto Discovery option from the Network submenu.
The Auto Discovery window appears.
3. Enter the appropriate information into the following fields.

Field	Description
Starting IP Address	This is the Ethernet address from where you want to begin the search for all PING and SNMP-manageable devices.
Ending IP Address	This is the end of the range for which you want to search for the PING and SNMP-manageable devices. To enter the IP address into this field, press [Return] . The Starting IP Address appears in the Ending IP Address field. Use the mouse to drag the slider to reflect the Ending IP Address you need.

4. Choose the **OK** button.

HUBwatch does the following:

- Opens a Move window.
- Scans the network for all SNMP-manageable devices within the given IP address.
- Records any information about the device.
- Creates an icon that represents each device.
- Places the icon into the Move window.
- Displays the Notice dialog box indicating “Auto Discovery Process Complete.”

5. Move the icons to the network.

- a. Position the mouse cursor on the icon you need to move.
- b. Press **[Shift]** and hold MB1 while you drag the icon to the Network or Site window.
- c. Release the mouse button.

The Add *device name* window appears displaying information for the chosen device.

- d. Check the information. If the information is correct, then choose the **OK** button. If the information is not correct, then make the necessary changes.

The Community window appears displaying the agent name and the IP address.

- e. Change the read-only and read/write community strings as necessary. Each of these must be unique (within a single agent) to match what is stored in the selected device; otherwise, the agent will reject it.

- f. Choose the OK button.
- g. Continue this process until you move all the icons.
- h. After you move all the icons, close the Move window.
HUBwatch automatically initializes the database and displays the icons in green.

Adding Objects at the Hub View

At the Hub view, you can add only modules.

1. Set the view for the hub where you need to add the module.
2. Choose the Add option from the Configuration menu.
The cursor becomes a plus sign (+).
3. Position the plus sign on the hub slot where you need to add the device and click MB1.
The Add *device name* window appears.
4. Enter the name of the module you are adding and any other required information.
5. Choose the OK button. (Refer to the Adding Devices section for more information about adding devices.)
HUBwatch inserts the selected device into the selected slot.

Note

You can install only agent and bridge devices in a 12-volt slot (the two rightmost slots in a DEChub 90). For more information about the location of the devices, refer to the *DECagent 90 User Information* manual.

Deleting Objects at Network or Site View

You can remove the following objects from the network and site views:

- Devices
- Connections
- Other objects

Note

Because HUBwatch does not delete an object that has connections to other objects, you must first delete all connections to the object you want to delete.

Deleting Devices

Follow this procedure to delete devices:

1. Choose the Device option from the Delete submenu.
The pointer becomes a pair of scissors.
2. Position the scissors on the object you want to delete and click MB1.
The Notice dialog box displays the following message:
Connection exists, OK to delete?
3. To delete the device, choose the OK button.

Deleting Connections

Follow this procedure to delete a connection:

1. Choose the Connection option from the Delete submenu.
The pointer becomes a pencil with the eraser pointing up.
2. Position the eraser on the hotspot that is in the middle of the connection line you need to delete and click MB1.
The Delete Connection dialog box displays following message:
Are you sure?
3. Choose the Yes button.
The connection is deleted.

Note

The Delete Connection option deletes regular line connections. It does not delete graphic connections such as coaxial cable, wide area network, or X.25 network. Refer to the Deleting Other Icons (Specialized Network Icons) section to delete graphic connections.

Deleting Other Icons (Specialized Network Icons)

HUBwatch enables you to delete connectors such as a coaxial cable, WAN, or X.25 connectors represented by a graphic connection.

Follow this procedure to delete a graphic connection:

1. Choose the Delete option from the Configuration menu.
The Delete submenu appears.
2. Choose the Other option from the Delete submenu.
The pointer becomes a pair of scissors.
3. Position the scissors on the icon connection you need to delete and click MB1.
Depending on whether or not there are any connections to the object, one of the following occurs:

If ...	Then ...
There are no connections	The Delete Object dialog box displays this message: Are you sure?
The icon is connected to another icon	The Notice dialog box displays this message: Connection Exists, OK to delete?

4. To delete the icon and any connections, choose the Yes button.

Deleting Objects at Hub View

At the Hub view, you can only delete a module from the DEChub 90. Follow this procedure to delete a module from a DEChub 90:

1. Choose the Delete option from the Configuration menu.
The cursor becomes a minus sign (-).
2. Position the minus sign on the module you want to delete and click MB1.
HUBwatch displays a window with the following message:
Are you sure?
3. Choose the OK button.
The module is deleted from the hub.

Moving Objects

HUBwatch enables you to move objects as follows:

- Within a specific network view
- From one network view to another

Moving Objects within the Same Network View

Follow this procedure to move objects within a view:

1. Open the view from where you need to move the objects.
2. Position the cursor on the object you need to move. Press **[Shift]** and hold MB1 while you drag the object to the new location. Release MB1 to anchor the object.

All connections to the moved objects adjust (stretch, shrink, or move) automatically to follow the objects to which they are connected.

Moving Objects to Another Network View

The **M**ove option is used to move a network element, such as a site or device, from one site to another. It is available at the network and site views.

1. Open the view from where you need to move the objects.
2. Choose the **M**ove option from the **C**onfiguration menu.
The Move window appears at the bottom of the screen. The Move window is a holding place for any objects you want to move to another view.
3. To place the objects in the Move window, position the cursor on the object and press **[Shift]** and hold MB1 while you drag the object into the Move window.
4. Open the view where you are moving the objects to.
5. Move the desired objects from the Move window into the desired view.
Position the cursor on the objects you need to move, press **[Shift]** and hold MB1 while you drag the icons into the new view.
6. Repeat the two previous steps until all the objects are placed in the appropriate views.
7. When all objects are moved, you can either shrink the Move window to an icon or close the window. If you close it, items left in the window return to their original location.

Note

When a Move operation is pending, movement through the network hierarchy is restricted to network and site views.

Modifying Views

You can modify a network at any of the following views:

- Site
- Hub
- Device

Modifying Sites

Follow this procedure to modify a site:

1. At the site view, choose **M**odify from the **C**onfiguration menu.
The Modify Site dialog box appears (Figure 5–11).
2. Change the following as needed.

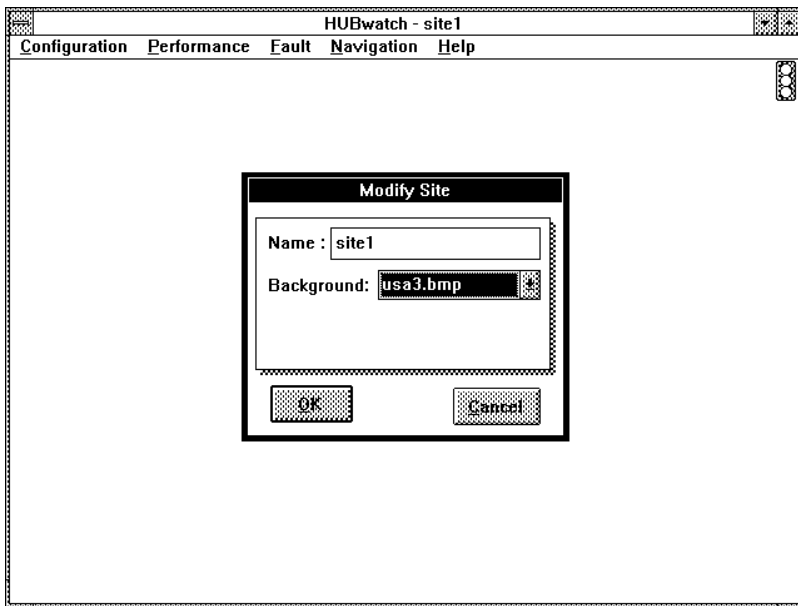
Field	Description
Name	The name of the site.
Background	The background (.bmp) for the site view.

Modifying Devices or Hubs

Follow this procedure to modify device or hub parameters:

1. Set your display to the device or hub view you need to modify.
2. Choose **M**odify from the **C**onfiguration menu.
The Modify Device window appears (Figure 5–12).
3. Change the following parameters as needed:

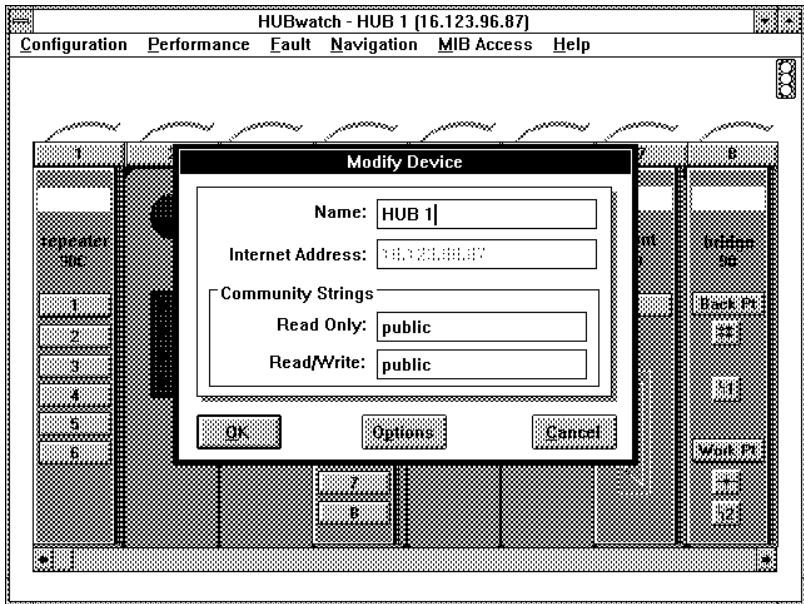
Figure 5–11 Modify Site Dialog Box



LJ-02676-SIX

Field	Description
Name	The name of the site.
Internet Address	The Ethernet address for the hub.
Read Only	Read-only access community string.
Read/Write	Read/write access community string.

Figure 5–12 Modify Device Window



LJ-02677-SIX

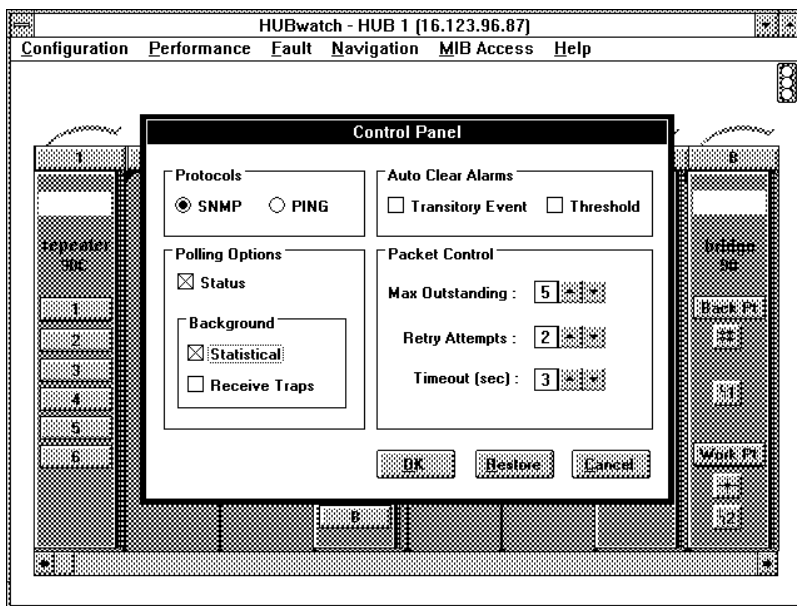
4. To accept the new parameters and remove the window, choose the OK button.
or
To make further modifications, choose the Options button.
The Control Panel window appears (Figure 5–13).

5. Change the following information as needed:

Field	Definition
Protocols	
SNMP	HUBwatch polls appropriate MIB objects in the selected device.
PING	HUBwatch polls only for connectivity using Ping.
Auto Clear Alarm	
Transitory Event	If the fault condition is temporary and this is checked, the alarm clears itself.
Threshold	If a count drops below the threshold and this field is checked, then the alarm clears itself.
Polling Options	
Status	HUBwatch polls for device status only. You can poll for Ping or SNMP.
Statistical	HUBwatch polls traffic statistics for a module. This is necessary for thresholding. HUBwatch requires that SNMP be selected for polling MIB objects.
Receive Traps	HUBwatch receives notification from the SNMP agent of a special event or condition.
Packet Control	
Max Outstanding	The number of SNMP packets that are sent before getting a reply from the agent.
Retry Attempts	The number of times HUBwatch will repeat a request without receiving a response.
Timeout (sec)	The length of time HUBwatch will wait for a response before it retries.

6. Choose the OK button to accept the new parameters or choose the Restore button to change back to the default parameters.

Figure 5–13 Control Panel Window



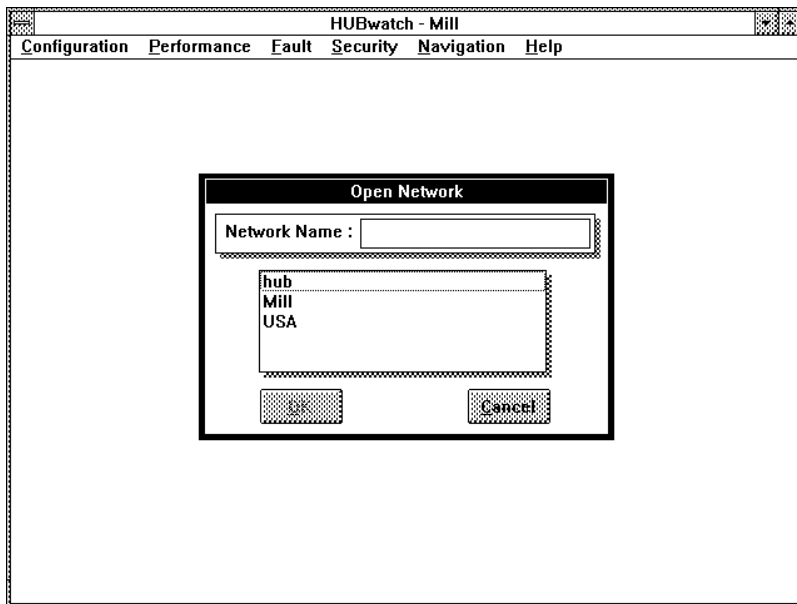
LJ-02678-SIX

Opening an Established Network

Follow this procedure to start an administrative session with an existing network:

1. Choose the Network option from the Configuration menu.
The Network submenu appears.
2. Choose the Open option from the Network submenu.
The Open Network dialog box appears (Figure 5–14). It lists all the existing networks.
3. Select the network you want to open and choose the OK button.
or
Double click MB1 on your selection from the Network list.

Figure 5–14 Open Network Window



LJ-02679-SIX

The Network view appears.

Note

If a password is required to open the Network, HUBwatch prompts for the required password before it opens the network view.

Backing Up a Network

Follow this procedure to back up a network:

1. Choose the Network option from the Configuration menu.
The Network submenu appears.
2. Choose the Backup option from the Network submenu.
The Network Backup screen appears (Figure 5–15).

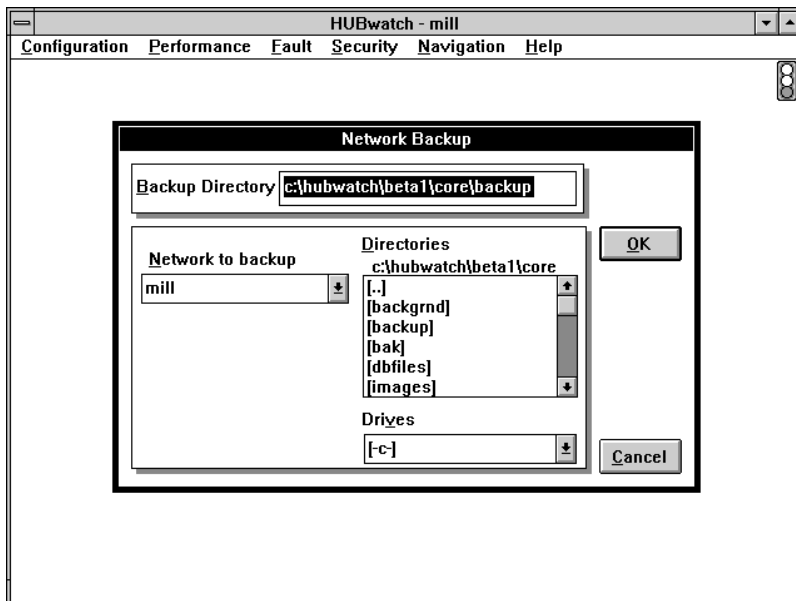
3. Enter information into the following fields:

Field	Description
Network to backup	Network configuration you need to back up the network.
Directories	Name of the directory in which the network is to be backed up.
Drives	Name of the drive where the network is to be backed up.

4. Choose the **OK** button.

HUBwatch begins the backup and keeps track of the last backup made.

Figure 5–15 Network Backup



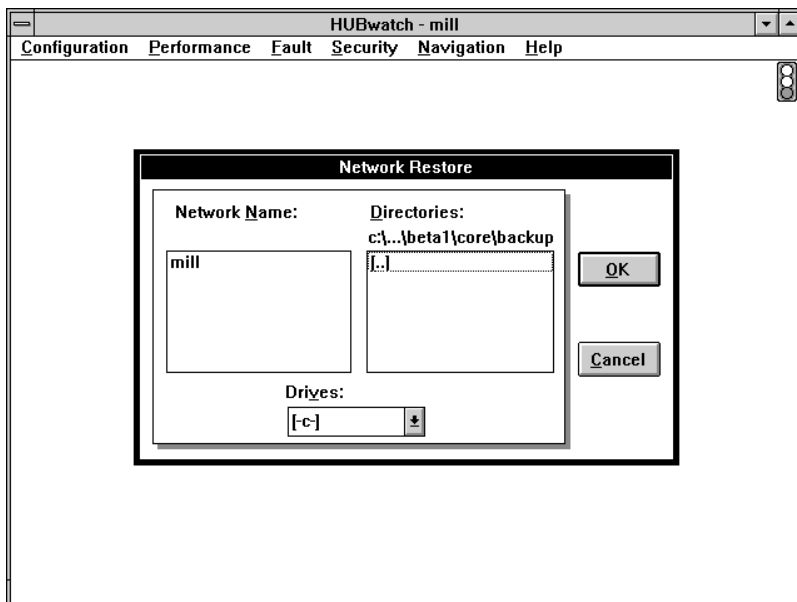
LJ-02680-SIX

Restoring a Network

HUBwatch recovers a network from the backup file. For information about how to back up a network, refer to the Backing Up a Network section. To recover the backup database for a specified network, follow this procedure:

1. Choose the Network option from the Configuration menu.
The Network submenu appears.
2. Choose the Restore option from the Network submenu.
The Network Restore window appears (Figure 5–16) and lists the following information:
 - Networks available for restoring
 - Backup files available for the specified network.

Figure 5–16 Restore Window



LJ-02681-SIX

3. Select the network and directory you need to restore.

4. Choose the OK button.

Note

You can restore multiple networks at one time. If you elect to restore the current network, HUBwatch closes the network, restores the configuration from the specified backup, and then reopens the network.

Deleting a Network

Follow this procedure to delete a network:

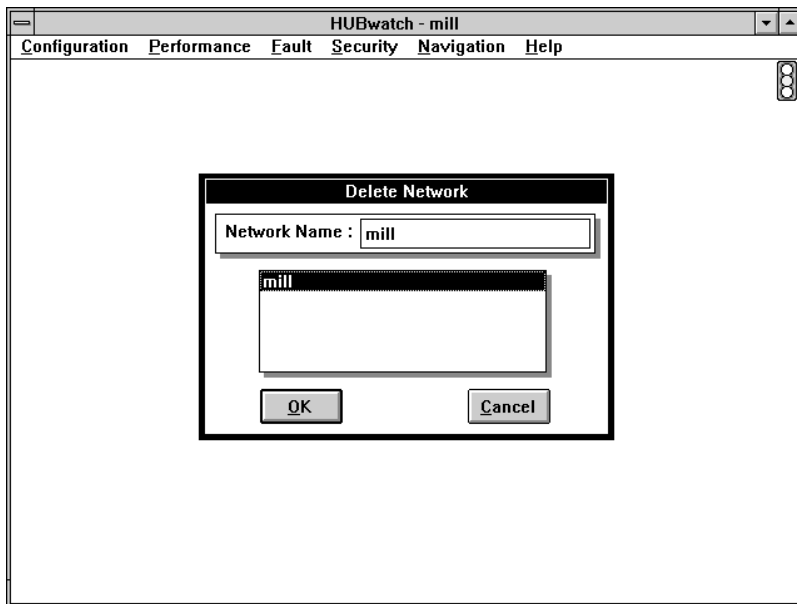
1. Choose the Network option from the Configuration menu.
The Network submenu appears.
2. Choose the Define option from the Network submenu.
The Delete Network Window appears. It lists all the network configuration files in that location.
3. Select the network you need to delete.
The selected network appears in the Network Name field (Figure 5–17).
4. Choose the OK button.
The Delete Network dialog box displays the following message:
Delete Network total
5. Choose the OK button.
The selected network is deleted.

Modifying a Network

HUBwatch enables you to change specific network attributes. Follow this procedure to modify a network:

1. Choose the Network option from the Configuration menu.
The Network submenu appears.
2. Choose the Modify option from the Network submenu.
The Modify Network window appears (Figure 5–18).

Figure 5-17 Delete Network Window



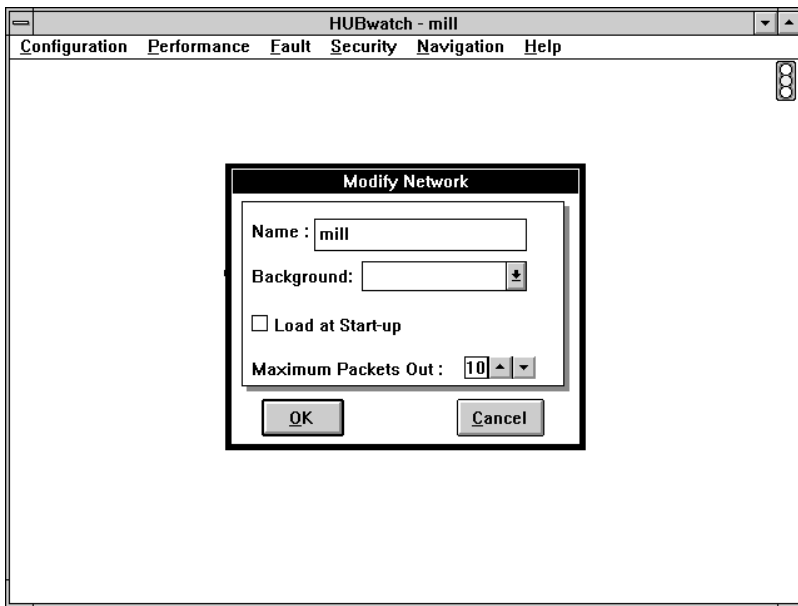
LJ-02682-SIX

3. Change any of the following attributes for the network, as needed.

Field	Description
Name	The name of the network.
Background	The type of background display you need for the network.
Load at Start-up	Choose this option, if you want the selected network to automatically load at startup.
Maximum Packets Out	Controls the rate at which HUBwatch polls network devices.

4. Choose the OK button.
The changes are entered.

Figure 5-18 Modify Network Window



LJ-02683-SIX

6

Managing Networks

Introduction

This chapter explains how to use the HUBwatch application to manage your network. The following topics are included in this chapter:

- Managing Network Performance
- Managing Network Faults
- Managing Network Security
- Accessing the MIB
- Accessing Network Reports

Managing Network Performance

Through the HUBwatch Performance menu, you can access network statistics, polling controls, and reports about network activity.

The Performance menu is available at all views, but depending on the view, some of the options may not be activated. When you choose the Performance menu, HUBwatch opens a pull-down menu with the following options:

- Statistics
- Graph
- Ping

Statistics Option

HUBwatch provides access to relevant statistical information for specified device experts.

To look at statistical information:

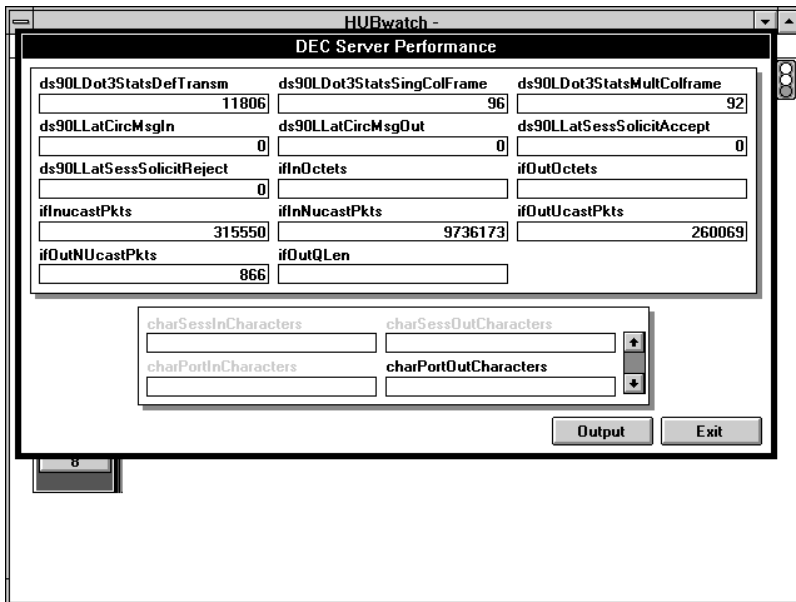
1. Set the view to the device for which you need the information.
2. Choose the Statistics option from the Performance pull-down menu.

HUBwatch displays the *module name* Performance window for the selected device. Figure 6–1 is an example of the Performance window for a DECserver module.

Note

The Statistics option is available only at the module view.

Figure 6–1 DECserver Performance Window



LJ-02684-SIX

Graph Option

Using the Graph option, you can display an interactive graph of the count overtime for a selected MIB node and object.

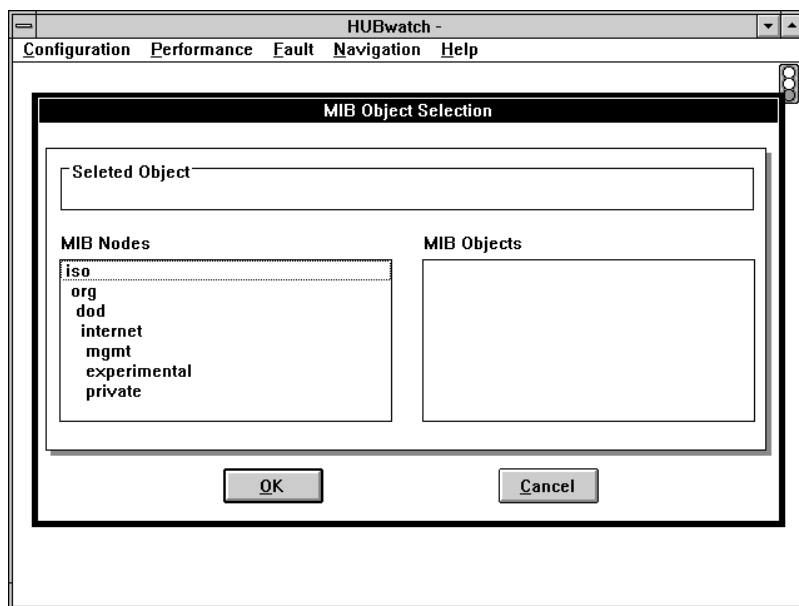
Note

The Graph option is available only at the hub and device views.

Follow this procedure to display the graph:

1. Set the view for the desired device or hub.
2. Choose the Graph option from the Performance menu.
HUBwatch displays the MIB Object Selection window for the specified view (Figure 6–2).

Figure 6–2 MIB Object Selection Window



LJ-02685-SIX

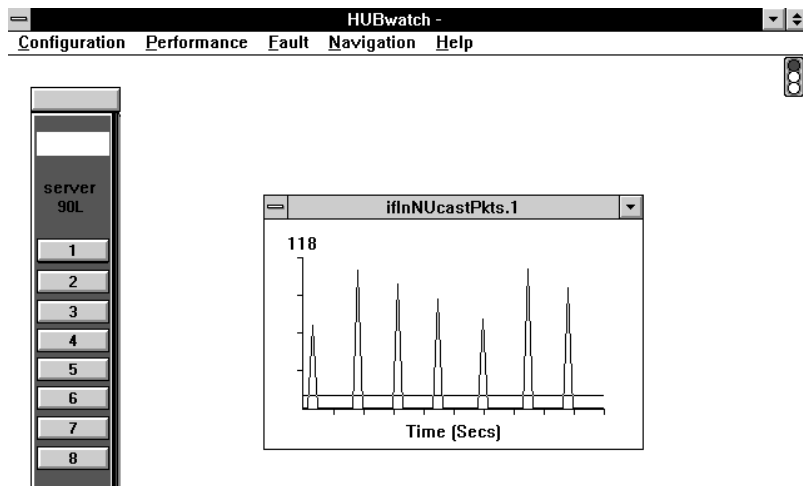
3. Double click MB1 on the MIB node for which you need to display the groups.

The groups for the selected MIB node appears.

4. Double click MB1 on the group for which you need to display the MIB object. The MIB objects for the selected group appear in the MIB Objects field.
5. Add instance information in the Selected Object field.
6. Select the MIB object for which you need the count. This must be a counter or gauge type object.
7. To display the graph, choose the OK button.

HUBwatch displays an interactive graph showing a count of the selected object overtime. Figure 6-3 shows a graph for a server object.

Figure 6-3 MIB Object Graph



LJ-02919-SIX

Note

By clicking the down arrow in the upper right corner of the graph window, you can change the graph window to an icon. This enables you to continue with your work, yet monitor the count in order to discern any problems.

You can create as many icons as you need in order to observe different counts at the same time.

Ping Option

The HUBwatch Ping utility tests the IP-level connectivity of selected devices and then displays the results. The device must support ICMP.

The default message size is 128 bytes, but you can set a packet size from 64 to 512 bytes.

Note

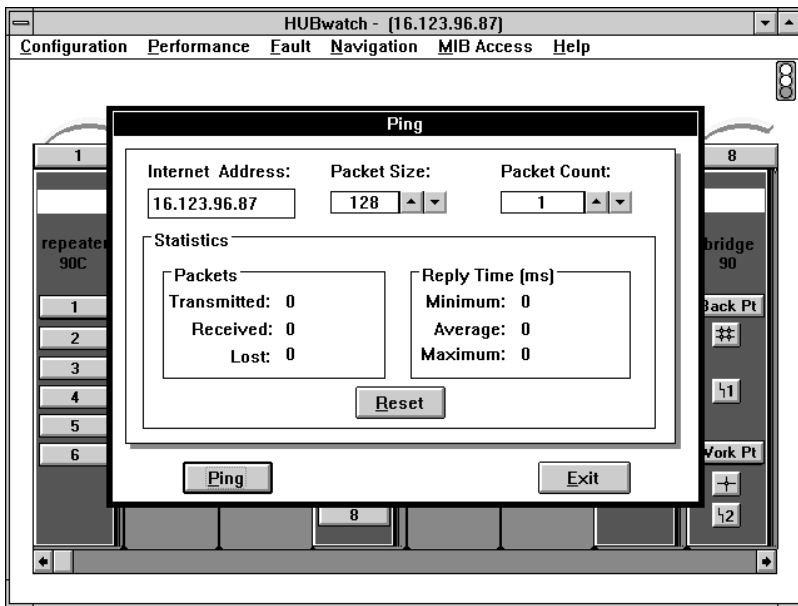
The Ping option is available only at the hub and device views.

Follow this procedure to use the Ping utility:

1. Set the view for the device you want to ping.
2. Choose the Ping option from the Performance pull-down menu.

An ICMP echo request is sent to the network element and the Ping window appears (Figure 6-4). The Ping window indicates the ping responses of the packets transmitted and received.

Figure 6-4 Ping Window



LJ-02686-SIX

Managing Network Faults

HUBwatch maintains a record of all network faults or alarms, whether major or minor. HUBwatch differentiates the type of alarm as follows:

Alarm	Description
Major	Conditions that may disrupt service for multiple users. For example: <ul style="list-style-type: none">• The management station is unable to communicate with a configured device.• A device restarts.• A device fails its self-test.• A device reports an internal problem.
Minor	Conditions that may disrupt service for a single user. For example, when a counter threshold is exceeded.

When an alarm is triggered, an event record is written to the alarm log with a timestamp and information about the alarm.

With HUBwatch, you can view the alarm reports, access the device responsible for the alarm, and set the alarm to sound or not.

You access alarm information through the Fault menu. The Fault menu is available at all views. When you pull down the Fault menu, HUBwatch displays the following options:

- Error Statistics
- Alarms
- Audible Alarms
- Report (Alarm Log)

Note

You can also set thresholds for each of these alarms. For more detailed information, refer to the Setting Thresholds section.

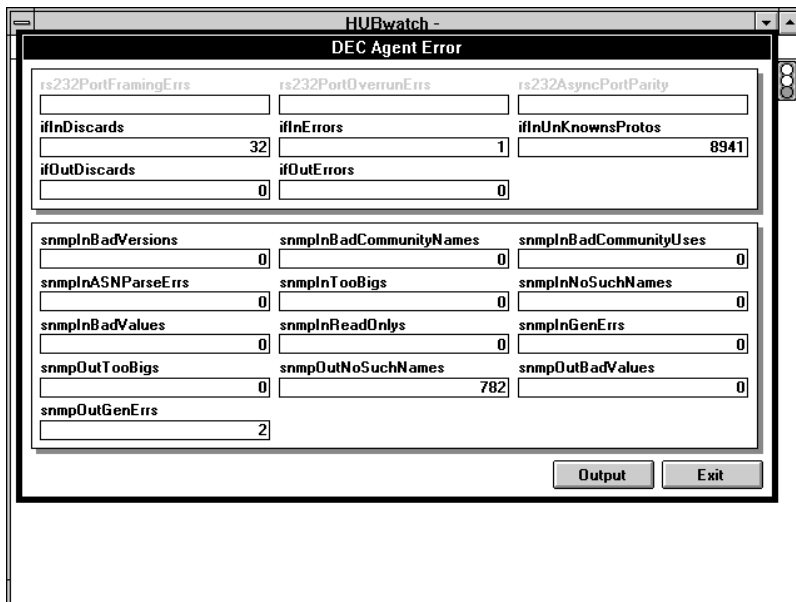
Viewing Error Statistics

HUBwatch provides statistical information about specific modules. Follow these procedures to view the error statistics:

1. Display the desired module for which you need to view the statistics.
2. Pull down the Fault menu and choose the Error Statistics option.

The *module name* Error window appears (Figure 6–5). You can either view the information or choose the Output button to print the report.

Figure 6–5 DECagent Error Window



LJ-02687-SIX

Note

The Error Statistics option is available only at the module view.

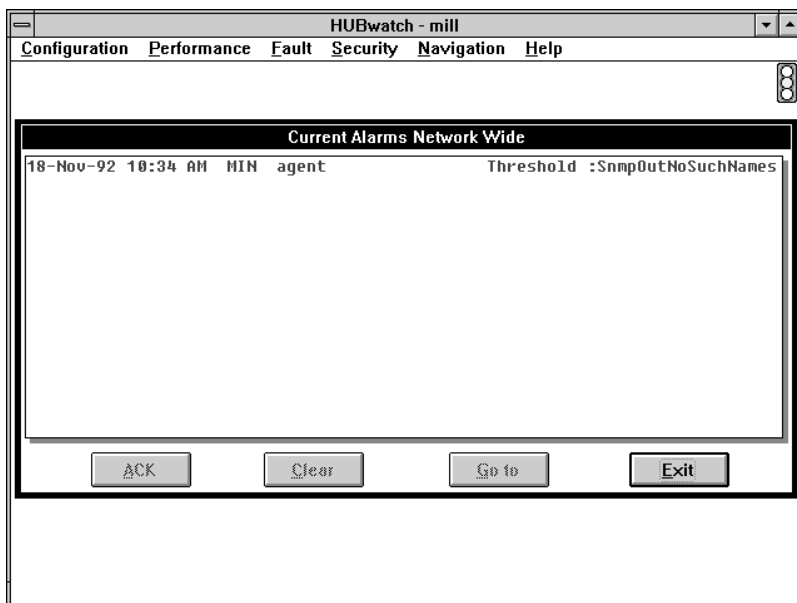
Viewing Alarms

HUBwatch provides an interactive list of all active alarms on the network. You can look at the list by doing one of the following:

- Choose the Alarms option from the Fault pull-down menu.
- Double click MB1 on the alarm icon.

HUBwatch displays the Current Alarms Network Wide window (Figure 6-6).

Figure 6-6 Current Alarms Network Wide Window



LJ-02688-SIX

The Current Alarms Network Wide window displays a record of all current alarms. The alarm list includes the following:

- Date and time of the alarm

- Whether it is a major (MAJ) or minor (MIN) alarm
- The name of the device, module or port associated with the alarm
- The HUBwatch application's assessment of the nature of the alarm

From this window you can scroll through the list, choose a specific alarm, and do any of the following for the selected alarm:

Choose this button . . .	Result
<u>A</u> CK	Acknowledges the alarm and turns the color of the particular list from red to black.
<u>C</u> lear	Dismisses the alarm from the list. After fixing the alarm, the alarm problem may be cleared automatically. If not, you may need to use the <u>C</u> lear button.
<u>G</u> o to	Brings you directly to the view of the device for which the alarm is indicated.

Note

The Alarm Log keeps a record of the alarm condition even after the alarm is cleared. Refer to the Viewing the Alarm Log section for more information.

Setting Audible Alarms

When an alarm is triggered, HUBwatch sends an audible signal (three short beeps) at one-minute intervals until the alarm is acknowledged or cleared. You can disable or enable audible alarms for major and minor alarms.

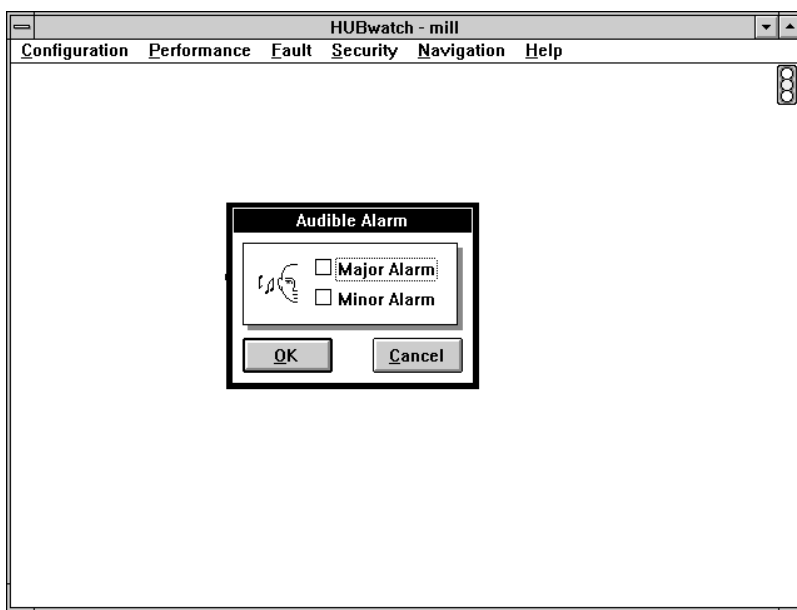
Follow this procedure to set an audible alarm:

1. Choose Audible Alarms option from the Fault menu.
HUBwatch opens the Audible Alarm dialog box (Figure 6–7).
2. Click on the appropriate box to make the major or minor alarms audible, or leave them blank if you do not want to hear the alarms.

Note

The Audible Alarms option is available at all views.

Figure 6–7 The Audible Alarm Dialog Box



LJ-02689-SIX

Setting Thresholds

You can set thresholds for various types of errors that are counted by a device. You can set each threshold to an arbitrary percentage of total frames (frames received without error) or turn off the threshold checking for that type of error. The default value for the thresholds is 5 percent. You can set the threshold value from 1 to 100 percent. HUBwatch enables you to look at and select threshold defaults so you can moderate a module's tolerance for certain kinds of errors.

Follow this procedure to set a threshold for module errors:

1. Choose the module for which you want to look at the threshold settings.
2. Choose the Set Threshold option from the Fault menu.

Depending on the module you select, one of the following happens:

If you select this module . . .	Then . . .
Agent	The Set Thresholds window appears for the agent (Figure 6–8).
Bridge	<p>A pull-down menu with the following options appears:</p> <ul style="list-style-type: none"> • Interface 1 • Interface 2
Server	<p>Select the appropriate option to display the Set Thresholds window.</p> <p>A pull-down menu with the following options appears:</p> <ul style="list-style-type: none"> • Interface • Slot
Repeater	<p>Select the appropriate option to display the Set Thresholds window.</p> <p>The Select Port dialog box appears. Select the port for which you need to set the thresholds and choose the OK button to display the Set Thresholds window (Figure 6–8).</p>

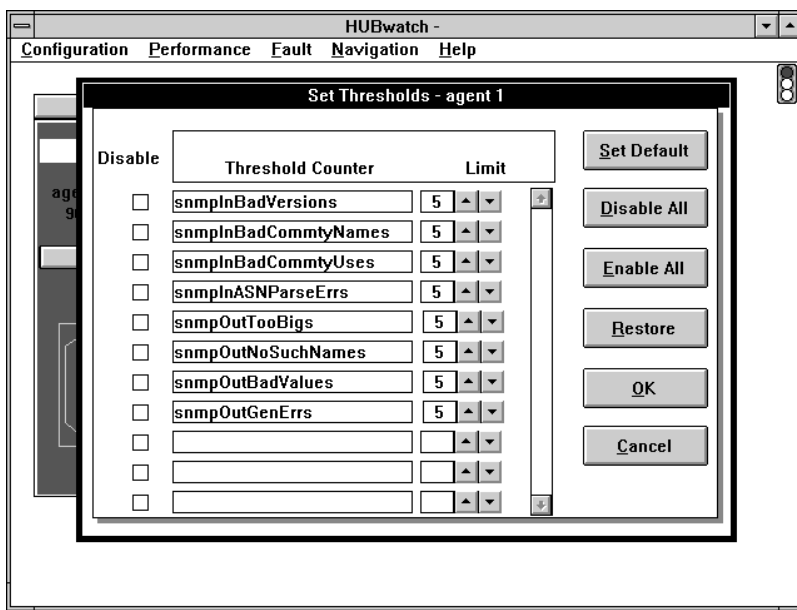
From the Set Thresholds window, you can do the following:

- Set thresholds to their default setting.
- Disable all the threshold counters.
- Enable all the threshold counters.
- Change specific threshold counters.
- Restore threshold settings to their original settings.

Note

The Set Thresholds option is available only at the module view.

Figure 6–8 Set Thresholds Window



LJ-02690-SIX

For more information about setting thresholds for the agent or other modules, refer to the following information:

To set thresholds for this module . . .	Refer to . . .
DECagent	The Setting Thresholds - DECagent section in this document
DECbridge	<i>HUBwatch for Windows DECbridge 90 Management</i>
DECserver	<i>HUBwatch for Windows DECserver 90 Management</i>
DECrepeater	<i>HUBwatch for Windows DECrepeater 90 Management</i>

Viewing the Alarm Log

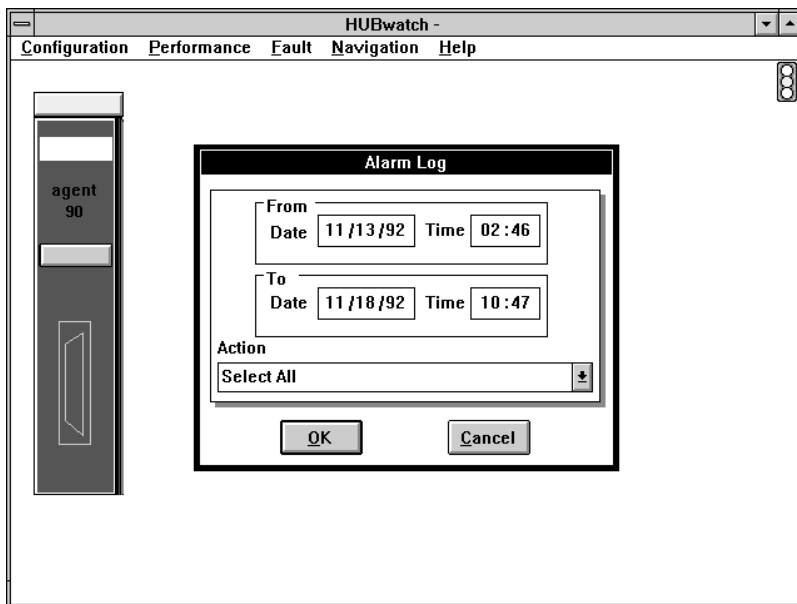
The Alarm Log is a record of all the alarms recorded by HUBwatch. This log includes the following information about alarms:

- Time
- Date
- Object Name
- Alarm Type

Follow this procedure to look at the alarms:

1. Choose the Report Alarm Log option from the Fault menu.
HUBwatch displays the Alarm Log window (Figure 6–9). It shows the date and time for which it will display the alarms.

Figure 6–9 Alarm Log Window



LJ-02691-SIX

2. To look at alarms within a different time period, change the times and dates.

3. To select a specific type of alarm, scroll through the various alarms and select the type of alarms you want to look at. You have the following choices:
 - All
 - Major
 - Minor
 - Unknown
4. When you are finished defining all the parameters, choose the OK button. The Alarm Log report appears (Figure 6–10).

Figure 6–10 Alarm Log Report

Date	Time	Sev	Object Name	Alarm Type
11/18/92	10:34 AM	Minor	agent	Threshold :SnmOutNoSuchNa
11/18/92	09:35 AM	Minor	agent	Threshold :SnmOutNoSuchNa
11/17/92	03:52 pM	Minor	agent	Threshold :SnmOutNoSuchNa
11/13/92	05:52 pM	Major	CLR Port 1	Repeater port autopartitio
11/13/92	05:52 pM	Major	CLR Port 2	Repeater port autopartitio
11/13/92	05:52 pM	Major	CLR Port 3	Repeater port autopartitio
11/13/92	05:52 pM	Major	CLR Port 4	Repeater port autopartitio
11/13/92	05:52 pM	Major	CLR Port 5	Repeater port autopartitio
11/13/92	05:52 pM	Major	CLR Port 6	Repeater port autopartitio
11/13/92	05:51 pM	Major	Port 6	Repeater port autopartitio
11/13/92	05:51 pM	Major	Port 5	Repeater port autopartitio
11/13/92	05:51 pM	Major	Port 4	Repeater port autopartitio
11/13/92	05:51 pM	Major	Port 3	Repeater port autopartitio
11/13/92	05:51 pM	Major	Port 2	Repeater port autopartitio
11/13/92	05:51 pM	Major	Port 1	Repeater port autopartitio
11/13/92	05:22 pM	Major	Port 6	Repeater port autopartitio
11/13/92	05:22 pM	Major	Port 5	Repeater port autopartitio
11/13/92	05:22 pM	Major	Port 4	Repeater port autopartitio
11/13/92	05:22 pM	Major	Port 3	Repeater port autopartitio
11/13/92	05:22 pM	Major	Port 2	Repeater port autopartitio
11/13/92	05:22 pM	Major	Port 1	Repeater port autopartitio
11/13/92	02:53 pM	Minor	agent	Threshold :SnmOutNoSuchNa
11/13/92	02:46 pM	Major		Repeater port autopartitio
11/13/92	02:46 pM	Major		Repeater port autopartitio

LJ-02692-SIX

Note

The Alarm Log Report option is available from all views.

Alarm Log Report Menu

The Alarm Log Report menu includes the following options:

Menu Option	Purpose
<u>F</u> ile	Includes the following options: <ul style="list-style-type: none">• <u>S</u>ave — Saves the current alarm log as a text or a delimited ASCII file.• <u>P</u>rint — Prints the alarm log on the defined printer.• <u>P</u>rinter <u>S</u>etup — Configures a printer for the HUBwatch application.• <u>E</u>xit <u>R</u>eport — Exits the Alarm Log Report Menu and returns you to the active view.
<u>F</u> ont	Provides access to the Font window where you can choose the font, font stem, font size, and font color for the report.
<u>R</u> eport	Includes the following options: <ul style="list-style-type: none">• <u>C</u>lear — Clears the entries in the alarm log.• <u>S</u>et <u>S</u>ize — Sets the maximum number of entries kept in the alarm log. When the maximum is exceeded, HUBwatch deletes the oldest items to make room for the newest items.
<u>H</u> elp	Provides access to online help for HUBwatch.

Managing Network Security

You can access network security reports and network password information through the Security menu.

The Security menu is available only at the Network view. When you choose the Security menu, a pull-down menu with the following options appears:

- Network Password
- Report

Setting Passwords

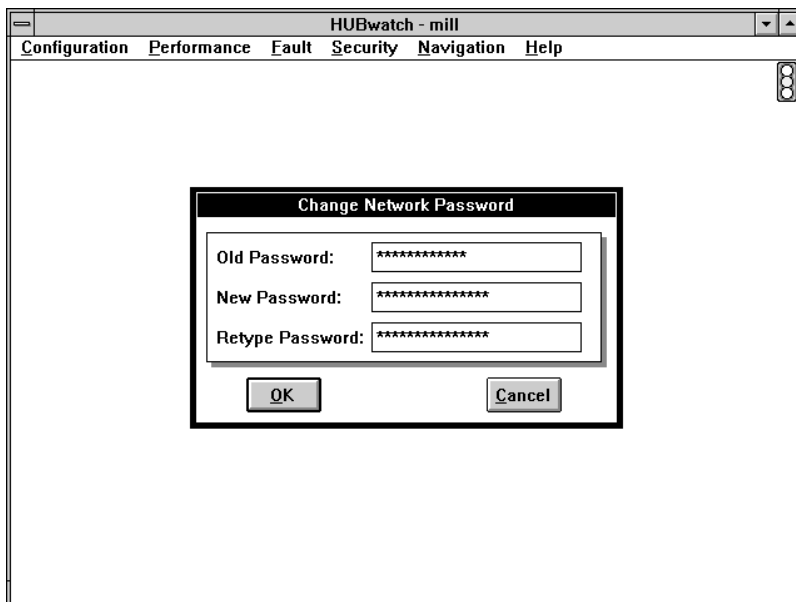
HUBwatch enables you to set or reset passwords for a network. Follow this procedure to set passwords:

1. Choose the Network Password option from the Security menu.
HUBwatch displays the Change Network Password window (Figure 6–11).
2. Enter the following information:

Field	Description
Old Password	Your present network password, if one exists.
New Password	The password that will replace the old password.
Retype Password	The new password (the same password you entered in the New Password field).

3. To set the password, choose the OK button.

Figure 6–11 Change Network Password Window



LJ-02693-SIX

Viewing Security Reports

The Security Log is a record of all security diagnostics or changes to the system between a certain date and time. It is available only from the network view.

For each log, the Security Log lists the following:

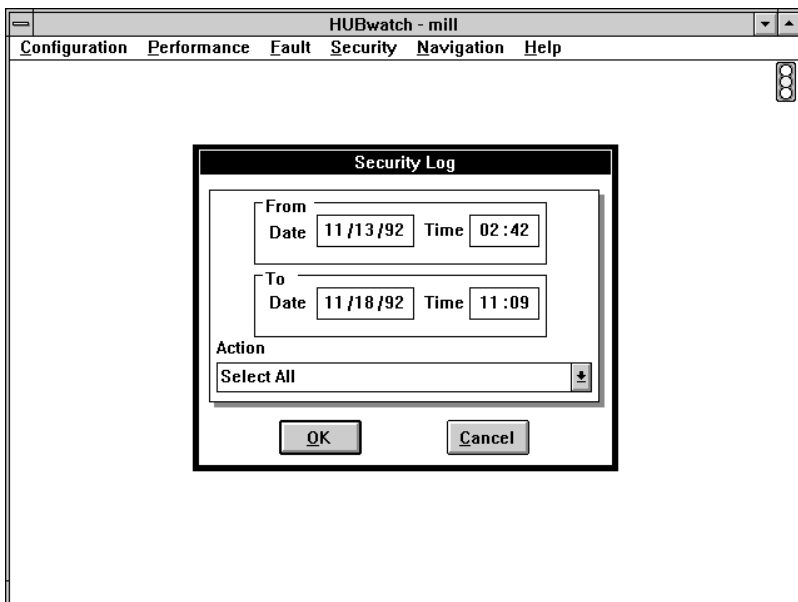
- Date
- Time
- Action

Follow this procedure to look at the Security Log:

1. Choose the Report option from the Security menu.
The Report submenu appears.
2. Choose the Security Log option from the Report menu.

HUBwatch displays the Security Log window (Figure 6–12).
It shows the date and time for which it will display the Security Log.

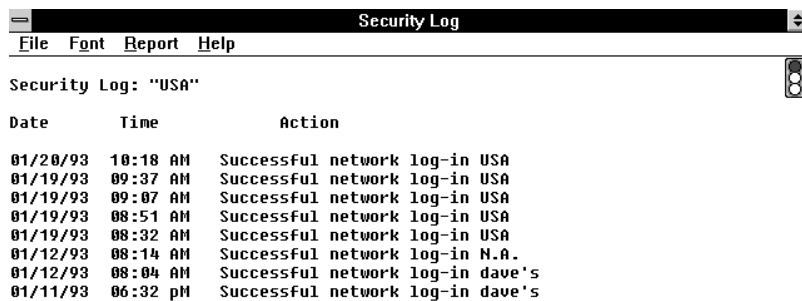
Figure 6–12 Security Log Window



LJ-02694-SIX

3. To look at log entries within a different time period, change the times and dates.
4. To choose a specific type of report, scroll through the Action list and select one of the following:
 - Backup network
 - Delete network
 - Diagnostic password changed
 - Network log-in failure
 - Network password changed
 - Invalid diagnostic password
 - Diagnostic log-in failure
 - Successful diagnostic log-in
 - Successful network log-in
5. After you define all the parameters, choose the OK button.
The Security Log report appears (Figure 6–13).

Figure 6–13 Security Log Report



Security Log: "USA"

Date	Time	Action
01/20/93	10:18 AM	Successful network log-in USA
01/19/93	09:37 AM	Successful network log-in USA
01/19/93	09:07 AM	Successful network log-in USA
01/19/93	08:51 AM	Successful network log-in USA
01/19/93	08:32 AM	Successful network log-in USA
01/12/93	08:14 AM	Successful network log-in N.A.
01/12/93	08:04 AM	Successful network log-in dave's
01/11/93	06:32 PM	Successful network log-in dave's

LJ-02695-SIX

Security Log Report Menu

The Security Log Report menu includes the following options:

Menu Option	Purpose
<u>F</u> ile	Includes the following options: <ul style="list-style-type: none">• <u>S</u>ave — Saves the current security log as a text or a delimited ASCII file.• <u>P</u>rint — Prints the security log on the defined printer.• <u>P</u>rinter Setup — Configures a printer for the HUBwatch application.• <u>E</u>xit Report — Exits the report and returns you to the active view.
<u>F</u> ont	Provides access to the Font window where you can choose the font, font stem, font size, and font color for the report.
<u>R</u> eport	Includes the following options: <ul style="list-style-type: none">• <u>C</u>lear — Clears the entries in the security log.• <u>S</u>et Size — Sets the maximum number of entries kept in the security log. When the maximum is exceeded, HUBwatch deletes the oldest items to make room for the newest items.
<u>H</u> elp	Provides access to online help for HUBwatch.

Accessing the Management Information Base

This section defines the menu options for accessing the management information base (MIB). Many of these options display a table of collected information for the various protocols. For detailed information about each of the MIB objects within the tables, refer to Appendix E.

Use the MIB Access menu to get and set SNMP management information from devices including bridges, routers, hosts, concentrators, workstations, and other generic SNMP-manageable devices.

Note

The MIB Access menu is available only from the device and hub view.

When you choose the MIB Access menu, HUBwatch opens a pull-down menu with the following options:

- MIBII
- Bridge
- Char-like
- RS232-like
- MIB ext

When you select one of the items, HUBwatch displays the major groups within each MIB.

Common Features of MIB Devices

MIB displays vary from object to object; however, all MIB displays share the following features:

- Each object is displayed in a separate text box.
- The name of each object appears above or to the left of the text box.
- Writable objects have a thick border on the left side of the text box.
- The cursor changes to the edit bar only within the text box of writable objects.
- Rows within a table are accessed using a scroll bar.
- Table index values are selected using a counter control, or by user input.

Managing MIB Information

To manage MIB information, use the Modify, Output, and Exit buttons. These buttons are defined in the following table.

Button	Purpose
<u>M</u> odify	Writes all changed variables to the device.
<u>O</u> utput	Sends displayed variables to the printer or a disk file.
<u>E</u> xit	Exits the MIB Access option.

Accessing the MIBII Groups

To access the MIBII group, pull down the MIB Access Menu and choose the MIBII option. The following options appear:

- System
- ITF
- AT
- IP
- ICMP
- TCP
- UDP
- EGP
- SNMP

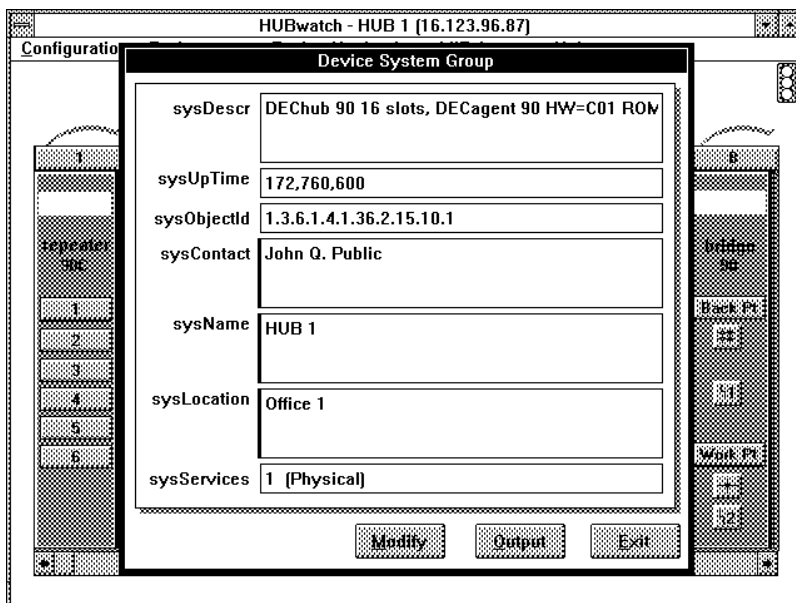
For detailed information about the MIB objects within the MIBII group tables, refer to Appendix E.

System Group

The System group must be implemented by all managed nodes. This group provides generic configuration information.

The System option provides access to this information through the Device System Group table (Figure 6-14).

Figure 6-14 Device System Group Table



LJ-02696-SIX

ITF Group

The Interface (ITF) group must be implemented by all managed nodes.

It displays generic information about the entities at the interface layer. It contains two top-level objects:

- The number of interface attachments on the node
- Information about the interfaces

The `ITF` option provides access to this information through the Interfaces Group table (Figure 6–15).

Figure 6–15 Interfaces Group Table

Interfaces Group			
ifNumber	ifDescr		
4	DECserver 90L+ V2.0 BL4.7		
ifIndex	ifType		
1	6		
ifMtu	ifLastChange	ifInErrors	ifOutDiscards
1,500	0	0	0
ifSpeed	ifInOctets	ifInUnknownProtos	ifOutErrors
10,000,000	209,705,238	17,776	0
ifPhysAddress	ifInUcastPkts	ifOutOctets	ifOutQLen
08-00-2B-2F-8B-F6	1,521,236	66,067,040	16
ifAdminStatus	ifInNUcastPkts	ifOutUcastPkts	ifSpecific
1 (up)	47,573,971	1,253,222	0.0.0
ifOperStatus	ifInDiscards	ifOutNUcastPkts	
1 (up)	92	4,206	

Modify Output Exit

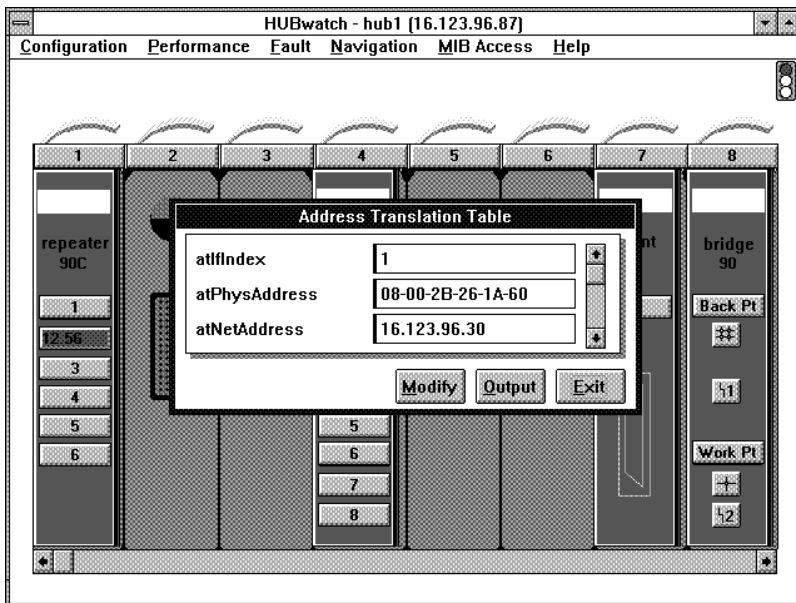
LJ-02697-SIX

AT Group

The Address Translation (AT) group must be implemented by all managed nodes. It contains address resolution information. A single table is used to map IP addresses into media-specific addresses.

The AT option provides access to this information through the Address Translation Table (Figure 6-16).

Figure 6-16 Address Translation Table



LJ-02698-SIX

IP Group

The IP group must be implemented by all managed nodes.

The IP option provides access to a submenu that contains four options. These options represent the four IP group tables. They include the following:

- Group Table
- Address Table
- Routing Table
- Net-To-Media Table

Follow this procedure to access the IP tables:

1. Pull down the MIB Access menu and choose MIBII.
The MIBII submenu appears.
2. Choose the IP option.
A submenu that lists the IP Group tables appears.

Group Table

The Group option provides access to the IP Group table (Figure 6–17).

Figure 6–17 IP Group Table

The screenshot shows a window titled "HUBwatch - HUB 1 (16.123.96.87)" with a menu bar containing "Configuration", "Performance", "Fault", "Navigation", "MIB Access", and "Help". The main content area displays the "IP Group" configuration table. The table has four columns and five rows of data. Below the table are three buttons: "Modify", "Output", and "Exit".

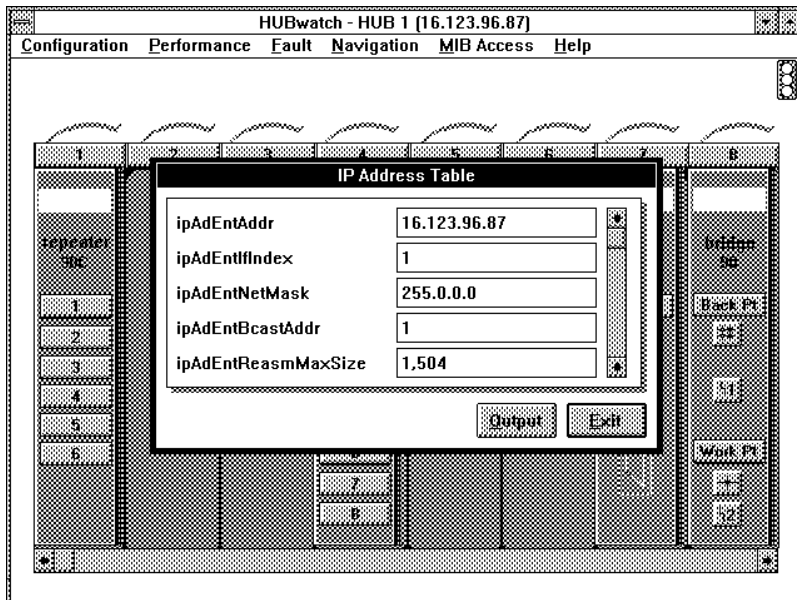
ipForwarding	ipForwDatagrams	ipOutDiscards	ipReasmFails
2 [not-forwarding]	0	0	0
ipDefaultTTL	ipInUnknownProtos	ipOutNoRoutes	ipFragOKs
30	0	0	0
ipInReceives	ipInDiscards	ipReasmTimeout	ipFragFails
13,920	0	60	0
ipInHdrErrors	ipInDelivers	ipReasmReqds	ipFragCreates
0	13,920	0	0
ipInAddrErrors	ipOutRequests	ipReasmOKs	
0	13,916	0	

LJ-02699-SIX

Address Table

The Address Table option provides access to the IP Address Table (Figure 6–18). This table displays the IP addresses associated with the managed table.

Figure 6–18 IP Address Table

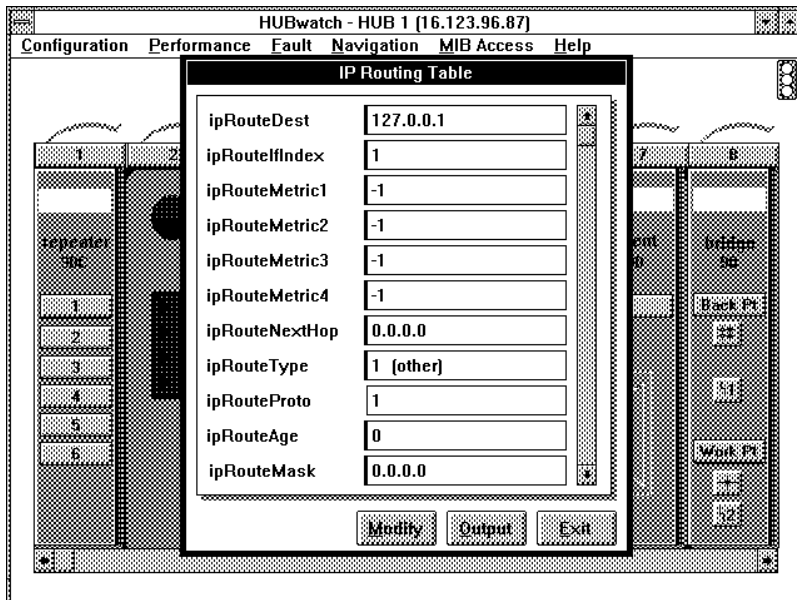


LJ-02700-SIX

Routing Table

The Routing Table option provides access to the IP Routing Table (Figure 6–19). This table displays the IP routes associated with the managed node.

Figure 6–19 IP Routing Table

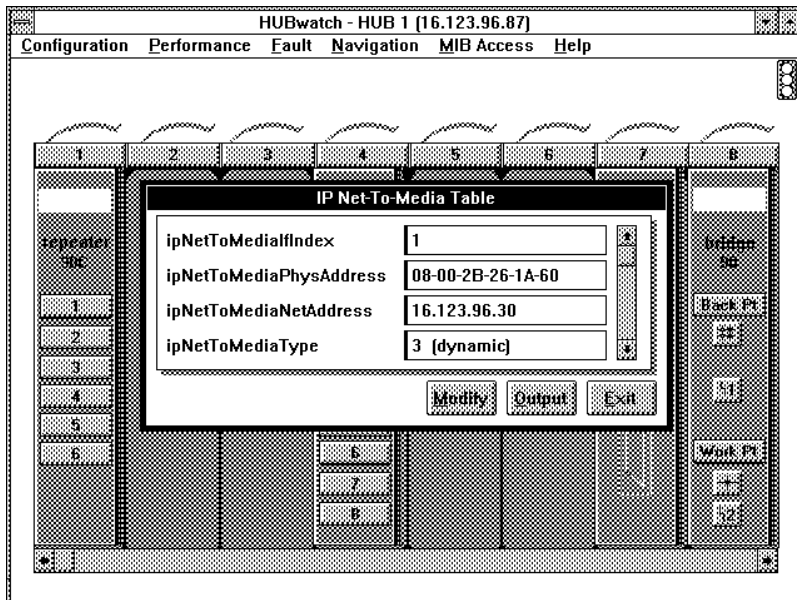


LJ-02701-SIX

Net-To-Media Table

The Net-To-Media Table option provides access to the IP Net-To-Media Table (Figure 6–20). This table displays IP and media-specific addresses.

Figure 6–20 IP Net-To-Media Table



LJ-02702-SIX

ICMP Group

The Internet Control Message Protocol (ICMP) group must be implemented by all managed nodes. It provides low-level statistical feedback about how the internet layer is operating. The `ICMP` option provides access to this information through the ICMP Group table (Figure 6–21). The group table consists of 26 counters. For each ICMP message type, two counters exist:

- One counter counts the number of times a message type is generated by the local IP entity.
- The other counter counts the number of times a message is received by the local IP entity.

Figure 6–21 ICMP Group Table

ICMP Group			
icmpInMsgs	4	icmpOutMsgs	4
icmpInErrors	0	icmpOutErrors	0
icmpInDestUnreachs	0	icmpOutDestUnreachs	0
icmpInTimeExcds	0	icmpOutTimeExcds	0
icmpInParmProbs	0	icmpOutParmProbs	0
icmpInSrcQuenchs	0	icmpOutSrcQuenchs	0
icmpInRedirects	0	icmpOutRedirects	0
icmpInEchos	4	icmpOutEchos	0
icmpInEchoReps	0	icmpOutEchoReps	4
icmpInTimestamps	0	icmpOutTimestamps	0
icmpInTimestampReps	0	icmpOutTimestampReps	0
icmpInAddrMasks	0	icmpOutAddrMasks	0
icmpInAddrMaskReps	0	icmpOutAddrMaskReps	0

LJ-02703-SIX

TCP Group

The Transmission Control Protocol (TCP) group must be implemented by all managed nodes which implement TCP. It contains several fields and a table that displays application entities using TCP.

The TCP option provides access to this information through the TCP Group table (Figure 6–22).

Figure 6–22 TCP Group Table

The screenshot shows a configuration window titled "HUBwatch - HUB 1 (16.123.96.87)". The main area is titled "TCP Group" and contains several input fields for configuration parameters:

tcpRtoAlgorithm	tcpPassiveOpens	tcpCurrEstab
tcpRtoMin	tcpInErrs	tcpInSegs
tcpRtoMax	tcpOutRsts	tcpOutSegs
tcpMaxConn	tcpAttemptFails	tcpRetransSegs
tcpActiveOpens	tcpEstabResets	

Below these fields is a section titled "TCP Connections Table" with the following fields:

tcpConnState	tcpConnLocalPort	tcpConnRemPort
tcpConnLocalAddress	tcpConnRemAddress	

At the bottom of the window are "Output" and "Exit" buttons.

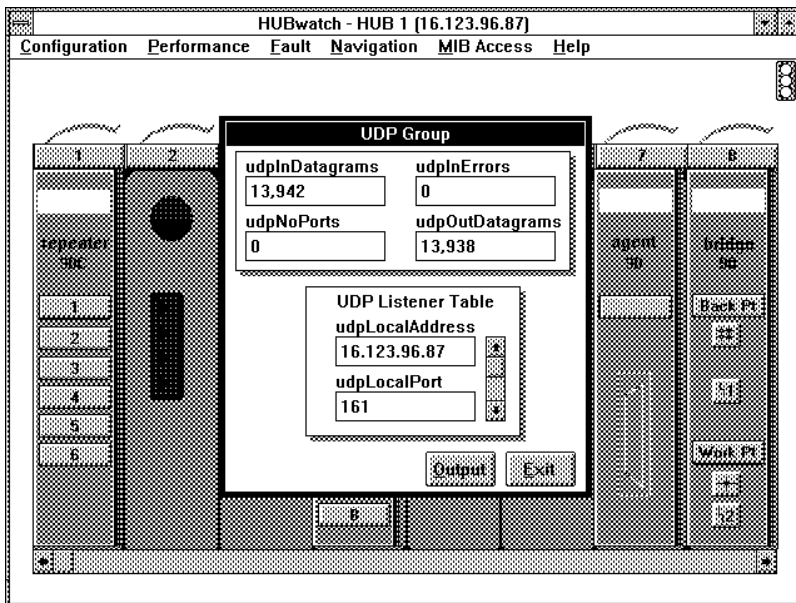
LJ-02704-SIX

UDP Group

The User Datagram Protocol (UDP) group must be implemented by all managed nodes that implement UDP. It contains four counters and a table to keep track of the application entities which are using UDP.

The UDP option provides access to this information through the UDP Group table (Figure 6-23).

Figure 6-23 UDP Group Table



LJ-02705-SIX

EGP Group

The Exterior Gateway Protocol (EGP) group is mandatory for those managed nodes which implement the Exterior Gateway Protocol. The EGP exchanges reachability information between autonomous systems and the core.

The EGP option provides access to this information through the EGP Group table (Figure 6–24). For detailed information about each of the MIB object names, refer to Appendix E.

Figure 6–24 EGP Group Table

The screenshot shows a configuration window titled "EGP Group" with two main sections: "EGP Group" and "EGP Neighbor Table".

EGP Group		
egpInMsgs	egpOutMsgs	egpAs
<input type="text"/>	<input type="text"/>	<input type="text"/>
egpInErrors	egpOutErrors	
<input type="text"/>	<input type="text"/>	

EGP Neighbor Table		
egpNeighState	egpNeighOutMsgs	egpNeighStateDowns
<input type="text"/>	<input type="text"/>	<input type="text"/>
egpNeighAddr	egpNeighOutErrs	egpNeighIntervalHello
<input type="text"/>	<input type="text"/>	<input type="text"/>
egpNeighAs	egpNeighInErrMsgs	egpNeighIntervalPoll
<input type="text"/>	<input type="text"/>	<input type="text"/>
egpNeighInMsgs	egpNeighOutErrMsgs	egpNeighMode
<input type="text"/>	<input type="text"/>	<input type="text"/>
egpNeighInErrs	egpNeighStateUps	egpNeighEventTrigger
<input type="text"/>	<input type="text"/>	<input type="text"/>

Buttons: Output, Exit

LJ-02706-SIX

SNMP Group

A set of managed objects are defined for Simple Network Management Protocol (SNMP) application entities. The SNMP option provides access to this information through the SNMP Group table (Figure 6–25).

Figure 6–25 SNMP Group Table

SNMP Group			
snmplnBadCommunityNames	snmplnBadCommunityUses	snmplnASNParseErrs	
3	0	0	
snmplnBadVersions	snmplnBadTypes	snmplnGetResponses	snmpOutGetResponses
0		0	13,946
snmplnPkts	snmpOutPkts	snmplnNoSuchNames	snmpOutNoSuchNames
13,949	13,945	0	35
snmplnReadOnlys	snmpOutReadOnlys	snmplnTooBig	snmpOutTooBig
0		0	0
snmplnBadValues	snmpOutBadValues	snmplnTraps	snmpOutTraps
0	2	0	0
snmplnGetNexts	snmpOutGetNexts	snmplnGenErrs	snmpOutGenErrs
487	0	0	0
snmplnGetRequests	snmpOutGetRequests	snmplnTotalSetVars	snmplnTotalReqVars
13,412	0	10	26,914
snmplnSetRequests	snmpOutSetRequests	snmpEnableAuthTraps	
10	0	0	

Modify Output Exit

LJ-02707-SIX

Accessing Bridge Groups

To access the Bridge groups, pull down the MIB Access menu and choose the Bridge option. The following bridge groups appear:

- Dot1dBase
- Dot1dStp
- Dot1dSr
- Dot1dTp
- Dot1dStatic

For detailed information about the MIB objects within the Bridge Group tables, refer to Appendix E.

Dot1dBase Group

Choose the Dot1dBase option to display the Bridge - dot1dBase Group table (Figure 6–26).

Figure 6–26 Bridge - dot1dBase Group Table

The screenshot shows a configuration window titled "HUBwatch - HUB 1 (16.123.96.87)" with a menu bar containing "Configuration", "Performance", "Fault", "Navigation", "MIB Access", and "Help". The main content area is titled "Bridge - dot1dBase Group" and contains two sections of configuration fields:

dot1dBase Group	
dot1dBaseBridgeAddress	08-00-2B-30-D2-2B
dot1dBaseNumPorts	2
dot1dBaseType	2 [transparent-only]

dot1dBase Group	
dot1dBasePort	1
dot1dBaseIndex	3
dot1dBasePortCircuit	0.0
dot1dBasePortDelayExcdDiscards	0
dot1dBaseMtuExcdDiscards	31

At the bottom of the configuration area are two buttons: "Output" and "Exit".

LJ-02708-SIX

Dot1dStp Group

Choose the Dot1dStp option to display the Bridge - dot1dStp Group table (Figure 6–27).

Figure 6–27 Bridge - dot1dStp Group Table

The screenshot shows a configuration window titled "HUBwatch - HUB 1 (16.123.96.87)". The main menu includes "Configuration", "Performance", "Fault", "Navigation", "MIB Access", and "Help". The current view is "Bridge - dot1dStp Group".

dot1dStpProtocolSpecification	dot1dStpPriority	dot1dStpTimeSinceTopChange
3 (ieee8021d)	0	0
dot1dStpTopChanges	dot1dStpDesignatedRoot	dot1dStpRootCost
0	0.5.8.0.43.38.81.241	10
dot1dStpRootPort	dot1dStpMaxAge	dot1dStpHelloTime
1		1
dot1dStpHoldTime	dot1dStpForwardDelay	dot1dStpBridgeMaxAge
1.500	15	15
dot1dStpBridgeHelloTime	dot1dStpBridgeForwardDelay	
1	15	

dot1dStpPort	dot1dStpPortPriority	dot1dStpPortState
dot1dStpPortEnable	dot1dStpPortPathCost	dot1dStpPortDesignatedRoot
dot1dStpPortDesignatedCost	dot1dStpPortDesignatedBridge	dot1dStpPortDesignatedPort
dot1dStpPortForwardTransitions		

Buttons: Modify, Output, Exit

LJ-02709-SIX

Dot1dSr Group

Choose the Dot1dSr option to display the Bridge - dot1dSr Group table (Figure 6–28).

Figure 6–28 Bridge - dot1dSr Group Table

The screenshot shows a web-based configuration interface for a network device. The main window is titled "HUBwatch - HUB 1 (16.123.96.87)" and has a menu bar with "Configuration", "Performance", "Fault", "Navigation", "MIB Access", and "Help". The central area displays a configuration page for "Bridge - dot1dSr Group".

The configuration page is titled "Bridge - dot1dSr Group" and contains a table labeled "dot1dSrPort Table". The table has the following fields:

dot1dSrPort	dot1dSrPortHopCount	dot1dSrPortLocalSegment
<input type="text"/>	<input type="text"/>	<input type="text"/>
dot1dSrPortBridgeNum	dot1dSrPortTargetSegment	dot1dSrPortLargestFrame
<input type="text"/>	<input type="text"/>	<input type="text"/>
dot1dSrPortSTESpanMode	dot1dSrPortSpecInFrame	dot1dSrPortSpecOutFrame
<input type="text"/>	<input type="text"/>	<input type="text"/>
dot1dSrPortApInFrame	dot1dSrPortApOutFrame	dot1dSrPortSteInFrame
<input type="text"/>	<input type="text"/>	<input type="text"/>
dot1dSrPortSteOutFrame	dot1dSrPortSegMismatch	dot1dSrPortDupSegDisc
<input type="text"/>	<input type="text"/>	<input type="text"/>
dot1dSrPortHopCountExcDisc		
<input type="text"/>		

At the bottom of the configuration page, there are three buttons: "Modify", "Output", and "Exit".

LJ-02710-SIX

Dot1dTp Group

Choose the Dot1dTp option to display the Bridge - dot1dTp Group table (Figure 6–29).

Figure 6–29 Bridge - dot1dTp Group Table

The screenshot shows a configuration window titled "Bridge - dot1dTp Group" with three sections of settings:

- Dot1dTp Group**
 - dot1TpLearnedEntryDiscard: 0
 - dot1TpAgingTime: 910
- dot1TpFdbTable**
 - dot1TpFdbAddress: 08-00-2B-2D-4E-6C
 - dot1TpFdbPort: 2
 - dot1TpFdbStatus: 3 (learned)
- dot1dTpPortTable**
 - dot1TpPort: 1
 - dot1TpPortMaxInfo: 1,500
 - dot1TpPortInFrames: 1,556,684,224
 - dot1TpPortOutFrames: 51,098
 - dot1TpPortInDiscards: 601,962,875

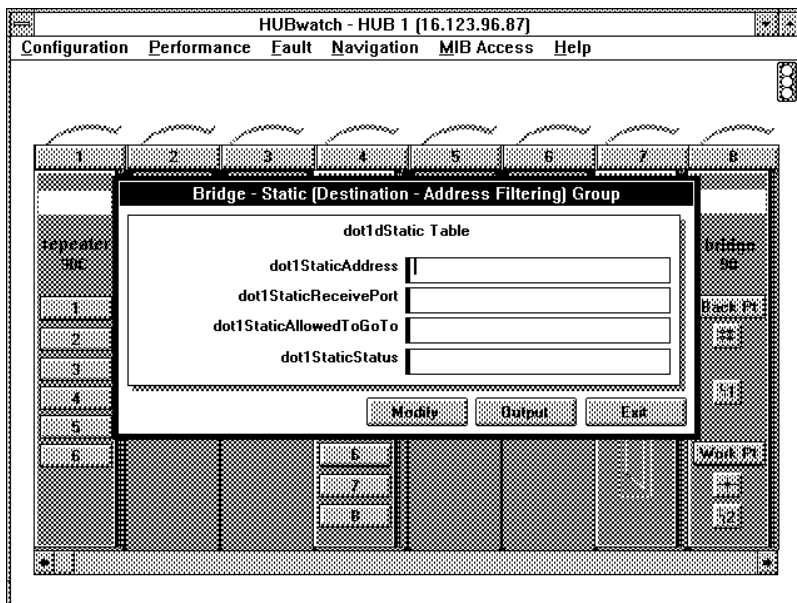
At the bottom of the window are buttons for "Modify", "Output", and "Exit".

LJ-02711-SIX

Dot1dStatic Group

Choose the Dot1dStatic option to display the Bridge - Static (Destination - Address Filtering) Group table (Figure 6-30).

Figure 6-30 Bridge - Static (Destination - Address Filtering) Group Table



LJ-02712-SIX

Accessing Char-like Groups

To access the Char-like groups, pull down the MIB Access menu and choose the Char-like option. The following char-like groups appear:

- General
- Session

For detailed information about the MIB objects within the Char-like Group tables, refer to Appendix E.

General Group (char-like)

Choose the General option to display the Character Port Table (Figure 6–31).

Figure 6–31 Character Port Table

Character Port Table					
charNumber	<input type="text"/>	charPortName	<input type="text" value="PORT_1"/>		
charPortIndex	<input type="text" value="401"/>	charPortType	<input type="text" value="1 (physical)"/>		
charPortHardware	<input type="text" value="1.3.6.1.2.1.10.33"/>	charPortReset	<input type="text" value="1 (ready)"/>	charPortAdminStatus	<input type="text" value="1 (enabled)"/>
charPortOperStatus	<input type="text" value="2 (down)"/>	charPortLastChange	<input type="text" value="1,728,274"/>	charPortInFlowType	<input type="text" value="2 (xonXoff)"/>
charPortOutFlowType	<input type="text" value="2 (xonXoff)"/>	charPortInFlowState	<input type="text" value="4 (go)"/>	charPortOutFlowState	<input type="text" value="4"/>
charPortInCharacters	<input type="text"/>	charPortOutCharacters	<input type="text"/>	charPortAdminOrigin	<input type="text" value="3 (local)"/>
charPortSessionMaximum	<input type="text" value="4"/>	charPortSessionNumber	<input type="text" value="0"/>	charPortSessionIndex	<input type="text"/>

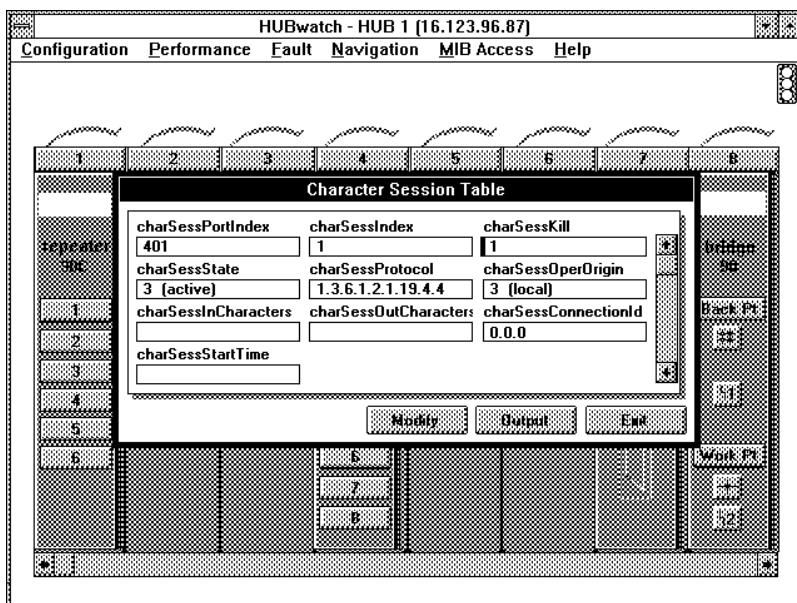
Modify Output Exit

LJ-02714-SIX

Session Group

Choose the Session option to display the Character Session Table (Figure 6–32).

Figure 6–32 Character Session Table



LJ-02713-SIX

Accessing RS232-like Groups

To access the RS232-like groups, pull down the MIB Access menu and choose the RS232-like option. The following RS232-like groups appear:

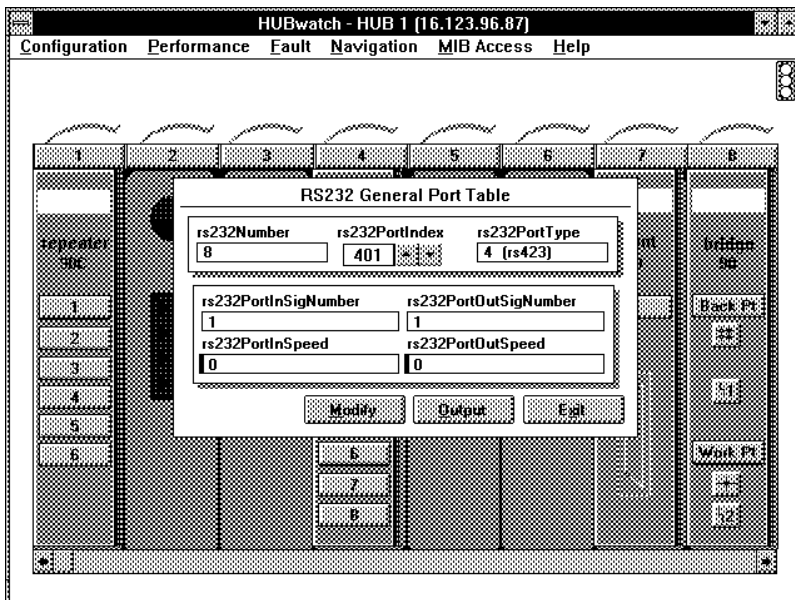
- General
- Async
- Sync

For detailed information about the MIB objects within the RS232-like Group tables, refer to Appendix E.

General Groups (RS232-like)

Choose the General option to display the RS232 General Port Table (Figure 6–33).

Figure 6–33 RS232 General Port Table

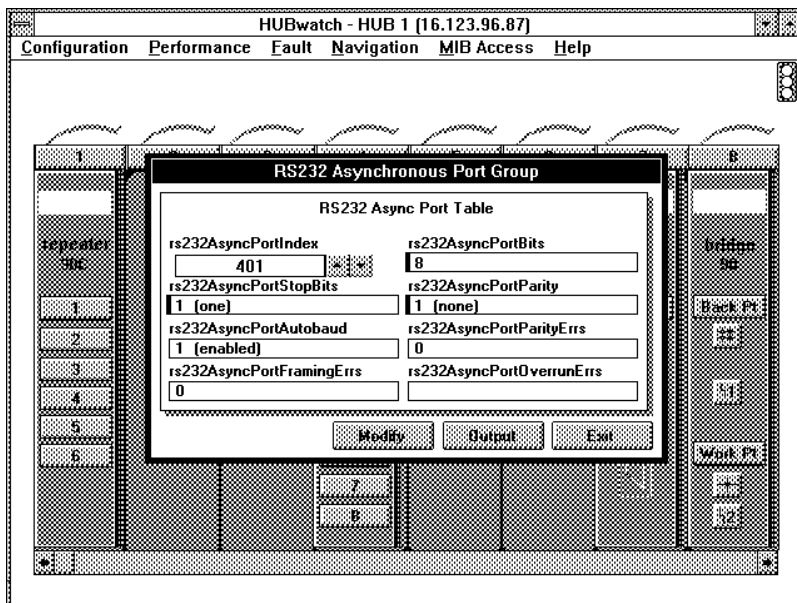


LJ-02715-SIX

Async Group

Choose the Async option to display the RS232 Asynchronous Port Group Table (Figure 6–34).

Figure 6–34 RS232 Asynchronous Port Group Table



LJ-02716-SIX

Sync Group

Choose the Sync option to display the RS232 Synchronous Port Group Table (Figure 6–35).

Figure 6–35 RS232 Synchronous Port Group Table

The screenshot shows a configuration window titled "HUBwatch - HUB 1 [16.123.96.87]". The main area is titled "RS232 Synchronous Port Group" and contains a "Synchronous Port Table" with the following fields:

rs232SyncPortIndex	rs232SyncPortClockSource
rs232SyncPortFrameCheckErrs	rs232SyncPortTransUnderErrs
rs232SyncPortRecvTransOverErrs	rs232SyncPortInteruptedFrames
rs232SyncPortAbortedFrames	

Below the main table are two sub-tables:

Input Signal Table	Output Signal Table
rs232InSigPortIndex: 401	rs232OutSigPortIndex: 401
rs232InSigName: 3 (dsr)	rs232OutSigName: 4 (dtr)
rs232InSigState: 2 (on)	rs232OutSigState: 3 (off)
rs232InSigChanges: 0	rs232OutSigChanges: 0

At the bottom of the window are three buttons: "Modify", "Output", and "Exit".

LJ-02717-SIX

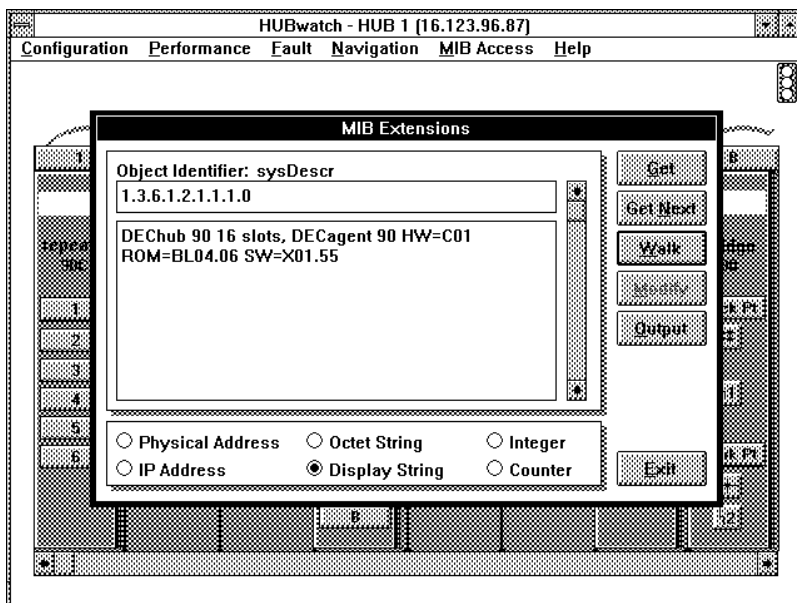
Viewing the MIB Variable

The MIB Ext option allows access to any management variable defined in accordance with the Structure of Management Information (SMI). Depending on the SNMP agent on the managed device, these variables may be read/write (modifiable), read-only, write-only, or non-accessible.

Each variable represents an end point on the MIB tree and is defined in a MIB document. Standard MIB variables are defined in RFC (Request for Comment) 1213, known as MIBII. For private extensions (also called enterprise extensions), the variables and their meaning are usually defined in a document created by the private organization. The DEChub MIB specification is in Appendix E.

To access management variables, pull down the MIB Access menu and choose the MIB Ext option. The MIB Extensions window appears (Figure 6-36).

Figure 6-36 MIB Extensions Window



LJ-02718-SIX

Through the MIB Extensions window you can use the following buttons to look at or change the MIB variables:

Button	Action
<u>G</u> et	Displays the Object Identifier listed.
Get <u>N</u> ext	Finds and displays the Object Identifier listed after the one in the Object Identifier field.
<u>W</u> alk or <u>S</u> top	When activated, the <u>W</u> alk button changes to a <u>S</u> top button. Until the <u>S</u> top button is clicked, HUBwatch rapidly moves forward through the MIB starting from the object displayed in the Object Identifier field. The system displays each subsequent object in the MIB and defines it by one of the following: <ul style="list-style-type: none">• Physical address• IP address• Octet string• Display string• Integer• Counter
<u>M</u> odify	<i>Not available for HUBwatch for Windows, Version 1.0.</i>
<u>O</u> utput	Enables you to print the MIB Extensions window on a printer, or generate output in a text or ASCII file.
<u>E</u> xit	Cancels the MIB Extensions window.

Viewing a Specific MIB Variable

Follow this procedure to view a given variable:

1. Enter the position of the variable on the MIB tree in the Object Identifier field.
2. Choose the Get button.
HUBwatch displays the value of the variable.

Walking the MIB

Follow this procedure to walk the MIB:

1. Enter the start position in the MIB tree in the Object Identifier field.
2. Choose the Walk button.

The network management station automatically walks through the variables, reading each value until one of the following occurs:

- The last variable is read.
- The Stop button is chosen.
- The system runs out of storage space for the MIB variables.

Accessing Network Reports

General network reports are accessed through the Configuration menu at the network, site, and hub views. Reports under the Configuration menu at the Module view are module specific. For module-specific reports, refer to the following:

- The Configuring Your Network Management Station for HUBwatch section in Chapter 2 of this document
- *HUBwatch for Windows DECbridge 90 Management*
- *HUBwatch for Windows DECserver 90 Management*
- *HUBwatch for Windows DECrepeater 90 Management*

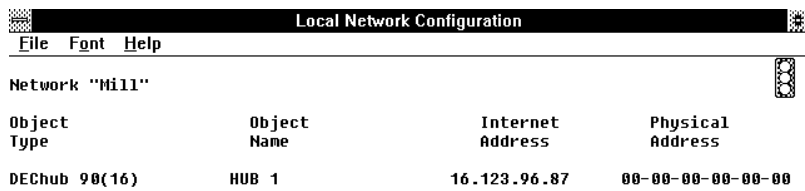
To access the network reports, choose the Report option in the Configuration pull-down menu. HUBwatch opens a submenu with the following report options:

- Local Configuration
- Brief Configuration
- Full Configuration
- Change Log

Local Configuration Report

The Local Configuration report lists the sites and devices in the current view (Figure 6-37). This option is available at the network, site, and hub view.

Figure 6-37 Local Configuration Report



The screenshot shows a terminal window titled "Local Network Configuration". At the top, there is a menu bar with "File", "Font", and "Help". Below the menu bar, the text "Network 'Mill'" is displayed. A table follows, listing network objects. The table has four columns: "Object Type", "Object Name", "Internet Address", and "Physical Address". One row is visible, showing "DEChub 98(16)" as the object type, "HUB 1" as the object name, "16.123.96.87" as the internet address, and "00-00-00-00-00-00" as the physical address. To the right of the table, there is a vertical status indicator consisting of three small circles.

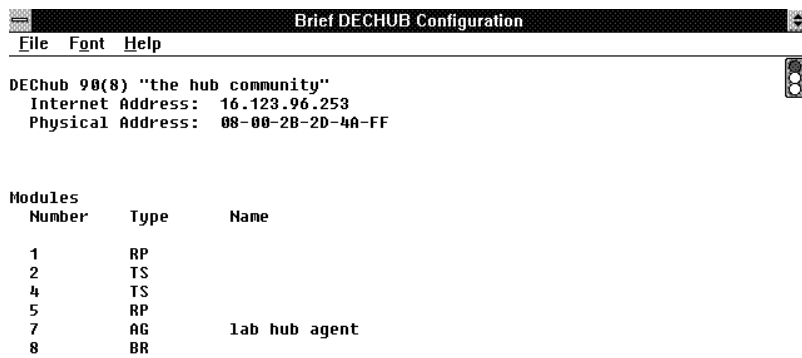
Object Type	Object Name	Internet Address	Physical Address
DEChub 98(16)	HUB 1	16.123.96.87	00-00-00-00-00-00

LJ-02719-SIX

Brief Configuration Report

The Brief Configuration Report lists the device and modules (Figure 6–38). This option is available at the device and hub view.

Figure 6–38 Brief Configuration Report



```
DECHub 90(8) "the hub community"
Internet Address: 16.123.96.253
Physical Address: 08-00-2B-2D-4A-FF

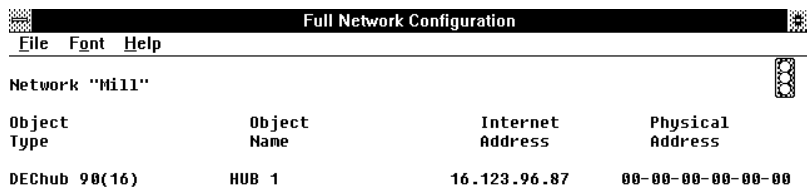
Modules
Number  Type  Name
1       RP
2       TS
4       TS
5       RP
7       AG    lab hub agent
8       BR
```

LJ-03057-SIX

Full Configuration Report

The Full Configuration report lists the sites and devices throughout the complete network (Figure 6-39). This option is available at the network, site, and hub view.

Figure 6-39 Full Configuration Report



Object Type	Object Name	Internet Address	Physical Address
DEChub 98(16)	HUB 1	16.123.96.87	00-00-00-00-00-00

LJ-02720-SIX

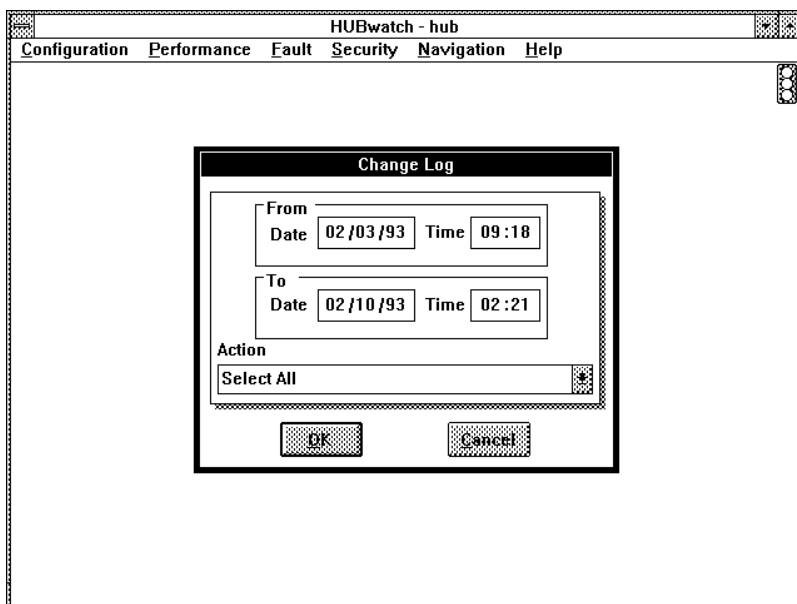
Change Log

The Change Log records the date, time, and action for any changes made to the network.

Follow this procedure to access the change log:

1. Choose the Report option from the Configuration menu.
The Report submenu appears.
2. Choose the Change Log option from the Report menu.
The Change Log dialog box appears (Figure 6–40).

Figure 6–40 Change Log Dialog Box

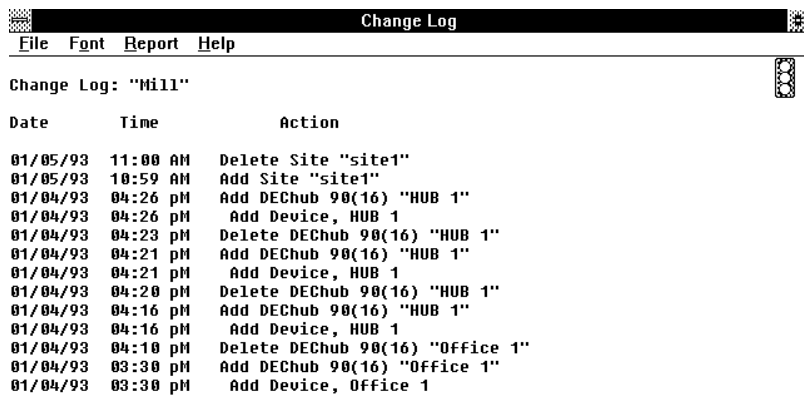


LJ-02721-SIX

3. If necessary, adjust the date and time for the period you want the report to reflect.
4. Click on the scroll bar in the Action field to choose the types of changes you want displayed in the report.
5. Choose the OK button.

HUBwatch displays the Change Log (Figure 6–41).

Figure 6–41 Change Log Report



Change Log: "Mill"

Date	Time	Action
01/05/93	11:00 AM	Delete Site "site1"
01/05/93	10:59 AM	Add Site "site1"
01/04/93	04:26 PM	Add DEChub 90(16) "HUB 1"
01/04/93	04:26 PM	Add Device, HUB 1
01/04/93	04:23 PM	Delete DEChub 90(16) "HUB 1"
01/04/93	04:21 PM	Add DEChub 90(16) "HUB 1"
01/04/93	04:21 PM	Add Device, HUB 1
01/04/93	04:20 PM	Delete DEChub 90(16) "HUB 1"
01/04/93	04:16 PM	Add DEChub 90(16) "HUB 1"
01/04/93	04:16 PM	Add Device, HUB 1
01/04/93	04:10 PM	Delete DEChub 90(16) "Office 1"
01/04/93	03:30 PM	Add DEChub 90(16) "Office 1"
01/04/93	03:30 PM	Add Device, Office 1

LJ-02722-SIX

Change Log Report Menu

The Change Log Report menu options are defined as follows:

Option	Purpose
<u>F</u> ile	Includes the following options: <ul style="list-style-type: none">• <u>S</u>ave — Saves the current change log as text or a delimited ASCII file.• <u>P</u>rint — Prints the Change Log on the defined printer.• <u>P</u>rinter <u>S</u>etup — Configures a printer for the HUBwatch application.• <u>E</u>xit Report — Exits the report and returns to the active view.
<u>F</u> ont	Provides access to the Font window allowing you to choose the font, font stem, font size, and font color for the report.
<u>R</u> eport	Includes the following options: <ul style="list-style-type: none">• <u>C</u>lear — Clears the entries in the change log.• <u>S</u>et <u>S</u>ize — Sets the maximum number of entries kept in the change log. When the maximum is exceeded, HUBwatch deletes the oldest items to make room for the newest items.
<u>H</u> elp	Provides access to online help for HUBwatch.

7

Managing Generic Devices

Introduction

This chapter explains how to manage generic devices on your network. These include any of the following devices that are not manufactured by Digital:

- Bridges
- Routers
- Hosts
- Generic devices

You can represent each of the generic devices in a graphic form on your network display. You can manage these generic network devices only to the degree that they support the Simple Network Management Protocol (SNMP). HUBwatch, however, does provide detailed control over device configuration, performance monitoring, and management. Each Device Expert understands the device-specific MIB parameters of a particular family of devices.

For Version 1.0 of HUBwatch, you must manage the DECserver 90M, DECserver 90TL, and the DECwanrouter 90 as standalone generic devices, even if they are in a hub.

Accessing Generic Devices

Follow this procedure to access information for the generic devices:

1. Position the cursor on the icon for the generic device you need to manage and double click MB1.

An enlarged version of the device icon appears in the left corner of the screen and the Device System Group window overlays the screen. From this window, you can print the report or modify the information that appears in bold type.

The following table describes each field in the Device System Group window.

Field	Description
SysDescr	The product name and version information.
SysUpTime	The total time that the agent has been running since it was last reset.
SysObjectId	The registered Object ID (OID) for the object that is currently being managed.
SysContact	The person or group to contact for service.
SysName	The name of the agent or hub.
SysLocation	The physical location of the agent.
SysServices	An encoded value representing the set of services offered.

- When you are finished using the Device System Group window, choose the Exit button to cancel the window.

Use the MIB Access menu to get and set SNMP management information from devices including bridges, routers, hosts, concentrators, workstations, and other generic SNMP-manageable devices.

The device icon remains on the screen. You can now manage the generic devices from the following menus:

To do this . . .	Use this menu . . .	Refer to . . .
Configure a generic device	<u>C</u> onfiguration	Chapter 5
Manage network performance	<u>P</u> erformance	Chapter 6
Manage network faults	<u>F</u> ault	Chapter 6
Access the MIB	<u>M</u> IB Access	Chapter 6

8

Managing DECagent 90 Modules

Introduction

This chapter describes how to manage a selected DECagent 90 using HUBwatch for Windows.

The DECagent 90 is a network agent designed for IEEE 802.3 CSMA/CD networks and uses the Simple Network Management Protocol (SNMP). You can use HUBwatch to manage the configuration, performance, and faults related to a selected DECagent.

A DECagent 90 may be installed in either of the two right-most slots (7 or 8) in any DEChub 90. It can also be installed as a standalone on a ThinWire segment anywhere on the same LAN as the bus being managed. For further information about the location of the DECagent 90, refer to *DECagent 90 User Information* manual.

Note

Prior to reading this section, you should be familiar with the *DECagent 90 User Information* manual, EK-DENMA-UI. For ordering information, refer to Appendix F.

Selecting the DECagent 90 Module

To manage a specific agent, you must define that agent by selecting it:

1. Set the display to either the hub view, to select a DECagent,

or

Set the display to the network or site view to select a standalone DECagent.

2. Choose the Zoom In option from the Navigation pull-down menu.

The cursor becomes a magnifying glass.

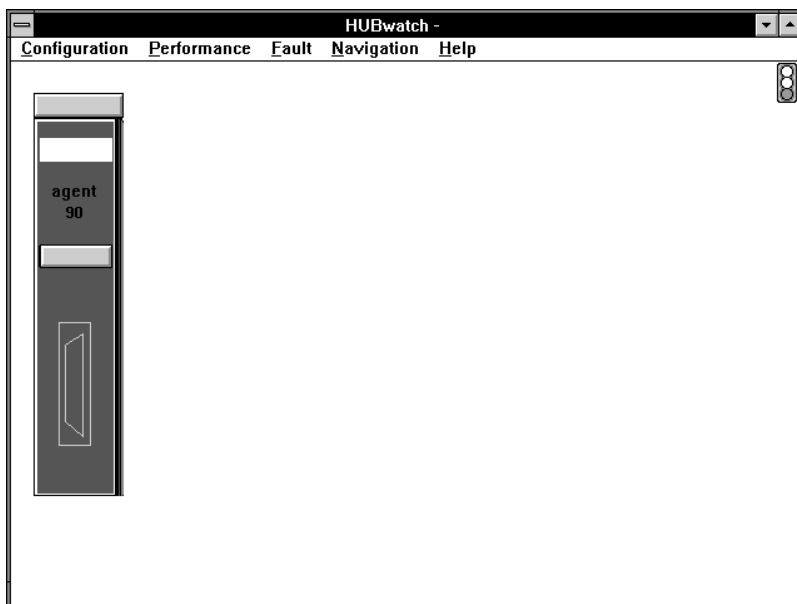
3. Position the magnifying glass on the DECagent module you need to manage and click MB1.

HUBwatch displays the module view with the DECagent module on the left side of your display (Figure 8-1).

Note

You can also access the DECagent module by double clicking MB1 on the module itself.

Figure 8-1 Selecting a DECagent Module



LJ-02750-SIX

Accessing DECagent Information

You can access DECagent information and manage the selected DECagent through the following pull-down menus:

- Configuration
- Performance
- Fault
- MIB Access

Note

The MIB Access menu is available only at the device view. It is not available at the module view.

Managing the DECagent 90 Configuration

You can manage the configuration for a DECagent at the module view or a standalone DECagent at the device view. The following table lists the options that appear at either the module or device view when you select the Configuration menu. It also lists the tasks you can perform using these options.

Option	View	Task
Report	Module	View DECagent configuration reports.
Modify	Device	Modify the selected device. This option is not module specific. For further information, refer to Modifying Devices or Hubs.
Note	Module and Device	Enables you to make an annotation about the selected device. This option is not module specific. For further information, refer to Notes.

Note

This chapter reviews the configuration information for management of the agent module only. The configuration information at other views varies. For general information about the Configuration menu, refer to Chapter 6.

DECagent Configuration Reports

Through the [Configuration](#) menu, at the module view, you can access information about the configuration of the selected DECagent. The configuration reports include the following:

- Agent Module
- Edit Strings

Agent Module Report

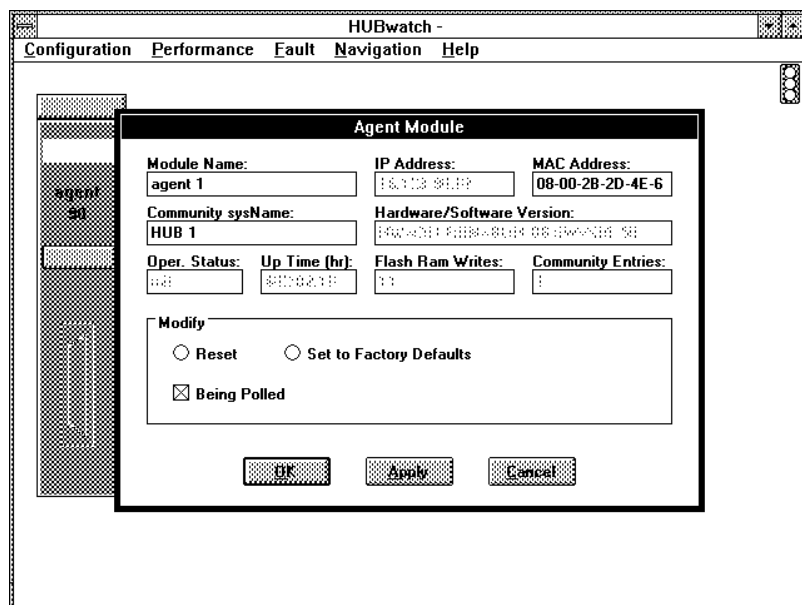
The Agent Module report (Figure 8–2) is accessed from the [Agent Info](#) option. It displays the following configuration information for the selected agent.

Field Name MIB Object	Description
Module name dh90SlotModule	The name of the agent.
IP Address dh90SlotIPAddress	The IP address of the agent.
MAC Address dh90SlotPhysicalAddress	The MAC address of the agent.
Community sysName sysName	The community name of the agent.
Hardware/Software version dh90SlotModuleVersion	The hardware and the software version numbers.
Oper. status ifOperStatus	The operational status of the agent.
Up time (hr) sysUpTime	The agent up time, in hours.
Flash RAM writes da90FlashErasures	The number of times that the flash RAM has been written to.
Community entries da90CommunityNumber	The number of communities associated with the agent.

Field Name	Description
MIB Object	Modify
Reset da90Maintenance	If selected, causes the agent to cease operations and execute its self-test. If the self-test passes, the agent is ready to accept new SNMP packets. The agent's counters are cleared by this action, but no previously stored MIB objects are affected.
Being Polled No MIB Object	Controls whether the management station is polling this agent. For a given hub, you poll all modules or you poll no modules.
Set to Factory Defaults da90Maintenance	Enables you to reset the defaults or set them back to the factory settings.

You can change any of the field definitions displayed in bold type. When completed, choose the Apply button to accept the changes or the OK button to accept the changes and dismiss the window.

Figure 8-2 Agent Module Report



LJ-02723-SIX

Edit Strings Report

The Edit Strings report (Figure 8–3) is accessed through the Edit Strings option. Each community supported by the proxy agent has its own set of unique Read only and Read/Write strings. For further information on the DECagent and communities, refer to the *DECagent 90 User Information* manual, EK-DENMA-UI.

The Edit Strings report displays the following community string information for the selected agent:

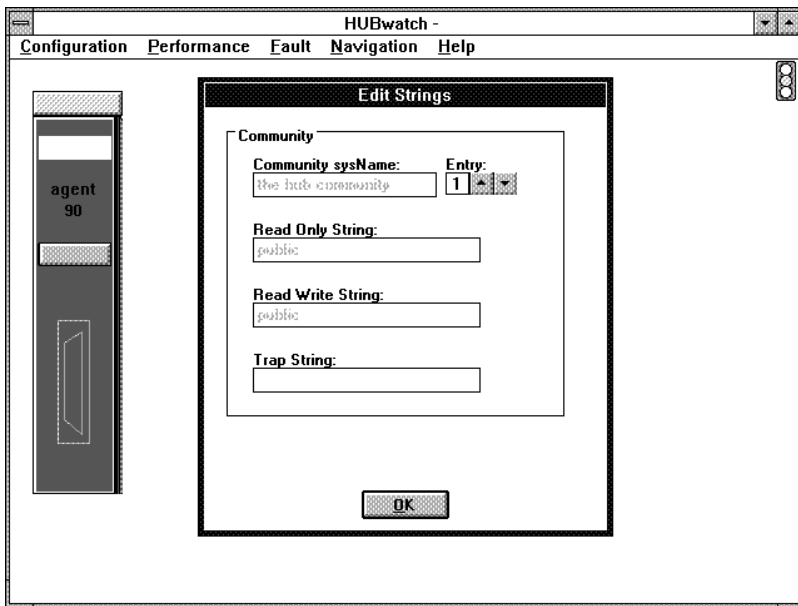
Field Name MIB Object	Description
Community sysName sysName	A user-assigned name for a community. The name should be unique within a single agent. The agent will retain up to 16 alphanumeric characters.
Read Only String da90CommunityROString	An alphanumeric string up to 32 characters in length that is checked by the agent before responding to a GET request for any MIB object. This string must be unique for each community supported by an agent.
Read Write String da90CommunityRWString	An alphanumeric string up to 32 characters in length that is checked by the agent before responding to a SET request for MIB objects. The community Read Write string must be unique for each community supported by an agent..
Trap String da90CommunityTrapString	<i>Traps are not supported in the DECagent 90 firmware, Version 1.0.</i>

To look at all the community strings, select the ↑ or ↓ in the Entry field.

Note

For this version, the Edit String report only allows you to look at the community strings for the selected agent. You cannot make any changes.

Figure 8-3 Agent Edit Strings Report



LJ-02724-SIX

Managing the DECagent 90 Performance

You can manage the performance for either a DECagent at the module view or a standalone DECagent at the device view. The following table lists the options that appear at either the module or the device view when you select the Performance menu. It also lists the tasks you can perform using these options.

Option	View	Task
Statistics	Module	View DECagent performance reports.
Graph†	Module and Device	Display an interactive graph of the count overtime for a selected MIB node and object.
Ping†	Device	Test the IP level connectivity of selected devices.

†This option is not module specific. For further information, refer to Managing Network Performance.

DECagent Performance Reports

Through the Performance menu, at the module view, you can access information about the performance of the selected DECagent. The performance reports include the following:

- DECagent Performance
- MIB Object Graph - Agent

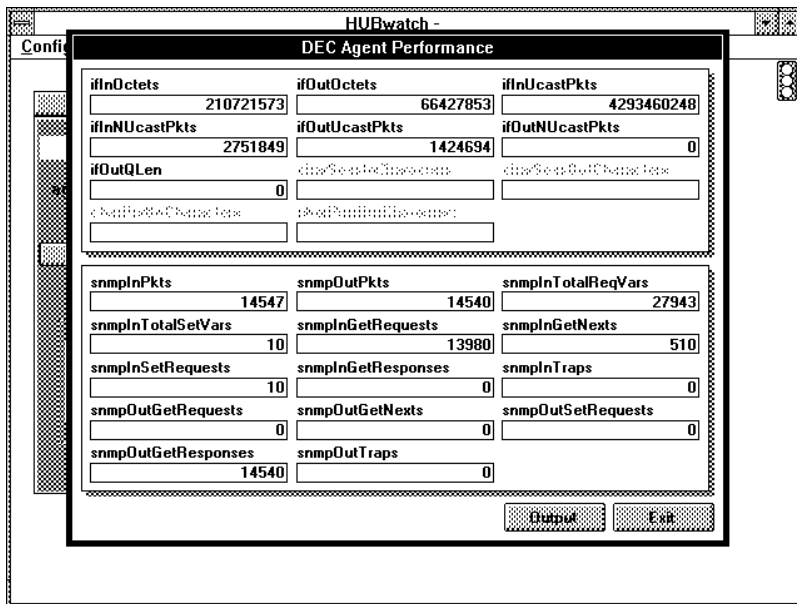
DECagent Performance Report

The DECagent Performance report (Figure 8–4) displays MIB information about the selected agent. You can access the DECagent Performance report through the Statistics option. You can only print out the information on the DECagent Performance report; you cannot change it. To print the report, refer to Printing DECagent Reports. For detailed information about the MIB entries, refer to Appendix E.

MIB Object Graph - Agent

The MIB Object Graph report displays an interactive graph of the count overtime for a selected MIB node and object. You access the report through the Graph option. For further information, refer to Graph Option.

Figure 8-4 DECagent Performance Report



LJ-02725-SIX

Managing the DECagent 90 Faults

You can manage the faults for either a DECagent at the module view or a standalone DECagent at the device view. The following table lists the options that appear at either the module or device view when you select the Fault menu. It also lists the tasks you can perform using these options.

Option	View	Task
Error Statistics	Module	Look at the DECagent Error Reports.
Set Thresholds	Module	Set threshold counters for the selected module.
Audible Alarms†	Module and Device	Set alarms so they are audible or inaudible.
Alarms†	Module and Device	Access the Current Alarms Network report.
Report†	Module and Device	Access the Alarm Log.

†This option is not module specific. For more detailed information, refer to Managing Network Faults.

Viewing DECagent Error Report

The DECagent Error report lists the pre-defined MIB objects for the selected agent and the associated errors.

To access the report, do the following:

1. Set the view to the agent you need to effect.
2. Pull down the Fault menu.
3. Choose the Error Statistics option.

The DECagent Error report appears (Figure 8–5).

Figure 8–5 DECagent Error Report

DEC Agent Error		
ifInDiscards	ifInErrors	ifInUnKnownsProtos
21	3	44364
ifOutDiscards	ifOutErrors	
0	0	
snmplnBadVersions	snmplnBadCommunityNames	snmplnBadCommunityUses
0	3	0
snmplnASNParseErrs	snmplnTooBig	snmplnNoSuchNames
3	0	0
snmplnBadValues	snmplnReadOnly	snmplnGenErrs
0	0	0
snmpOutTooBig	snmpOutNoSuchNames	snmpOutBadValues
0	40	2
snmpOutGenErrs		
0		

Output Exit

LJ-02727-SIX

You can either view or print the DECagent Error report. You cannot change any of the fields. For detailed information on each of the MIBs displayed in the DECagent Error report, refer to Appendix E. To print the report, refer to Printing DECagent Reports.

Setting Thresholds - DECagent

The Set Thresholds window displays the thresholds for the selected agent.

To access the Set Thresholds window, do the following:

1. Select the agent for which you want to view the threshold settings.
2. Choose the Set Threshold option from the Fault menu.

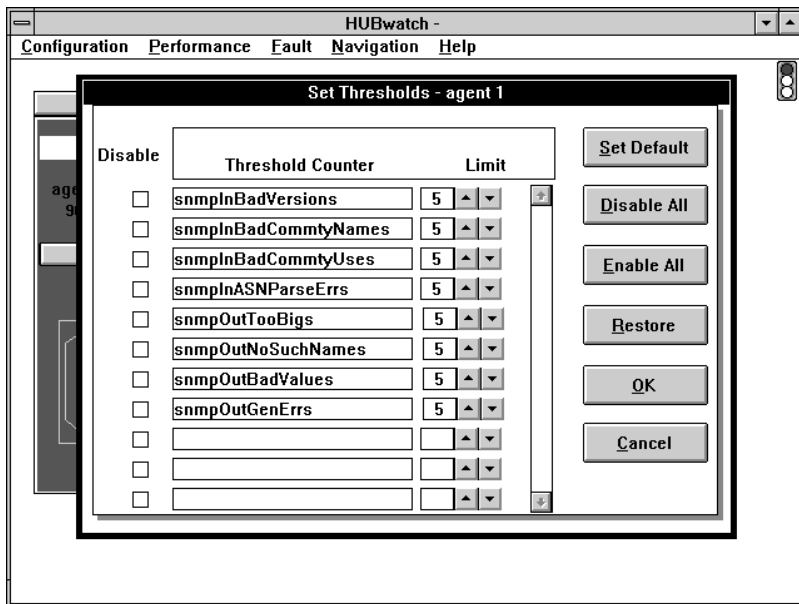
The Set Thresholds window for the selected agent appears (Figure 8–6).

You can do any of the following to the threshold counters for the selected agent.

To perform this task . . .	Do this . . .
Set the thresholds to their default.	Choose the <u>S</u> et Default button.
Disable all the threshold counters.	Choose the <u>D</u> isable All button.
Enable all the threshold counters.	Choose the <u>E</u> nable All button.
Restore the threshold counters to their original setting.	Choose the <u>R</u> estore button.
Change specific threshold counter limits.	Choose ↑ or ↓ for the setting you need to change.

3. Make the necessary changes to the threshold counters and choose the OK button to accept the changes and dismiss the window.

Figure 8–6 Set Thresholds Window



LJ-02690-SIX

Accessing the MIB

The MIB Access menu is not module specific. You can only access the MIB for a generic or standalone device at the device view. For a DEChub module, you must be at the hub view. For further information, refer to Accessing the Management Information Base in the Managing Networks chapter.

Printing DECagent Reports

Several of the report windows provide an option for printing the report. This is indicated by an Output button at the bottom of the window.

To print a report, do the following:

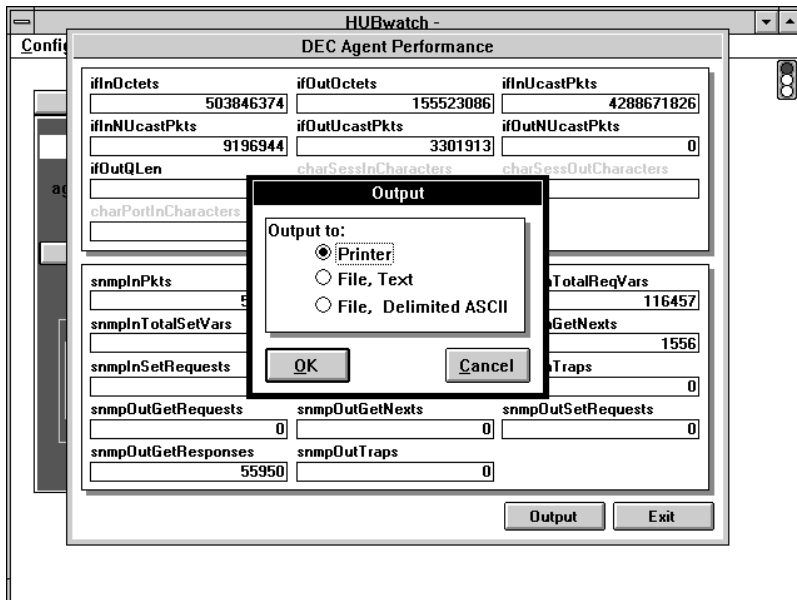
1. Choose the Output button.
The Output dialog box appears (Figure 8–7).
2. Choose one of the following options for output:

Printer
File, Text

File, Delimited ASCII

3. Choose the OK button.

Figure 8-7 Output Dialog Box



LJ-02920-SIX

A

Menus

Introduction

This appendix contains each of the pull-down menus used within the different views for HUBwatch for Windows. The pull-down menus include:

- Network View
- Site View
- Hub View
- Module View
- Device View

Network View Menus

Figure A-1 displays the pull-down menus for the Network View.

Figure A-1 Network View Menus

Network View

Configuration	Performance	Fault	Security	Navigation	Help
<ul style="list-style-type: none"> <u>N</u>etwork <u>O</u>pen <u>N</u>ew <u>B</u>ackup <u>R</u>estore <u>D</u>elete <u>M</u>odify <u>A</u>uto Discovery <u>P</u>rinter Setup <u>A</u>dd <u>S</u>ite <u>D</u>evice <u>C</u>onnection <u>O</u>ther <u>D</u>elete <u>D</u>evice <u>C</u>onnection <u>O</u>ther <u>M</u>ove <u>R</u>eport <u>L</u>ocal Configuration <u>F</u>ull Configuration <u>C</u>hange Log <u>N</u>ote <u>E</u>xit HUBwatch 	<ul style="list-style-type: none"> <u>S</u>tatistics N/A 	<ul style="list-style-type: none"> <u>E</u>rror Statistics N/A <u>A</u>larms <u>A</u>udible Alarms <u>R</u>eport <u>A</u>larm Log 	<ul style="list-style-type: none"> <u>N</u>etwork Password <u>R</u>eport <u>S</u>ecurity Log 	<ul style="list-style-type: none"> <u>F</u>ind <u>F</u>ind <u>M</u>e <u>Z</u>oom In 	<ul style="list-style-type: none"> <u>I</u>ndex F1 <u>C</u>ommands <u>P</u>rocedures <u>G</u>lossary <u>U</u>sing <u>H</u>elp <u>A</u>bout HUBwatch...








N/A - This option is not available at this level.

Site View Menus

Figure A-2 contains the pull-down menus for the site view.

Figure A-2 Site View Menu

Site View

Configuration	Performance	Fault	Navigation	Help
<u>M</u> odify	<u>S</u> tatistics N/A	<u>E</u> rror Statistics N/A	<u>F</u> ind	<u>I</u> ndex F1
<u>A</u> dd		<u>A</u> larms	<u>F</u> ind Me	<u>C</u> ommands
<u>S</u> ite		<u>A</u> udible Alarms	<u>Z</u> oom In	<u>P</u> rocedures
<u>D</u> evice		<u>R</u> eport	<u>Z</u> oom Out	<u>G</u> lossary
<u>C</u> onnection		<u>A</u> larm Log	<u>Z</u> oom <u>T</u> op	<u>U</u> sing Help
<u>O</u> ther				<u>A</u> bout HUBwatch...
<u>D</u> elete				
<u>D</u> evice				
<u>C</u> onnection				
<u>O</u> ther				
<u>M</u> ove N/A				
<u>R</u> eport				
<u>L</u> ocal Configuration				
<u>F</u> ull Configuration				
<u>C</u> hange Log				
<u>N</u> ote				
<u>E</u> xit HUBwatch				

LJ-02874-T10

N/A - This option is not available at this level.

Hub View Menus

Figure A-3 contains the pull-down menus for the hub view.

Figure A-3 HUB View Menus

HUB View

<u>C</u> onfiguration	<u>P</u> erformance	<u>F</u> ault	<u>N</u> avigation	<u>M</u> IB Access	<u>H</u> elp
Add	Statistics N/A	Error Statistics	F <u>ind</u>	MIB11	<u>I</u> ndex F1
<u>D</u> elete	<u>G</u> raph	<u>A</u> larms	F <u>ind</u> Me	<u>S</u> ystem	<u>C</u> ommands
<u>M</u> odify	<u>P</u> ing	A <u>udible</u> Alarms	<u>Z</u> oom In	<u>I</u> TF	<u>P</u> rocedures
M <u>o</u> ve		R <u>e</u> port	<u>Z</u> oom <u>O</u> ut	<u>I</u> P	<u>G</u> lossary
R <u>e</u> port		<u>A</u> larm Log	<u>Z</u> oom <u>T</u> op	<u>G</u> roup	Using <u>H</u> elp
<u>B</u> rief Configuration				<u>A</u> ddress Table	<u>A</u> bout HUBwatch...
<u>F</u> ull Configuration				<u>R</u> outing Table	
<u>N</u> ote				<u>N</u> et-To-Media Table	
<u>E</u> xit HUBwatch				<u>I</u> CMP	
				<u>T</u> CP	
				<u>U</u> DP	
				<u>E</u> GP	
				<u>S</u> NMP	
				<u>B</u> ridge	
				<u>D</u> ot1dBase	
				<u>D</u> ot1dStp	
				<u>D</u> ot1dSr	
				<u>D</u> ot1dTp	
				<u>D</u> otdStatic	
				<u>C</u> har-like	
				<u>G</u> eneral	
				<u>S</u> ession	
				<u>R</u> S232-like	
				<u>G</u> eneral	
				<u>A</u> sync	
				<u>S</u> ync	
				<u>M</u> IB <u>E</u> xt	

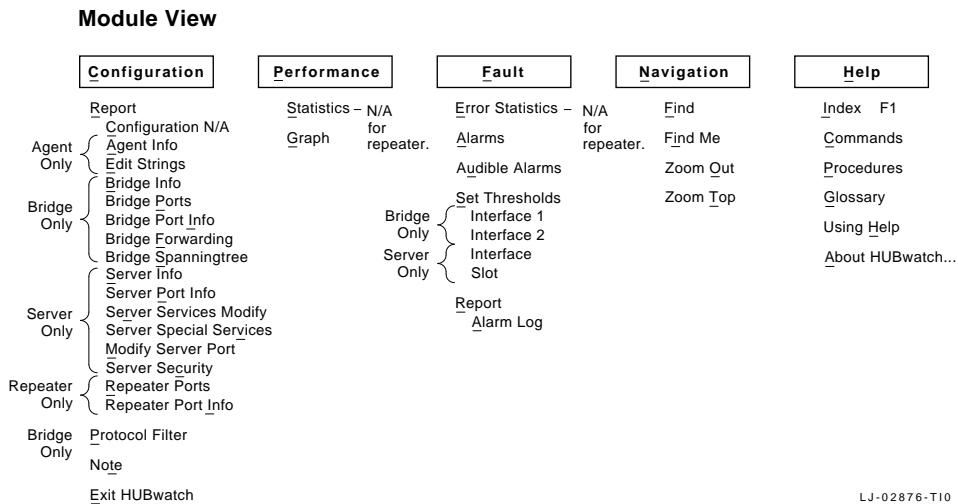
LJ-02875-T10

N/A - This option is not available at this level.

Module View Menus

Figure A-4 contains the pull-down menus for the module view.

Figure A-4 Module View Menus



N/A - This option is not available at this level.

Device View Menus

Figure A-5 contains the pull-down menus for the device view.

Figure A-5 Device View Menus

Device View

Configuration	Performance	Fault	Navigation	MIB Access	Help
Modify	Graph	Alarms	Find	MIB11	Index F1
Report N/A	Ping	Audible Alarms	Find Me	Bridge	Commands
Note		Report	Zoom Out	Char-like	Procedures
Exit HUBwatch		Alarm Log	Zoom Top	RS232-like	Glossary
				MIB Ext	Using Help
					About HUBwatch...

LJ-02878-T10

N/A - This option is not available at this level.

B

NOS with HUBwatch

Introduction

This appendix explains how to use HUBwatch on a PC that has a network operating system (NOS) in use.

HUBwatch uses a Clarkson Packet Driver to control the Network Interface Card (NIC) in the PC. It is assumed that all upper-layer protocols will interface to the network through the packet driver. This requirement is incompatible with most network operating systems, since each NOS also expects its primary protocol to communicate over a different driver such as NDIS for Digital's PATHWORKS and Microsoft's LAN Manager or ODI for Novell's Netware. A future version of HUBwatch will allow the use of NDIS or ODI drivers and will alleviate this problem.

For now, the PC that is running HUBwatch must also be used with a NOS. To enable this, there are two procedural approaches you should be familiar with:

- Printing HUBwatch reports on a networked printer
- Switching between HUBwatch and your NOS

Using a Networked Printer

If you need to print various HUBwatch reports to a printer that you normally use with your NOS, use the following procedure:

Step	Action
1.	Before printing, use Windows to redirect the output to a file.
2.	Exit HUBwatch and reboot the PC to run the NOS.

Step	Action
3.	Print the file.

To redirect the Output to a File

1.	From the Window Program Manager Main Program Group, select Print Manager.
2.	From the Options menu, choose Printer Setup.
3.	In the Installed Printers dialog box, select the name of the printer you want to use.
4.	Choose the Connect button.
5.	In the Ports dialog box, select File as your port and choose the OK button.
6.	Choose the Close button.

For more information, refer to *Microsoft Word for Windows User's Guide*, Version 2.x.

Switching Between HUBwatch and Your NOS

You must reboot your PC each time you want to change from running your NOS to running HUBwatch. The CONFIG.SYS and AUTOEXEC.BAT files will be different depending upon which operating environment you need to run after booting your system.

There are two ways to expedite the process of modifying these files:

- Keep two versions of each on your hard disk.
- Keep the HUBwatch version of these files on a special boot diskette.

Two Versions on Your Hard Disk

To keep two versions of AUTOEXEC and CONFIG on your hard disk, use the following procedure:

Step	Action
1.	Edit AUTOEXEC.BAT to replace all commands related to your network and NOS startup with the command that starts the packet driver for your NIC. Then save this file as AUTOEXEC.HUB.
2.	Edit CONFIG.SYS to replace the loading of your normal network driver with NETDEV.SYS and save this file as CONFIG.HUB.

Step	Action
3.	Copy your existing AUTOEXEC.BAT file to AUTOEXEC.NOS and your CONFIG.SYS file to CONFIG.NOS.
4.	Create a new file called HUBBOOT.BAT that renames AUTOEXEC.HUB to AUTOEXEC.BAT and CONFIG.HUB to CONFIG.SYS.
5.	Create a second new file call NOSBOOT.BAT that renames AUTOEXEC.NOS to AUTOEXEC.BAT and CONFIG.NOS to CONFIG.SYS.
6.	Before rebooting to run HUBwatch, run HUBBOOT.BAT, or before rebooting to run your NOS, run NOSBOOT.BAT.

Note

In the future, if you must make changes to AUTOEXEC.BAT or CONFIG.SYS, you will need to make the same edits to both the .HUB file and the .NOS file rather than the .BAT file.

Booting Your PC from a Diskette

To create a boot diskette, use the following procedure:

Step	Action
1.	Format a diskette as a system disk.
2.	Edit AUTOEXEC.BAT to replace all commands related to your network and NOS startup with the command that starts the packet driver for your NIC. Then save this file as AUTOEXEC.BAT on the diskette.
3.	Edit CONFIG.SYS to replace the loading of your normal network driver with NETDEV.SYS and save this file as CONFIG.SYS. on the diskette.

Note

If you make changes to your AUTOEXEC.BAT or CONFIG.SYS, which are unrelated to HUBwatch or your NOS, you must make the changes on both the hard drive and the boot diskette.

C

Clarkson Packet Drivers

Introduction

This appendix provides the following examples:

- Installation of the Clarkson Packet Driver with a Digital EtherWORKS Adapter and NETDEV.SYS
- Customizing the installed Clarkson Packet Driver

Overview

HUBwatch uses public domain Clarkson-type packet driver interfaces for Ethernet cards and operates with any card for which a Clarkson driver is available. A set of some common Clarkson packet drivers is included on diskette 2.

The Clarkson family of drivers was previously available through Clarkson University. These drivers are currently part of the Crynwr Packet Driver Collection. Questions should be directed to:

Russell Nelson
Crynwr Software
11 Grant Street
Potsdam, NY 13676
(315) 268-1925

Or send Internet mail to:

`NELSON@CRYNWR.COM`

Note

The distribution of these programs does not imply that Digital Equipment Corporation is responsible for the support or performance of the Crynwr Packet Drivers.

Note

Prior to installing the driver, you should install the Ethernet card. Follow the installation instructions supplied by the card manufacturer. Be sure to record the interrupt, I/O, and Base Memory Address settings.

Packet drivers for the NICs are available only from the NIC manufacturer.

Installation with a Digital EtherWORKS Adapter and NETDEV.SYS

The following example describes how to install the Clarkson packet driver (DEPCA.COM) for use with a Digital EtherWORKS PC/AT adapter and the device driver NETDEV.SYS.

1. Create a subdirectory C:\HUBWATCH> for the packet driver files you require.

Note

This is already created if the HUBwatch installation is completed.

2. Enter the following:

```
COPY C:\HUBWATCH
```

3. Press .

4. Copy the files for your card type from diskette 2 to the new subdirectory. For example, move to the C:\HUBWATCH> subdirectory and enter the following:

```
COPY A:\HUBWATCH\DEPCA.COM
```

Use different drive designations or directory names, if required.

5. Press .

6. Edit your AUTOEXEC.BAT file. To invoke the packet driver at startup, you must add the following command line:

```
C:\HUBWATCH\DEPCA P1 P2 P3 P4
```

For example:

```
C:\HUBWATCH\DEPCA 0x69 5 0x300 0xD000
```

These are represented as follows for the factory default EtherWORKS setting.

Value	Description
P1 - Software interrupt	The software interrupt level is a hexadecimal number in the range 0x60 to 0x80. (0x nn is the notation for a hexadecimal number and must be used here and in P3 and P4.) Try 0x69 first.
P2 - Hardware interrupt	The hardware interrupt value used by the card is usually 5, but another value may be required for your system to avoid conflict with another card.
P3 - I/O Port	The base address for the card's I/O port. This value is usually 0x300 (the factory default). You can change this value during card installation.
P4 - Memory address	The base address for the card's on-board memory. The factory default is 0xD000.

Note

The use and values for P1 through P4 may be different for your adapter card. Please check the file INSTALLATION.DOC on diskette 2 for the correct values.

Customization of the Installed Clarkson Packet Driver

To customize the installed Clarkson packet driver, do the following:

1. In the C:\HUBWATCH> directory, run CUSTOM to customize the device driver. Enter the following:

```
CUSTOM NETDEV.SYS
```

2. Press **[Return]**. The PC Network Customizer menu displays followed by a list of defaults and menu options.

Note

When selecting menu options, use lowercase letters. Each single-letter command is obeyed immediately. You will not have to press **[Return]**. CUSTOM ignores invalid input.

3. Customize the site:
 - a. Enter **s** at the Enter Command prompt to select the Do Site Customizations option.

The Site Customization menu displays. You must make one site-specific customization here. The remaining values can be left as they are.
 - b. Enter **a** at the Enter Command prompt to select the Set my internet address option.

The following prompt displays:

```
Enter my internet address
```
 - c. Enter the internet address and press **[Return]**.

The new address is immediately reflected in the list of settings at the top of your screen.
 - d. Enter any other relevant information.
 - e. Press **[Esc]** to return to the PC Network Customizer menu.
4. Customize the hardware:
 - a. Enter **h** at at the Enter Command prompt to select the Do hardware customizations option.

The Hardware Customization menu displays.
 - b. Check the settings at the top of the screen to verify that the Interrupt vector, I/O address and Base memory values are correct, then set the Ethernet Address.
 - c. Enter **e** at the Enter Command prompt to select the Set Ethernet address option.

The EtherNet address menu displays.
 - d. Enter **h** at the Enter Command prompt to select the Use hardware address option.

The following message displays at the top of the screen:

Uses hardware EtherNet address

- e. Press **[Esc]** to return to the Hardware Customization menu.
- f. Enter **n** at the Enter Command prompt to select the Set number of net interfaces on this machine option.

The following prompt displays:

Enter the number of net interfaces on this machine

- g. Enter the appropriate number and press **[Return]**.
The given number of net interfaces displays at the top of the screen.
- h. Enter any other relevant information.
- i. When completed, press **[Esc]** to return to the PC Network Customizer menu.

5. Enter **e** at the Enter Command prompt to exit and save the changes you have made.

Note

If you enter **q** to quit, your changes are not saved.

6. Edit your CONFIG.SYS file to add the command line that installs the network device driver.

```
DEVICE=C:\HUBWATCH\NETDEV.SYS
```

7. Reboot your PC.
8. When the DEPCA configuration command is run in AUTOEXEC.BAT, you should see some information about the packet driver. Check that the driver is installed. If the system displays an error message indicating that the driver was not installed, return to step 3. Also verify that the Ethernet address is correct for your card. Values of FF-FF-FF-FF-FF-FF or 00-00-00-00-00-00 are not valid.

9. To test the driver, run Ping:

```
C:\HUBWATCH\PING -T [TARGET IP ADDRESS]
```

Note

Target Internet Address is the IP address of a known good device. You will not be able to Ping the PC on which HUBwatch is installed. You also

cannot run Ping successfully from a DOS shell with Windows running.

10. Press **[Return]**.

At the bottom of the Ping display, you should see a line similar to one of the following:

Line similar to . . .	Indicates . . .
# of tries = 206, successes = 206	The setup works correctly.
# of tries = 3, successes = 0	The driver is not installed properly. You must start again with the customization procedure.

You can obtain more detailed information about driver installation failures by modifying the settings in NETDEV.SYS under the Set Debug options.

D

Background Maps

Introduction

This appendix contains a sampling of products that may be used to generate background geographical maps for use with HUBwatch for Windows. Most of the products deal with the U.S. geography only. Also, in most cases, these products do not export .BMP files directly. The output format can, however, be run through a conversion utility to get it into .BMP format.

Product Name	Vendor	Telephone
AUTOMAP for Windows	AUTOMAP Inc. 9831 South 51st Street Suite C113 Phoenix, AZ 85044-9930	800-545-6626 601-893-2400
Expert Maps	Expert Software P.O. Box 143376 Coral Gables, FL 33134	800-759-2562 FAX 305-443-3255
Keymap	SoftKey Products	Mail Order 800-851-2917 FAX 415-345-5575
MapExpert Street Atlas USA	DeLorme Mapping Main St. P.O. Box 298 Freeport, ME 04032	800-452-5931 207-865-1234 FAX 207-865-9628
Maps and Data	MapInfo Troy, NY	800-327-8627 518-274-6000

Many of these products, as well as others, may be available through a local PC software retailer.

E

MIB Descriptions

Introduction

This appendix contains a description for each of the MIBs associated with HUBwatch for Windows.

- MIBII Group
- Bridge MIB Group
- Character MIB Group
- RS232 MIB Group
- DH90 MIB Group

MIBII Groups

This section contains a description for each of the fields within the following MIBII Groups:

- System
- Interface (ITF)
- Address Translation (AT)
- Implementation (IP)
- Internet Control Message Protocol (ICMP)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Exterior Gateway Protocol (EGP)
- Simple Network Management Protocol (SNMP)

System Group

sysDescr

A text description of the entity. This value may include the full name and version identification of the system's hardware type, software operating-system and networking software. This entry must contain printable ASCII characters.

sysUpTime

The time (in hundredths of a second) since the network management portion of the system was last re-initialized.

sysObjectID

The vendor's authoritative identification of the network management subsystem contained in the entity.

sysContact

The text identification of the contact person for the managed node.

sysName

A name assigned by the administrator for the managed node.

sysLocation

The physical location of the node.

sysServices

A value that indicates the set of services the entity offers. The value is a sum that initially takes the value zero. Then for each layer, L, in the range 1 - 7, that the device performs a transaction for, the number 2^{L-1} is added to the value. For example, a device that only performs routing functions would have a value of $2^{3-1} = 4$. The layers are as follows:

Layer	Functionality
1	Physical (for example, repeaters)
2	Data-link (for example, bridges)
3	Internet (for example, supports IP)
4	End-to-end (for example, supports TCP)
7	Application (for example, supports SMTP)

For systems including OSI protocols, layers 5 and 6 may also be counted.

Interface (ITF) Group**ifTable**

A list of interface entries. The number of entries is given by the value of ifNumber.

ifIndex

A unique value for each interface. The value ranges between 1 and the value of ifNumber.

ifDescr

A text string containing information about the interface. This should include the name of the manufacturer, the product name, and the version of the hardware interface.

ifType

The type of interface, distinguished according to the physical/link protocol(s) immediately below the network layer in the protocol stack.

ifMtu

The size of the largest datagram that can be sent and/or received on the interface. It is specified in octets.

ifSpeed

An estimate of the interface's current bandwidth in bits per second.

ifPhysAddress

The interface's address at the protocol layer immediately below the network layer in the protocol stack.

ifAdminStatus

The desired state of the interface. Possible values are as follows:

Value	Description
1	up: ready to pass packets
2	down
3	testing (no packets can be passed)

ifOperStatus

The current operational state of the interface. The values are the same as for ifAdminStatus.

ifLastChange

The value of sysUpTime at the time the interface entered its current operational state. If the current state is entered prior to the last re-initialization of the local network management subsystem, this object contains a zero value.

ifInOctets

The total number of octets received on the interface.

ifInUcastPkts

The number of subnetwork-unicast packets delivered to a higher-layer protocol.

ifInNUcastPkts

The number of non-unicast (subnetwork broadcast or subnetwork multicast) packets delivered to a higher level.

ifInDiscards

The number of inbound packets discarded without error detection. This is often done to free up buffer space.

ifInErrors

The number of inbound packets not delivered to a higher layer protocol because of errors.

ifInUnknownProtos

The number of packets received by the interface, but discarded due to an unknown or unsupported protocol.

ifOutOctets

The total number of octets transmitted out of the interface. This includes framing characters.

ifOutUcastPkts

The total number of packets that higher level protocols request for transmission to a subnetwork-unicast address. This includes those discarded or not sent.

ifOutNUcastPkts

The total number of packets that higher level protocols request for transmission to a non-unicast address. This includes those discarded or not sent.

ifOutDiscards

The number of outbound packets discarded due to resource limitations.

ifOutErrors

The number of outbound packets discarded due to errors.

ifOutQLen

The length of the output packet queue.

ifSpecific

A reference to MIB definitions specific to the particular media used to realize the interface.

Address Translation (AT) Group

Implementation of the Address Translation group is mandatory for all systems.

atTable

The Address Translation tables contain the NetworkAddress to physical address equivalences.

atEntry

Each entry contains one NetworkAddress to physical address equivalence.

atIfIndex

The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

atPhyAddress

The media-dependent physical address. Setting this object to a null string invalidates the corresponding entry in the atTable object.

atNetAddress

The NetworkAddress corresponding to the media-dependent physical address.

Internet Protocol (IP) Group

Implementation of the IP group is mandatory for all systems.

ipForwarding

Indicates whether this entity is acting as a gateway or as a host.

ipDefaultTTL

The default value, for the Time-To-Live field of the IP packets.

ipInReceives

The total number of input datagrams received from interfaces, including those received in error.

ipInHdrErrors

The number of input datagrams discarded due to errors in their IP headers, including bad checksums, version number mismatch, other format errors, time-to-live exceeded, and errors discovered in processing their IP options.

ipInAddrErrors

The number of input datagrams discarded because the IP address in their IP header's destination field is not a valid address. This count includes invalid addresses (for example, 0.0.0.0) and addresses of unsupported Classes (for example, Class E). For entities that are not IP Gateways and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address.

ipForwDatagrams

The number of input datagrams for which this entity is not their final IP destination. As a result an attempt is made to find a route to forward them to the final destination. In entities that do not act as IP Gateways, this counter includes only those packets that are Source-Routed by this entity.

ipInUnknownProtos

The number of locally-addressed datagrams received successfully but discarded because of an unknown or unsupported protocol.

ipInDiscards

The number of input IP datagrams for which no problems are encountered to prevent their continued processing, but which are discarded (for example, for lack of buffer space). Note that this counter does not include any datagrams discarded while awaiting re-assembly.

ipInDelivers

The total number of input datagrams successfully delivered to IP user protocols (including ICMP).

ipOutRequests

The total number of IP datagrams that local IP user protocols (including ICMP) supply to IP in requests for transmission. Note that this counter does not include any datagrams counted in ipForwDatagrams.

ipOutDiscards

The number of output IP datagrams for which no problem is encountered to prevent their transmission to their destination, but which are discarded due to resource limitations.

ipOutNoRoutes

The number of IP datagrams discarded because no route could be found to transmit them to their destination.

ipReasmTimeout

The maximum number of seconds which received fragments are held while they are awaiting re-assembly at this entity.

ipReasmReqds

The number of IP fragments received needing reassembly.

ipReasmOKs

The number of IP datagrams successfully reassembled.

ipReasmFails

The number of reassembly failures.

ipFragOKs

The number of IP datagrams successfully fragmented at this entity.

ipFragFails

The number of IP datagrams discarded because they need to be fragmented at this entity but can not. This may be due to the setting of the Don't Fragment flag.

ipFragCreates

The number of IP datagram fragments generated as a result of fragmentation at this entity.

IP Address Table**ipAdEntAddr**

The IP address to which this entry's addressing information pertains.

ipAdEntIfIndex

The index value that uniquely identifies the interface to which this entry is applicable. The interface identified here is the same interface identified in `ifIndex`.

ipAdEntNetMask

The subnet mask associated with the IP address of this entry. The value of the mask is an IP address with all the network bits set to 1 and all the hosts bits set to 0.

ipAdEntBcastAddr

The value of the least significant bit in the IP broadcast address used for sending datagrams on the (logical) interface associated with the IP address of this entry.

ipAdEntReasmMaxSize

The size of the largest IP datagram that this entry can reassemble from incoming IP fragmented datagrams received on this interface.

IP Routing Table**ipRouteDest**

The destination IP address of this route. An entry of the value of 0.0.0.0 is considered a default route. Multiple routes to a single destination can appear in the table, but access to such multiple entries is dependent on the table-access mechanisms defined by the network management protocol in use.

ipRouteIfIndex

The index value that uniquely identifies the local interface through which the next hop of this route should be reached. The interface identified by a particular value of this index is the same interface as identified by the same value of `ifIndex`.

ipRouteMetric1

The primary routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route's `ipRouteProto` value. If this metric is not used, its value should be set to -1.

ipRouteMetric2

See `ipRouteMetric1` for a definition.

ipRouteMetric3

See `ipRouteMetric1` for a definition.

ipRouteMetric4

See ipRouteMetric1 for a definition.

ipRouteNextHop

The IP address of the next hop (gateway IP address for indirect routing) of this route.

ipRouteType

The type of route.

ipRouteProto

The routing mechanism by which this route was determined. Inclusion of values for gateway routing protocols is not intended to imply that hosts should support those protocols.

Value	Description
1	other: none of the following
2	local: non-protocol information
3	netmgmt: set by network management protocol
4	icmp: obtained by ICMP (e.g., Redirect)

The remaining values are all gateway routing protocols:

Value	Description
5	egp
6	ggp
7	hello
8	rip
9	is-is

ipRouteAge

The number of seconds since this route was last updated or otherwise determined to be correct.

ipRouteMask

Indicates the subnet mask for route. For those systems that do not support arbitrary subnet masks, an agent constructs the value of the ipRouteMask by determining whether the value of the correspondent ipRouteDest field belongs to a class A, B, or C network, and then using one of the following:

Mask	Network
255.0.0.0	Class A
255.255.0.0	Class B
255.255.255.0	Class C

If the value of the ipRouteDest is 0.0.0.0 (a default route), the mask value is also 0.0.0.0. It should be noted that all IP routing subsystems implicitly use this mechanism.

IP Address Translation Table (Net-to-Media Table)**ipNetToMediaIfIndex**

The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

ipNetToMediaPhysAddress

The media-dependent physical address.

ipNetToMediaNetAddress

The IP Address corresponding to the media-dependent physical address.

ipNetToMediaType

The type of mapping.

Internet Control Message Protocol (ICMP) Group

Implementation of the ICMP group is mandatory for all systems.

icmplnMsgs

The total number of ICMP messages that the entity received.

icmplnErrors

The number of ICMP messages that the entity received but determined as having ICMP-specific errors.

icmpInDestUnreachs

The number of ICMP Destination Unreachable messages received.

icmpInTimeExcds

The number of ICMP Time Exceeded messages received.

icmpInParmProbs

The number of ICMP Parameter Problem messages received.

icmpInSrcQuenchs

The number of ICMP Source Quench messages received.

icmpInRedirects

The number of ICMP Redirect messages received.

icmpInEchos

The number of ICMP Echo (request) messages received.

icmpInEchoReps

The number of ICMP Echo Reply messages received.

icmpInTimestamps

The number of ICMP Timestamp (request) messages received.

icmpInTimestampReps

The number of ICMP Timestamp Reply messages received.

icmpInAddrMasks

The number of ICMP Address Mask Request messages received.

icmpInAddrMaskReps

The number of ICMP Address Mask Reply messages received.

icmpOutMsgs

The number of ICMP messages that this entity attempted to send.

icmpOutErrors

The number of ICMP messages this entity did not send due to problems discovered with ICMP.

icmpOutDestUnreachs

The number of ICMP Destination Unreachable messages sent.

icmpOutTimeExcds

The number of ICMP Time Exceeded messages sent.

icmpOutParmProbs

The number of ICMP Parameter Problem messages sent.

icmpOutSrcQuenchs

The number of ICMP Source Quench messages sent.

icmpOutRedirects

The number of ICMP Redirect messages sent. For a host, this object is always zero since hosts do not send redirects.

icmpOutEchos

The number of ICMP Echo (request) messages sent.

icmpOutEchoReps

The number of ICMP Echo Reply messages sent.

icmpOutTimestamps

The number of ICMP Timestamp (request) messages sent.

icmpOutTimestampReps

The number of ICMP Timestamp Reply messages sent.

icmpOutAddrMasks

The number of ICMP Address Mask Request messages sent.

icmpOutAddrMaskReps

The number of ICMP Address Mask Reply messages sent.

Transmission Control Protocol (TCP) Group

Implementation of the TCP group is mandatory for all systems.

tcpRtoAlgorithm

The algorithm used to determine the timeout value used for retransmitting unacknowledged octets. The values may be one of the following:

Value	Description
1	other: none of the following
2	constant: a constant rto
3	rsre: MIL-STD-1778
4	vanj: Van Jacobson's algorithm

tcpRtoMin

The minimum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.

tcpRtoMax

The maximum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.

tcpMaxConn

The limit on the total number of TCP connections the entity can support.

tcpActiveOpens

The number of times TCP connections make a distant transition to the SYN-SENT state from the CLOSED state.

tcpPassiveOpens

The number of times TCP connections make a direct transition to the SYN-RCVD state from the LISTEN state.

tcpInErrs

The total number of segments received in error.

tcpOutRsts

The number of TCP segments sent.

tcpAttemptFails

The number of times TCP connections make a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections make a direct transition to the LISTEN state from the SYN-RCVD state.

tcpEstabResets

The number of times TCP connections make a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.

tcpCurrEstab

The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.

tcpInSegs

The total number of segments received, including those received in error. This count includes segments received on currently established connections.

tcpOutSegs

The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.

tcpRetransSegs

The total number of TCP segments transmitted containing one or more previously transmitted octets.

TCP Connection Table**tcpConnTable**

A table containing TCP connection-specific information.

tcpConnEntry

Information about a particular current TCP connection. An object of this type is transient, in that it ceases to exist when (or soon after) the connection makes the transition to the CLOSED state.

tcpConnState

The state of this TCP connection. Values may be any of the following:

Value	Description
1	closed
2	listen
3	synSent
4	synReceived
5	established
6	finWait1
7	finWait2
8	closeWait
9	lastAck
10	closing
11	timeWait

tcpConnLocalAddress

The local IP address for this TCP connection.

tcpConnLocalPort

The local port number for this TCP connection.

tcpConnRemAddress

The remote IP address for this TCP connection.

tcpConnRemPort

The remote port number for this TCP connection.

User Datagram Protocol (UDP) Group**udpInDatagrams**

The total number of UDP datagrams delivered to UDP users.

udpNoPorts

The total number of received UDP datagrams for which there is no application at the destination port.

udpInErrors

The number of received UDP datagrams that are not delivered for reasons other than the lack of an application at the destination port.

udpOutDatagrams

The total number of UDP datagrams sent from this entity.

UDP Listener Table**udpLocalAddress**

The local IP address for this UDP listener.

udpLocalPort

The local port number for this UDP listener.

Exterior Gateway Protocol (EGP) Group**egpInMsgs**

The number of EGP messages received without error.

egpInErrors

The number of EGP messages received that are in error.

egpOutMsgs

The total number of locally generated EGP messages.

egpOutErrors

The number of locally EGP messages not sent due to resource limitations within an EGP entity.

egpAs

The autonomous system number of the EGP entity.

EGP Neighbor Table**egpNeighTable**

The EGP neighbor table.

egpNeighEntry

Information about this entity's relationship with a particular EGP neighbor.

egpNeighState

The EGP state of the local system with respect to this entry's EGP neighbor. Entries may be as follows:

Value	Description
1	idle
2	acquisition
3	down
4	up
5	cease

egpNeighAddr

The IP address of this entry's EGP neighbor.

egpNeighAs

The autonomous system of this EGP peer. Zero is specified if the autonomous system number of the neighbor is not yet known.

egpNeighInMsgs

The number of EGP messages received without error from this EGP peer.

egpNeighInErrs

The number of EGP messages received from this EGP peer that are in error. For example, bad EGP checksum.

egpNeighOutMsgs

The number of locally generated EGP messages to this EGP peer.

egpNeighOutErrs

The number of locally generated EGP messages not sent to this EGP peer due to resource limitations within an EGP entity.

egpNeighInErrMsgs

The number of EGP-defined error messages received from this EGP peer.

egpNeighOutErrMsgs

The number of EGP-defined error messages sent to this EGP peer.

egpNeighStateUps

The number of EGP state transitions to the UP state with the EGP peer.

egpNeighStateDowns

The number of EGP state transitions from the UP state to any other state with this EGP peer.

egpNeighIntervalHello

The interval between EGP Hello command retransmissions.

egpNeighIntervalPoll

The interval between EGP poll command retransmissions.

egpNeighMode

The polling mode of this EGP entity. Values may be the following:

- 1 active
- 2 passive

egpNeighEventTrigger

A control variable used to trigger operator-initiated Start and Stop events. The values are as follows:

- 1 start
- 2 stop

Simple Network Management Protocol (SNMP) Group

Implementation of the SNMP group is mandatory for all systems that support an SNMP protocol entity.

snmplnBadCommunityNames

The total number of SNMP PDUs delivered to the SNMP protocol entity that use an SNMP community name not known to said entity.

snmplnBadCommunityUses

The total number of SNMP PDUs delivered to the SNMP protocol entity that represented an SNMP operation not allowed by the SNMP community named in the PDU.

snmplnASNParseErrs

The total number of ASN.1 parsing errors (either in encoding or syntax) encountered by the SNMP protocol entity when decoding received SNMP PDUs.

snmplnBadVersions

The total number of syntactically correct SNMP PDUs delivered to the SNMP protocol entity but intended for an unsupported SNMP version.

snmplnPkts

The total number of PDUs delivered to the SNMP entity from the transport service.

snmplnReadOnlys

The total number of valid SNMP PDUs delivered to the SNMP protocol entity for which the value of the ErrorStatus component is readOnly.

snmplnBadValues

The total number of valid SNMP PDUs delivered to the SNMP protocol entity for which the value of the ErrorStatus component is badValue.

snmplnGetNexts

The total number of SNMP Get-Next PDUs accepted and processed by the SNMP protocol entity.

snmplnGetRequests

The total number of SNMP Get-Request PDUs accepted and processed by the SNMP protocol entity.

snmplnSetRequests

The total number of SNMP Set-Request PDUs accepted and processed by the SNMP protocol entity.

snmplnBadTypes

The total number of SNMP PDUs delivered to the SNMP protocol entity with an unknown PDU type.

snmpOutPkts

The total number of SNMP PDUs passed from the SNMP protocol entity to the transport service.

snmpOutReadOnlys

The total number of valid SNMP PDUs generated by the SNMP protocol entity for which the value of the ErrorStatus component is readOnly.

snmpOutBadValues

The total number of valid SNMP PDUs generated by the SNMP protocol entity for which the value of the `ErrorStatus` component is `badValue`.

snmpOutGetNexts

The total number of SNMP Get-Next PDUs generated by the SNMP protocol entity.

snmpOutGetRequests

The total number of SNMP Get-Request PDUs generated by the SNMP protocol entity.

snmpOutSetRequests

The total number of SNMP Set-Request PDUs generated by the SNMP protocol entity.

snmpInGetResponses

The total number of SNMP Get-Response PDUs accepted and processed by the SNMP protocol entity.

snmpInNoSuchNames

The total number of valid SNMP PDUs delivered to the SNMP protocol entity for which the value of the `ErrorStatus` component is `noSuchName`.

snmpInTooBig

The total number of valid SNMP PDUs delivered to the SNMP protocol entity for which the value of the `ErrorStatus` component is `tooBig`.

snmpInTraps

The total number of SNMP Trap PDUs accepted and processed by the SNMP protocol entity.

snmpInGenErrs

The total number of valid SNMP PDUs delivered to the SNMP protocol entity for which the value of the `ErrorStatus` component is `genErr`.

snmpInTotalSetVars

The total number of MIB objects retrieved successfully by the SNMP protocol entity as the result of receiving valid SNMP Set-Request PDUs.

snmpEnableAuthTraps

Indicates whether the SNMP agent process is configured to generate authentication-failure traps.

snmpOutGetResponses

The total number of SNMP Get-Response PDUs generated by the SNMP protocol entity.

snmpOutNoSuchNames

The total number of valid SNMP PDUs generated by the SNMP protocol entity for which the value of the ErrorStatus component is noSuchName.

snmpOutTooBig

The total number of valid SNMP PDUs generated by the SNMP protocol entity for which the value of the ErrorStatus component is tooBig.

snmpOutTraps

The total number of SNMP Trap PDUs generated by the SNMP protocol entity.

snmpOutGenErrs

The total number of valid SNMP PDUs generated by the SNMP protocol entity for which the value of the ErrorStatus component is genErr.

snmpInTotalReqVars

The total number of MIB objects retrieved successfully by the SNMP protocol entity as the result of receiving valid SNMP Get-Request and Get-Next PDUs.

Bridge MIB

This section contains a description for each of the MIB objects within the following Bridge MIB Groups:

- dot1dBase
- dot1dStp
- dot1dSr
- dot1dTp
- dot1dStatic

dot1dBase Group

Implementation of the dot1dBase group is mandatory for all bridges.

dot1dBaseBridgeAddress

The MAC address used by this bridge when it must be referred to in a unique fashion. It is recommended that this be the numerically smallest MAC address of all ports that belong to this bridge. However, it is only required to be unique. When concatenated with dot1dStpPriority, a unique BridgeIdentifier is formed which is used in the Spanning Tree Protocol.

dot1dBaseNumPorts

The number of ports controlled by this bridging entity.

dot1dBaseType

Indicates what type of bridging this bridge can perform. If a bridge is actually performing a certain type of bridging this will be indicated by entries in the port table for the given type.

Generic Bridge Port Table

dot1dBasePortTable

A table that contains generic information about every port that is associated with this bridge. Transparent, source-route and srt ports are included.

dot1dBasePortEntry

A list of information for each port of the bridge.

dot1dBasePort

The port number of the port for which this entry contains bridge management information.

dot1dBasePortIfIndex

The value of the instance of the ifIndex object for the interface corresponding to this port.

dot1dBasePortCircuit

For a port which (potentially) has the same value of dot1dBasePortIfIndex as another port on the same bridge, this object contains the name of an object instance unique to this port. For example, in the case where multiple ports correspond one-to-one with multiple X.25 virtual circuits, this value might identify an (e.g., the first) object instance associated with the X.25 virtual circuit corresponding to this port. For a port which has a unique value of dot1dBasePortIfIndex, this object can have the value { 0 0 }.

dot1dBasePortDelayExceededDiscards

The number of frames discarded by this port due to excessive transit delay through the bridge. It is incremented by both transparent and source route bridges.

dot1dBasePortMtuExceededDiscards

The number of frames discarded by this port due to an excessive size. It is incremented by both transparent and source route bridges.

dot1dStp Group

Implementation of the dot1dStp group is optional. It is implemented by those bridges that support the Spanning Tree Protocol. Transparent, Source Route and SRT bridges will implement this group only if they support the Spanning Tree Protocol.

dot1dStpProtocolSpecification

An indication of what version of the Spanning Tree Protocol is being run. The value decLb100 (2) indicates the DEC LANbridge 100 Spanning Tree protocol. IEEE 802.1d implementations will return ieee8021d (3). If future versions of the IEEE Spanning Tree Protocol are released that are incompatible with the current version, a new value will be defined.

dot1dStpPriority

The value of the writable portion of the Bridge ID, i.e., the first two octets of the (8 octet long) Bridge ID. The other (last) 6 octets of the Bridge ID are given by the value of dot1dBaseBridgeAddress.

dot1dStpTimeSinceTopologyChange

The time (in hundredths of a second) since the last time a topology change was detected by the bridge entity.

dot1dStpTopChanges

The total number of topology changes detected by this bridge since the management entity was last reset or initialized.

dot1dStpDesignatedRoot

The bridge identifier of the root of the spanning tree as determined by the Spanning Tree Protocol as executed by this node. This value is used as the Root Identifier parameter in all Configuration Bridge PDUs originated by this node.

dot1dStpRootCost

The cost of the path to the root as seen from this bridge.

dot1dStpRootPort

The port number of the port which offers the lowest cost path from this bridge to the root bridge.

dot1dStpMaxAge

The maximum age of Spanning Tree Protocol information learned from the network on any port before it is discarded, in units of hundredths of a second. This is the actual value that this bridge is currently using.

dot1dStpHelloTime

The amount of time between the transmission of configuration bridge PDUs by this node on any port when it is the root of the spanning tree or trying to become so, in units of hundredths of a second. This is the actual value that this bridge is currently using.

dot1dStpHoldTime

This time value determines the interval length during which no more than two configuration bridge PDUs shall be transmitted by this node, in units of hundredths of a second.

dot1dStpForwardDelay

This time value, measured in units of hundredths of a second, controls how fast a port changes its spanning state when moving towards the Forwarding state. The value determines how long the port stays in a particular state before moving to the next state. For example, how long a port stays in the Listening state when moving from Blocking to Learning. This value is also used, when a topology change has been detected and is under way, to age all dynamic entries in the Forwarding Database. [Note that this value is the one that this bridge is currently using, in contrast to dot1dStpBridgeForwardDelay which is the value that this bridge and all others would start using if/when this bridge were to become the root.]

dot1dStpBridgeMaxAge

The value that all bridges use for MaxAge when this bridge is acting as the root. Note that 802.1d/D9 specifies that the range for this parameter is related to the value of dot1dStpBridgeHelloTime. The granularity of this timer is specified by 802.1d/D9 to be 1 second. An agent may return a badValue error if a set is attempted to a value which is not a whole number of seconds.

dot1dStpBridgeHelloTime

The value that all bridges use for HelloTime when this bridge is acting as the root. The granularity of this timer is specified by 802.1d/D9 to be 1 second. An agent may return a badValue error if a set is attempted to a value which is not a whole number of seconds.

dot1dStpBridgeForwardDelay

The value that all bridges use for ForwardDelay when this bridge is acting as the root. Note that 802.1d/D9 specifies that the range for this parameter is related to the value of dot1dStpBridgeMaxAge. The granularity of this timer is specified by 802.1d/D9 to be 1 second. An agent may return a badValue error if a set is attempted to a value which is not a whole number of seconds.

Spanning Tree Port Table

dot1dStpPortTable

A table that contains port-specific information about the Spanning Tree Protocol.

dot1dStpPortEntry

A list of information maintained by every port about the Spanning Tree Protocol state for that port.

Dot1dStpPortEntry

The port number of the port where this entry contains Spanning Tree Protocol management information.

dot1dStpPortPriority

The value of the priority field that is contained in the first (in network byte order) octet of the (2 octet long) Port ID. The other octet of the Port ID is given by the value of dot1dStpPort.

dot1dStpPortState

The port's current state as defined by application of the Spanning Tree Protocol. This state controls what action a port takes on reception of a frame. If the bridge has detected a port that is malfunctioning, it will place that port into the broken(6) state. For ports which are disabled (see dot1dStpPortEnable), this object will have a value of disabled(1).

dot1dStpPortEnable

The enabled/disabled status of the port.

dot1dStpPortPathCost

The contribution of this port to the path cost of paths towards the spanning tree root which include this port.

dot1dStpPortDesignatedRoot

The unique bridge identifier of the bridge recorded as the Root in the Configuration BPDUs transmitted by the Designated Bridge for the segment to which the port is attached.

dot1dStpPortDesignatedCost

The path cost of the Designated Port of the segment connected to this port. This value is compared to the Root Path Cost field in received bridge PDUs.

dot1dStpPortDesignatedBridge

The Bridge Identifier of the bridge which this port considers to be the Designated Bridge for this port's segment.

dot1dStpPortDesignatedPort

The Port Identifier of the port on the Designated Bridge for this port's segment.

dot1dStpPortForwardTransitions

The number of times this port has transitioned from the Learning state to the Forwarding state.

dot1dSr Group

Implementation of the dot1dSr group is optional. It is implemented by those bridges that support the source route bridging mode, including Source Route and SRT bridges.

dot1dSrPortTable

A table that contains information about every port that is associated with this source route bridge.

dot1dSrPortEntry

A list of information for each port of a source route bridge.

dot1dSrPort

The port number of the port where this entry contains Source Route management information.

dot1dSrPort

The maximum number of routing descriptors allowed in an All Paths or Spanning Tree Explorer frames.

dot1dSrPortHopCount

The maximum number of routing descriptors allowed in an All Paths or Spanning Tree Explorer frames.

dot1dSrPortLocalSegment

The segment number that uniquely identifies the segment to which this port is connected. Current source routing protocols limit this value to the range: 0 through 4095. A value of 65535 signifies that no segment number is assigned to this port.

dot1dSrPortBridgeNum

A bridge number uniquely identifies a bridge when more than one bridge is used to span the same two segments. Current source routing protocols limit this value to the range: 0 through 15. A value of 65535 signifies that no bridge number is assigned to this bridge.

dot1dSrPortTargetSegment

The segment number that corresponds to the target segment this port is considered to be connected to by the bridge. Current source routing protocols limit this value to the range: 0 through 4095. A value of 65535 signifies that no target segment is assigned to this port.

dot1dSrPortLargestFrame

The maximum size of the INFO field (LLC and above) that this port can send/receive. It does not include any MAC level (framing) octets. The value of this object is used by this bridge to determine whether a modification of the LargestFrame field of the Routing Control field of the Routing Information Field is necessary. Valid values as defined by the 802.5 source routing bridging specification are 516, 1500, 2052, 4472, 8144, 11407, 17800 and 65535 octets. Behavior of the port when an illegal value is written is implementation specific. It is recommended that a reasonable legal value be chosen.

dot1dSrPortSTESpanMode

Determines how this port behaves when presented with a Spanning Tree Explorer frame. The values are as follows:

Value	Definition
auto-span (1)	Can only be returned by a bridge that both implements the Spanning Tree Protocol and has use of the protocol enabled on this port. The behavior of the port for Spanning Tree Explorer frames is determined by the state of dot1dStpPortState. If the port is in the forwarding state, the frame will be accepted or propagated. Otherwise, it will be silently discarded.
disabled (2)	Indicates that the port will not accept or send Spanning Tree Explorer packets; any STE packets received will be silently discarded.
forced (3)	Indicates the port will always accept and propagate Spanning Tree Explorer frames. This allows a manually configured Spanning Tree for this class of packet to be configured. Note that unlike transparent bridging this is not catastrophic to the network if there are loops.

dot1dSrPortSpecInFrames

The number of specifically routed frames that have been received from this port's segment.

dot1dSrPortSpecOutFrames

The number of specifically routed frames that this port has transmitted on its segment.

dot1dSrPortApelnFrames

The number of all paths explorer frames that have been received by this port from its segment.

dot1dSrPortApeOutFrames

The number of all paths explorer frames that have been transmitted by this port on its segment.

dot1dSrPortSteInFrames

The number of spanning tree explorer frames that have been received by this port from its segment.

dot1dSrPortSteOutFrames

The number of spanning tree explorer frames that have been transmitted by this port on its segment.

dot1dSrPortSegmentMismatchDiscards

The number of explorer frames that have been discarded by this port because the routing descriptor field contained an invalid adjacent segment value.

dot1dSrPortDuplicateSegmentDiscards

The number of frames that have been discarded by this port because the routing descriptor field contained a duplicate segment identifier.

dot1dSrPortHopCountExceededDiscards

The number of explorer frames that have been discarded by this port because the Routing Information Field has exceeded the maximum route descriptor length.

dot1dTp Group

Implementation of the dot1dTp group is optional. It is implemented by those bridges that support the transparent bridging mode. A transparent or SRT bridge will implement this group.

dot1dTpLearnedEntryDiscards

The total number of Forwarding Database entries, which have been or would have been learned, but have been discarded due to a lack of space to store them in the Forwarding Database. If this counter is increasing, it indicates that the Forwarding Database is regularly becoming full (a condition which has unpleasant performance effects on the subnetwork). If this counter has a significant value but is not presently increasing, it indicates that the problem has been occurring but is not persistent.

dot1dTpAgingTime

The timeout period in seconds for aging out dynamically learned forwarding information.

Forwarding Database for Transparent Bridges

dot1dTpFdbTable

A table that contains information about unicast entries for which the bridge has forwarding and/or filtering information. This information is used by the transparent bridging function in determining how to propagate a received frame.

dot1dTpFdbEntry

Information about a specific unicast MAC address for which the bridge has some forwarding and/or filtering information.

dot1dTpFdbAddress

Either the value 0 or the port number of the port on which a frame having a source address equal to the value of the corresponding instance of dot1dTpFdbAddress has been seen. A value of 0 indicates that the port number has not been learned but that the bridge does have some forwarding/filtering information about this address (e.g. in the dot1dStaticTable). Implementors are encouraged to assign the port value to this object whenever it is learned even for addresses for which the corresponding value of dot1dTpFdbStatus (3) is not learned.

dot1dTpFdbStatus

The status of this entry. The meanings of the values are listed in the following table:

Value	Definition
other (1)	None of the following. This would include the case where some other MIB object (not the corresponding instance of dot1dTpFdbPort, nor an entry in the dot1dStaticTable) is being used to determine if and how frames addressed to the value of the corresponding instance of dot1dTpFdbAddress are being forwarded.
invalid (2)	This entry is no longer valid (e.g., it was learned but has since aged-out), but has not yet been flushed from the table.
learned (3)	The value of the corresponding instance of dot1dTpFdbPort was learned, and is being used.
self (4)	The value of the corresponding instance of dot1dTpFdbAddress represents one of the bridge's addresses. The corresponding instance of dot1dTpFdbPort indicates which of the bridge's ports has this address.
mgmt (5)	The value of the corresponding instance of dot1dTpFdbAddress is also the value of an existing instance of dot1dStaticAddress.

Port Table for Transparent Bridges

dot1dTpPortTable

A table that contains information about every port that is associated with this transparent bridge.

dot1dTpPortEntry

A list of information for each port of a transparent bridge.

dot1dTpPort

The port number of the port for which this entry contains Transparent bridging management information.

dot1dTpPortMaxInfo

The maximum size of the INFO (non-MAC) field that this port will receive or transmit.

dot1dTpPortInFrames

The number of frames that have been received by this port from its segment. Note that a frame received on the interface corresponding to this port is only counted by this object if, and only if, it is for a protocol being processed by the local bridging function.

dot1dTpPortOutFrames

The number of frames that have been transmitted by this port to its segment. Note that a frame transmitted on the interface corresponding to this port is only counted by this object if, and only if, it is for a protocol being processed by the local bridging function.

dot1dTpPortInDiscards

Count of valid frames received which were discarded (i.e., filtered) by the Forwarding Process.

Static (Destination-Address Filtering) Database:

Implementation of this group is optional.

dot1dStaticTable

A table containing filtering information configured into the bridge by (local or network) management specifying the set of ports to which frames received from specific ports and containing specific destination addresses are allowed to be forwarded. The value of zero in this table as the port number from which frames with a specific destination address are received, is used to specify all ports for which there is no specific entry in this table for that particular destination address. Entries are valid for unicast and for group/broadcast addresses.

dot1dStaticEntry

Filtering information configured into the bridge by (local or network) management specifying the set of ports to which frames received from a specific port and containing a specific destination address are allowed to be forwarded.

dot1dStaticAddress

The destination MAC address in a frame to which this entry's filtering information applies. This object can take the value of a unicast address, a group address or the broadcast address.

dot1dStaticReceivePort

Either the value 0 learned or the port number of the port from which a frame must be received in order for this entry's filtering information to apply. A value of 0 indicates that this entry applies on all ports of the bridge for which there is no other applicable entry.

dot1dStaticAllowedToGoTo

The set of ports to which frames received from a specific port and destined for a specific MAC address are allowed to be forwarded. Each octet within the value of this object specifies a set of eight ports, with the first octet specifying ports 1 through 8, the second octet specifying ports 9 through 16, etc. Within each octet, the most significant bit represents the lowest numbered port, and the least significant bit represents the highest numbered port. Thus, each port of the bridge is represented by a single bit within the value of this object. If that bit has a value of 1 then that port is included in the set of ports; the port is not included if its bit has a value of 0. (Note that the setting of the bit corresponding to the port from which a frame is received is irrelevant.)

dot1dStaticStatus

This object indicates the status of this entry.

Status	Description
other (1)	This entry is currently in use but the conditions under which it will remain so are different from each of the following values.
invalid (2)	Writing this value to the object removes the corresponding entry.
permanent (3)	This entry is currently in use and will remain so after the next reset of the bridge.
deleteOnReset (4)	This entry is currently in use and will remain so until the next reset of the bridge.
deleteOnTimeout (5)	This entry is currently in use and will remain so until it is aged out.

Char-like MIB

This section contains a description of the MIB objects within the generic Character group.

Implementation of this group is mandatory for all systems that offer character ports.

Generic Character Group

charNumber

The number of entries in charPortTable, regardless of their current state.

Character Port Table

charPortTable

A list of port entries. The number of entries is given by the value of charNumber.

charPortEntry

Status and parameter values for a character port.

charPortIndex

A unique value for each character port. Its value ranges between 1 and the value of charNumber. By convention and if possible, hardware port numbers come first, with a simple, direct mapping. The value for each port must remain constant at least from one re-initialization of the network management agent to the next.

charPortName

An administratively assigned name for the port, typically with some local significance.

charPortType

The port's type:

Port Type	Definition
physical	If the port represents an external hardware connector.
virtual	If the port does not represent an external hardware connector.

charPortHardware

A reference to hardware MIB definitions specific to a physical port's external connector. For example, if the connector is RS-232, then the value of this object

refers to a MIB sub-tree defining objects specific to RS-232. If an agent is not configured to have such values, the agent returns the object identifier.

charPortReset

A control to force the port into a clean, initial state, both hardware and software, disconnecting all the port's existing sessions. In response to a get-request or get-next-request, the agent always returns ready as the value. Setting the value to execute causes a reset.

charPortAdminStatus

The port's desired state, independent of flow control:

Port State	Indicates . . .
enabled	That the port is allowed to pass characters and form new sessions.
disabled	That the port is allowed to pass characters but not form new sessions.
off	That the port is not allowed to pass characters or have any sessions.
maintenance	A maintenance mode, exclusive of normal operation, such as running a test.

charPortOperStatus

The port's actual, operational state, independent of flow control:

Port State	Indicates . . .
up	Able to function normally.
down	Inability to function for administrative or operational reasons.
maintenance	A maintenance mode, exclusive of normal operation, such as running a test.
absent	That port hardware is not present.
active	The port is up, with a user present.

charPortLastChange

The value of sysUpTime at the time the port entered its current operational state. the current state was entered prior to the last reinitialization of the local network management subsystem, then this object contains a zero value.

charPortInFlowType

The port's type of input flow control:

Control Type	Indicates
none	No flow control at this level or below.
xonXoff	Software flow control by recognizing XON and XOFF characters.
hardware	Flow control delegated to the lower level, for example a parallel port.

ctsRts and dsrDtr are specific to RS-232-like ports. Although not architecturally pure, they are included here for simplicity's sake.

charPortOutFlowType

The port's type of output flow control.

Control Type	Indicates...
none	No flow control at this level or below.
xonXoff	Software flow control by recognizing XON and XOFF characters.
hardware	Flow control delegated to the lower level, for example a parallel port.

ctsRts and dsrDtr are specific to RS-232-like ports. Although not architecturally pure, they are included here for simplicity's sake.

charPortInFlowState

The current operational state of input flow control on the port.

Control Type	Indicates . . .
none	Not applicable.
unknown	This level does not know.
stop	Flow not allowed.
go	Flow allowed.

charPortOutFlowState

The current operational state of output flow control on the port.

Control Type	Indicates . . .
none	Not applicable.
unknown	This level does not know.
stop	Flow not allowed.
go	Flow allowed.

charPortInCharacters

Total number of characters detected as input from the port since system re-initialization and while the port operational state was up, active or maintenance, including, for example, framing, flow control (i.e. XON and XOFF), each occurrence of a BREAK condition, locally-processed input and input sent to all sessions.

charPortOutCharacters

Total number of characters detected as output to the port since system re-initialization and while the port operational state was up, active or maintenance, including, for example, framing, flow control (i.e. XON and XOFF), each occurrence of a BREAK condition, locally-created output, and output received from all sessions.

charPortAdminOrigin

The administratively allowed origin for establishing session on the port. Dynamic allows network or local session establishment. None disallows session establishment.

charPortSessionMaximum

The maximum number of concurrent sessions allowed on the port. A value of -1 indicates no maximum. Setting the maximum to less than the current number of sessions has unspecified results.

charPortSessionNumber

The number of open sessions on the port that are in the connecting, connected, or disconnecting state.

charPortSessionIndex

The value of charSessIndex for the port's first or only active session. If the port has no active session, the agent returns the value zero.

Character Session Table

charSessTable

A list of port session entries.

charSessEntry

Status and parameter values for a character port session.

chharSessPortIndex

The value of charPortIndex for the port to which this session belongs.

charSessIndex

The session index in the context of the port, a non-zero positive integer. Session indexes within a port need not be sequential. Session indexes may be reused for different ports. For example, port 1 and port 3 may both have a session 2 at the same time. Session indexes may have any valid integer value, with any meaning convenient to the agent implementation.

charSessKill

A control to terminate the session. In response to a get-request or get-next-request, the agent always returns ready as the value. Setting the value to execute causes termination.

charSessState

The current operational state of the session, disregarding flow control:

Control Type	Indicates . . .
connected	Character data could flow on the network side of session
connecting	Moving from nonexistent toward connected.
disconnecting	Moving from connected or connecting to nonexistent.

charSessProtocol

The network protocol over which the session is running. Other OBJECT IDENTIFIER values may be defined elsewhere, in association with specific protocols. However, this document assigns those of known interest as of this writing.

charSessOperOrigin

The session's source of establishment.

charSessInCharacters

This session's subset of charPortInCharacters.

charSessOutCharacters

This session's subset of charPortOutCharacters.

charSessConnectionId

A reference to additional local MIB information. This should be the highest available related MIB, corresponding to charSessProtocol, such as Telnet. For example, the value for a TCP connection (in the absence of a Telnet MIB) is the object identifier of tcpConnState. If an agent is not configured to have such values, the agent returns the object identifier.

charSessStartTime

The value of sysUpTime in MIBII when the session entered connecting state.

RS232-like MIB

This section contains a description of the MIB objects for the RS232-like hardware devices.

Implementation of the generic RS232-like group is mandatory for all systems that have RS232-like hardware ports supporting higher level services such as character streams or network interfaces.

rs232Number

The number of ports (regardless of their current state) in the RS-232-like general port table.

rs232PortTable

A list of port entries. The number of entries is given by the value of rs232Number.

rs232PortEntry

Status and parameter values for a port.

rs232PortIndex

A unique value for each port. Its value ranges between 1 and the value of rs232Number. By convention, and if possible, hardware port numbers map directly to external connectors. The value for each port must remain constant at least from one re-initialization of the network management agent to the next.

rs232PortType

The port's hardware type.

rs232PortInSigNumber

The number of input signals for the port in the input signal table (rs232PortInSigTable). The table contains entries only for those signals the software can detect.

rs232PortOutSigNumber

The number of output signals for the port in the output signal table (rs232PortOutSigTable). The table contains entries only for those signals the software can assert.

rs232PortInSpeed

The port's input speed in bits per second.

rs232PortOutSpeed

The port's output speed in bits per second.

RS-232-like Asynchronous Port Group

Implementation of this group is mandatory if the system has any asynchronous ports. Otherwise, it is not present.

rs232AsyncPortTable

A list of asynchronous port entries. The maximum entry number is given by the value of rs232Number.

rs232AsyncPortEntry

Status and parameter values for an asynchronous port.

rs232AsyncPortIndex

A unique value for each port. Its value is the same as rs232PortIndex for the port.

rs232AsyncPortBits

The port's number of bits in a character.

rs232AsyncPortStopBits

The port's number of stop bits.

rs232AsyncPortParity

The port's sense of a character parity bit.

rs232AsyncPortAutobaud

A control for the port's ability to automatically sense input speed. When rs232PortAutoBaud is enabled, a port may autobaud to values different from the set values for speed, parity and character size. As a result a network management system may temporarily observe values different from what was previously set.

rs232AsyncPortParityErrs

Total number of characters with a parity error, input from the port since system re-initialization and while the port state was up or test.

rs232AsyncPortFramingErrs

Total number of characters with a framing error, input from the port since system re-initialization and while the port state was up or test.

rs232AsyncPortOverrunErrs

Total number of characters with an overrun error, input from the port since system re-initialization and while the port state was up or test.

RS-232-like Synchronous Port Group

Implementation of this group is mandatory if the system has any synchronous ports. Otherwise it is not present.

rs232SyncPortTable

A list of synchronous port entries. The maximum entry number is given by the value of rs232Number. Entries need not exist for asynchronous ports.

rs232SyncPortEntry

Status and parameter values for a synchronous port.

rs232SyncPortIndex

A unique value for each port. Its value is the same as rs232PortIndex for the port.

rs232SyncPortClockSource

Source of the port's bit rate clock. Split means the transmit clock is internal and the receive clock is external.

s232SyncPortFrameCheckErrs

Total number of frames with an invalid frame check sequence, input from the port since system re-initialization and while the port state was up or test.

rs232SyncPortTransmitUnderrunErrs

Total number of frames that failed to be transmitted on the port since system re-initialization and while the port state was up or test because data was not available to the transmitter in time.

rs232SyncPortReceiveOverrunErrs

Total number of frames that failed to be received on the port since system re-initialization and while the port state was up or test because the receiver did not accept the data in time.

rs232SyncPortInterruptedFrames

Total number of frames that failed to be received or transmitted on the port due to loss of modem signals since system re-initialization and while the port state was up or test.

rs232SyncPortAbortedFrames

Number of frames aborted on the port due to receiving an abort sequence since system re-initialization and while the port state was up or test.

Input Signal Table**rs232InSigTable**

A list of port input control signal entries.

rs232InSigEntry

Input control signal status for a hardware port.

rs232InSigPortIndex

The value of rs232PortIndex for the port to which this entry belongs.

rs232InSigName

Identification of a hardware signal, as follows:

Signal	Definition
rts	Request to Send
cts	Clear to Send
dsr	Data Set Ready
dtr	Data Terminal Ready
ri	Ring Indicator
dcd	Received Line Signal Detector
sq	Signal Quality Detector
srs	Data Signaling Rate Selector
srts	Secondary Request to Send
scts	Secondary Clear to Send
sdc	Secondary Received Line Signal Detector

rs232InSigState

The current signal state.

rs232InSigChanges

The number of times the signal has changed from on to off or from off to on.

Output Signal Table

rs232OutSigTable

A list of port output control signal entries.

rs232OutSigEntry

Output control signal status for a hardware port.

rs232OutSigPortIndex

The value of rs232PortIndex for the port to which this entry belongs.

rs232OutSigName

Identification of a hardware signal, as follows

Signal	Definition
rts	Request to Send
cts	Clear to Send
dsr	Data Set Ready
dtr	Data Terminal Ready
ri	Ring Indicator
dcd	Received Line Signal Detector
sq	Signal Quality Detector
srs	Data Signaling Rate Selector
srts	Secondary Request to Send
scts	Secondary Clear to Send
sdc	Secondary Received Line Signal Detector

rs232OutSigState

The current signal state.

rs232OutSigChanges

The number of times the signal has changed from on to off or from off to on.

DH_90 MIB

This section contains a description of the MIB objects within the DH_90 group.

- dh90 Group
- da90 Group
- ds90L Group
- drpt90 Group
- dbSysChar Group
- dbSysStatus Group
- dbIfTable Group
- db90Char Group
- db90Stat Group
- db90Coun Group
- db90Span Group

Implementation of this group is mandatory for all systems that operate with the DEChub 90.

dh90 Group

The dh90 variables contain information used to manage each DEChub 90. A DEChub 90 can contain multiple devices such as bridges, terminal servers and repeaters. The DECagent 90 is a module which provides proxy SNMP management for each of these devices. A single DECagent can manage multiple hubs, in which case each hub is a separate community identified by unique community strings. The dh90 group then provides management information not only for each hub, but also for the communities used to identify each hub.

The DECagent 90 can also manage individual bridge and terminal server modules which are not resident in a hub. In this case, each bridge or server is its own community with its own community strings and is a community of type standAloneCommunity.

Note also that two DEChub 90 hubs can be connected and managed as a single 16-slot hub. The dh90Type value for both is dechub90, and either 8 or 16 is set as the value for dh90NumberSlots. When a community is created, dh90NumberSlots has a value of 1.

dh90Type

Identifies the type of hub defined and identifies standalone configurations managed by DECagent 90 modules. Setting this variable to invalid(2) deletes the community associated with the community string and releases the RAM and flash memory resources used by the community.

dh90Backplane

Media on backplane of hub.

dh90LastChange

The value of the sysUpTime at the time this hub added or removed a module.

dh90NumberSlots

Number of slots in backplane. Setting this variable to values of 1, 8 or 16 defines the size of a hub in a particular community.

dh90SlotTable

A list of modules installed in the DEChub 90. To create a new row (module) in the slot table, set a EMPTY dh90SlotModuleType to the appropriate value. If adding a bridge or server, next set the dh90SlotPhysicalAddress for the row. Sets of the other variables in the row must be done in separate PDUs AFTER the MAC address of the device has been set correctly. When sweeping the slot table with getNext, rows of the table which are empty will be skipped. The only values returned are for slots which are occupied. To delete a module, set the dh90SlotModuleType (instanced by slot number) to empty(2). The row corresponding to the slot number will be deleted from the table.

dh90SlotEntry

Contains objects defining characteristics of modules in the DEChub 90.

Entry	Type
dh90SlotIndex	INTEGER
dh90SlotModuleType	INTEGER
dh90SlotModuleName	DisplayString
dh90SlotModuleVersion	DisplayString
dh90SlotCounterTime	Gauge
dh90SlotIfBase	INTEGER
dh90SlotIfNumber	INTEGER
dh90SlotPhysicalAddress	PhysAddress
dh90SlotNumberOfPorts	INTEGER
dh90SlotPassword	DisplayString
dh90SlotNewPassword	DisplayString

Entry	Type
dh90SlotPolling	INTEGER
dh90SlotPrimarySpecific	OBJECT IDENTIFIER
dh90SlotSecondarySpecific	OBJECT IDENTIFIER
dh90SlotIpAddress	IpAddress
dh90SlotCommunityString	ProxyString

dh90SlotIndex

The slot number containing the module.

dh90SlotModuleType

The type of the module in a slot. The following table contains the integers that correspond with the modules.

Num.	Module	Num.	Module
1	unknown	14	DECbridge90FL
2	empty	15	decwanrouter90
3	DECserver90L	16	DECrmon90
4	DECbridge90	17	DECrmon90plus
5	DECrepeater90C	18	DECserver90Lplus2
6	DECrepeater90T	19	DECserver90M
7	DECagent90	20	DECbrouter90T2A
8	DECserver90Lplus	21	DECwirecenter900-01
9	DECserver90TL	22	DECrepeater900-02
10	DECbrouter90T1	23	DECrepeater900-xx
11	DECbrouter90T2	24	DECagent900-06
12	DECrepeater90FL	25	DECcontroller900-07
13	DECrepeater90FA	26	DECrepeater900-08

dh90SlotModuleName

A textual description of the module.

dh90SlotModuleVersion

A textual description of the version level of the hardware and firmware of the module.

dh90SlotCounterTime

Time since the counters of this module were zeroed.

dh90SlotIfBase

An index into the ifTable for the first interface on this module. Add one to this value to reference the ifTable entry which corresponds to the second interface, etc. If no interface(s) is/are supported on the module, a value of zero is returned.

dh90SlotIfNumber

Number of interfaces on this module.

dh90SlotPhysicalAddress

A read-write variable for the physical address to allow devices to be added to the hub remotely. Writing this variable as part of a remote add device routine will cause the new physical address to be accessible in the corresponding ifPhysAddress variable in ifTable.

dh90SlotNumberOfPorts

The number of ports on this module. Returns 8 for DECserver90L and DECpeater90T, 6 for DECpeater90C.

dh90SlotPassword

Password used to enable privileged operations on the DECserver90L and access to the bridge. Reads to this variable will return a null string.

dh90SlotNewPassword

Setting this variable causes the agent to set a new password on the terminal server using the TSNewPassword value as the new password and TSPassword as the old password. Power on the terminal server must be cycled within a minute for the change to take effect. Reads to this variable will return a null string.

dh90SlotPolling

The current polling status of the module. If the DECagent 90 is polling this device, a get of this variable will return on, otherwise off is returned. Setting this variable to off causes the agent to stop polling the device in this slot. Setting this variable to on causes the agent to start polling the device in the specified slot. If a device does not respond to a poll, an moduleDown enterprise specific trap is generated.

dh90SlotPrimarySpecific

A reference to the Bridge MIB the Character-like MIB or the { drpt90 } repeater portion of this MIB definition. If the module realizes some other function, the object identifier nullSpecific OBJECT IDENTIFIER ::= { 0 0 } is returned.

dh90SlotSecondarySpecific

A reference to the RS232-like MIB definition, if dh90ModulePrimarySpecific identifies the Character-like MIB. Otherwise, the object identifier nullSpecific OBJECT IDENTIFIER ::= { 0 0 } is returned.

dh90SlotIpAddress

The IP address of the device in the given slot.

dh90SlotCommunityString

The community string of the device in the given slot.

dh90TrapAddressTable

A table of IP Addresses to which this device will send traps.

dh90TrapEntry

Each entry contains an IP Address to which all SNMP traps will be sent by this community.

dh90TrapAddress

An IP Address to which all SNMP traps generated by this device will be sent. Setting an instance to a value of zero will delete the row in the table.

da90 Group

The da90 group variables are used to manage the DECagent 90 module. The group exists once in the community which contains the proxy agent.

The community table is the basis of the proxy mechanism in the DECagent 90. A community is added to the agent by adding a row to the community table. Rows can be added from the console, or by way of sets to agent. To create a row, a set pdu must contain a set of active(3) to da90CommunityType and sets of strings to da90CommunityROString and da90CommunityRWString. A null value is allowed for one of the community strings, all non-null values must be unique among all the communities proxied by the agent. The hub corresponding to the community is created by a set to the dh90Type variable using the read-write community string defined above. A get-next sweep of the table will return those entries which are active.

To delete a community, first set the ModuleID for each slot in the community to empty, then set da90CommunityType to invalid(2).

da90FlashErasures

The number of times the flash memory on this DECagent 90 has been erased. This number is the total of erasures for all segments of flash.

da90Maintenance

A control variable to perform reset functions on a DECagent 90. In response to a get-request or a get-next-request, the agent returns ready(1) if SNMP sets are enabled, or setsDisabled(2) if they are disabled. Setting this value to setsDisabled(2) causes SNMP Sets to be disabled. Note, however, that you cannot re-enable SNMP Sets by way of this mechanism. SNMP sets can only be re-enabled by the console interface, or by restoring factory settings. Setting this value to reset(3) causes the entire module to be reset. Setting this value to resetToFactory(4) causes the entire device to be reset to the original factory settings. Setting either reset(3) and resetToFactory(4) results in an immediate reset with no response PDU being issued.

da90CommunityNumber

Number of entries in the da90CommunityTable.

da90CommunityTable

A table of community strings for client hubs.

da90CommunityEntry

Contains community type and strings, including the read-only profile, the read-write profile and a trap string.

Entry	Type
da90CommunityIndex	INTEGER
da90CommunityType	INTEGER
da90CommunityROString	ProxyString
da90CommunityRWString	ProxyString
da90CommunityTrapString	ProxyString

da90CommunityIndex

Index into hub community string pair table, identifies the community being accessed.

da90CommunityType

Identifies whether the current community information in the DENMA is active for this row in the community table. Setting to invalid frees the NVRAM used to store the community information (destroys the slot table and port tables for this community).

da90CommunityROString

Community string, with a read-only profile, used to access a client Hub. All community strings in the read-only set must be unique. Setting such a community string to a pre-existing value returns a status of BadValue. Reading this variable with any community string other than the agent's community string will return a zero-length string.

da90CommunityRWString

Community string, with a read-write profile, used to access a client Hub. All community strings in the read-only set must be unique. Setting such a community string to a pre-existing value returns a status of BadValue. Reading this variable with any community string other than the agent's read-write community string will return a zero length string.

da90CommunityTrapString

Community string sent with traps generated due to events caused by this community. The trap community strings do not have to be unique. This variable is read using the agent's community string.

ds90L Group

Each row of the ds90LModuleTable contains information on an individual DECserver 90L module. There is one row in the table for each DECserver 90L in the hub/community.

ds90LNumberModules

Number of DECserver 90L Modules represented in the ds90LModuleTable.

ds90LModuleTable

A table of DECserver 90L modules installed in the group. The number of entries is given in ds90LModuleNumber.

ds90LModuleEntry

Contains objects defining characteristics of DECserver 90L modules in the group.

ds90LSlotIndex	INTEGER
ds90LDot3StatsDeferredTransmissions	Counter
ds90LDot3StatsSingleCollisionFrames	Counter
ds90LDot3StatsMultipleCollisionFrames	Counter
ds90LEnetDataOverruns	Counter
ds90LLatCircMsgsIns	Counter
ds90LLatCircMsgsOuts	Counter
ds90LLatSessSolicitAccepts	Counter
ds90LLatSessSolicitRejects	Counter
ds90LLatCircDuplsMsgs	Counter
ds90LLatCircMsgRetransmits	Counter
ds90LLatSessIllegalSlots	Counter
ds90LIllegalMulticastRcv	Counter
ds90LLatCircKeepAlive	INTEGER
ds90LLatCircRetransmitLimit	INTEGER
ds90LLatCircInvalidMsgs	Counter
ds90LAuthorizeMode	INTEGER
ds90LMaintenance	INTEGER
ds90LPrompt	DisplayString

ds90LSlotIndex

The slot index value which addresses the dh90SlotTable for this module.

ds90LDot3StatsDeferredTransmissions

A count of frames for which the first transmission attempt on a particular interface is delayed because the medium is busy.

The count represented by an instance of this object does not include frames involved in collisions.

ds90LDot3StatsSingleCollisionFrames

A count of successfully transmitted frames on a particular interface for which transmission is inhibited by exactly one collision.

If DS90L is implementing the MIBII if group, the following also applies:

A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts or ifOutNUcastPkts object and is not counted by the corresponding instance of the ds90LDot3StatsMultipleCollisionFrames object.

ds90LDot3StatsMultipleCollisionFrames

A count of successfully transmitted frames on a particular interface for which transmission is inhibited by more than one collision.

If DS90L is implementing the MIBII if group, the following also applies:

A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts or ifOutNUcastPkts object and is not counted by the corresponding instance of the ds90LDot3StatsSingleCollisionFrames object.

ds90LEnetDataOverruns

A count of the number of frames arriving at the DS90L lost due to data overrun since it last reinitialized or zeroed counters.

ds90LLatCircMsgsI

A count of the number of LAT circuit messages received by this DS90L since it last reinitialized or zeroed counters.

ds90LLatCircMsgsO

A count of the number of LAT circuit messages transmitted by this DS90L since it last reinitialized or zeroed counters.

ds90LLatSessSolicitAccepts

A count of the number of LAT Solicits accepted by this DS90L since it last reinitialized or zeroed counters.

ds90LLatSessSolicitRejects

A count of the number of LAT Solicits rejected by this DS90L since it last reinitialized or zeroed counters.

ds90LLatCircDuplsMsgs

A count of the number of duplicate LAT messages the DS90L LAT implementation has discarded since it last reinitialized or zeroed counters.

ds90LLatCircMsgRetransmits

A count of the total number of LAT circuit messages this DS90L has retransmitted since it last reinitialized or zeroed counters.

ds90LLatSessIllegalSlots

A count of the number of illegal LAT slots the DS90L has received on all LAT sessions since it last reinitialized or zeroed counters.

ds90LIllegalMulticastRcv

This counts the number of illegal multicast LAT messages received on this DECserver 90L since it last reinitialized or zeroed counters.

ds90LLatCircKeepAlive

This defines the number of seconds the LAT circuits provider waits between transmitting LAT circuit layer keep-alive messages. Writing to this object is equivalent to issuing the CHANGE SERVER KEEPALIVE TIMER command through the user interface.

ds90LLatCircRetransmitLimit

This object defines the maximum number of LAT circuit message retransmissions the server will attempt before declaring the circuit failed. Writing to this object is equivalent to issuing the CHANGE SERVER RETRANSMIT LIMIT command via the user interface.

ds90LLatCircInvalidMsgs

This counts the number of invalid LAT messages received on its entry's LAT circuit since the DECserver90L last reinitialized or zeroed counters.

ds90LAuthorizeMode

Setting this value to enable causes the SET A password command to be sent to the DECserver 90. The module will enter authorized mode if the power is cycled on the module within one minute. The password used will be the current value of dh90ModulePassword. Setting this variable to disable results in authorized mode being cleared.

ds90LMaintenance

A control variable to reset the counters on a DECserver 90L. In response to a get-request or a get-next-request, the agent always returns ready. Setting the value to zeroCounters causes the if, LAT and dh90SlotCounterTime counters to be cleared. Setting the value to reset causes the entire module to be reset, any users and circuits will be disconnected. Setting the value to resetToFactory causes the entire device to be reset to the original factory settings. Passwords, port names and printer speeds will be cleared.

ds90LPrompt

A DECserver 90L+ (PLUS) variable. Contains the prompt displayed when a session is local to the PLUS version of the terminal server. The default value is ->. This variable is not present if the server is a DECserver 90L.

ds90LPortTable

Port specific variables for the DECserver 90L and DECserver 90L+. Most of the management of the ports is accomplished through the CHAR-like and RS232-like MIBs.

ds90LPortTable

Table of port specific variables on the DECserver90L.

ds90LPortEntry

A single LAT session instance this DS90L currently provides.

Entry	Type
ds90LPortIndex	INTEGER
ds90LPortRemoteModification	INTEGER
ds90LPortType	INTEGER
ds90LPortBreak	INTEGER
ds90LPortTest	INTEGER
ds90LPortAutoConfigure	INTEGER
ds90LPortOnDemandLoading	INTEGER

ds90LPortIndex

The value of charPortIndex for this port.

ds90LPortRemoteModification

Enables and disables remote modification of the port by the host system.

ds90LPortType

Selects the type of device connected to the port.

ds90LPortBreak

Determines if the port responds to local or remote break.

ds90LPortTest

Setting this variable to enable places the port into loopback mode. Loopback mode can be used to verify that a port is receiving and sending characters.

ds90LPortAutoConfigure

A DECserver 90L+ (PLUS) variable. Enables or disables the Autoconfiguration of the port, the ability of the port to autobaud. Disabling autoconfiguration also causes the speed and characteristics of the port to be stored in nonvolatile memory on the server. The variable is not present if the server is a DECserver 90L.

ds90LPortOnDemandLoading

A DECserver 90L+ (PLUS) variable. Enables or disables on demand loading of fonts. The variable is not present if the server is a DECserver 90L.

ds90LSessionTable

Session specific variables for the DECserver 90L and DECserver 90L+. Most of the management of the ports is accomplished through the CHAR-like and RS232-like MIBs.

ds90LSessionTable

The list of all LAT sessions active on this DECserver90L.

ds90LSessionEntry

A single LAT session instance this DS90L currently provides.

Entry	Type
ds90LSessionPort	INTEGER
ds90LSessionIndex	INTEGER
ds90LSessionRemoteNode	DisplayString
ds90LSessionService	DisplayString
ds90LSessionRemotePortId	DisplayString

ds90LSessionPort

The value of charPortIndex for the port to which this session belongs.

ds90LSessionIndex

The session index in the context of the port, a non-zero positive integer. Will be 1 for the DECserver90L, 4 for DECserver 90L+.

ds90LSessionRemoteNode

The name of the remote node providing the service for this session.

ds90LSessionService

The service to which this session is connected.

ds90LSessionRemotePortId

The name of the port at the remote node providing the connection, for example, LAT121.

drpt90 Group

These are the repeater port objects for the DECRepeater 90T and 90C.

drpt90PortTable

These are interface objects implemented in the DECRepeater 90T and 90C.

drpt90PortTable

A list of port entries. The index for a repeater port is given by (slot number*100) + port number. The number of ports is given by dh90SlotNumberOfPorts.

drpt90PortEntry

A collection of objects containing information for a given interface.

Entry	Type
drpt90PortIndex	INTEGER
drpt90PortName	DisplayString
drpt90PortAdminStatus	INTEGER
drpt90PortState	INTEGER
drpt90PortType	INTEGER
drpt90PortPartitions	Counter
drpt90PortAutoPartitionReason	INTEGER

Entry	Type
drpt90PortJamBits	INTEGER

drpt90PortIndex

Identifies the module and port. The value of this object is given by:
(slot number*100) + port number

drpt90PortName

The text description of the port given by management.

drpt90PortAdminStatus

The desired state of the port.

drpt90PortState

The state of the port. Unknown(1) is returned when the port state cannot be determined. AutoPartitioned(2) indicates that the repeater has disabled the port automatically. ManagementPartitioned(3) indicates that the port has been disabled remotely. Active(4) indicates that the port is operational.

drpt90PortType

The Physical Medium type of the port.

drpt90PortPartitions

The number of times this port has partitioned. This value is only significant if drpt90PortStatus is not unknown(1).

drpt90PortAutoPartitionReason

Errors currently detected on the port.

Num.	Error
0	notPartitioned
1	managementPartitioned
2	excessiveLength
3	excessiveLengthAndMgmtPart
4	excessiveCollisions
5	excessiveCollisionsAndMgmtPart

Num.	Error
6	jabber
7	jabberAndMgmtPart
8	nocarrierLoopback
9	nocarrierLoopbackAndMgmtPart
10	transmitCarrierDropout
11	transmitCarrierDropoutAndMgmtPart

drpt90PortJamBits

The number of jam bits being used on the port segment.

drpt90PortAddrTable

A list of MAC addresses paired with the repeater port on which they have been seen.

drpt90PortAddrEntry

A MAC address and the repeater port on which it resides.

Entry	Type
drpt90PortPhyAddr	PhysAddress
drpt90PortAddrIndex	INTEGER

drpt90PortPhyAddr

A MAC address seen on this port.

drpt90PortAddrIndex

Identifies the module and port. The value of this object the same as drpt90PortIndex and is given by:

(slot number*100) + port number

dbSysChar - System Characteristics Group

This group consists of information about the device's hardware and firmware. It also displays information about hardware and software switches that control device operation.

dbSysRomVersion

The version number of the software stored in ROM.

dbSysInitSwitch

This object allows the management action of initializing a device and forcing it to run self test. It can also be used to reset all information added to the device's NVRAM. When read, it returns a value of other(1). When set to a value of reset, the bridge will empty the forwarding data base and start relearning. When set to resetToFactory, the bridge will empty the forwarding data base, reset spanning tree and clear all protocol filters and password.

dbSysStatus - System Status Group

This group consists of operational status of the device.

dbSysDeviceState

The operational state of the device.

dbSysNvramFailed

A flag, that when set to True, indicates that the NVRAM failed self test following the last initialization of the device. Values set to variables maintained on the bridge will not be preserved over a power cycle of the bridge.

dbIfTable - Extended Interface Module Group

These are interface objects implemented in the DECbridge 90 over and above what is available through MIBII.

dbIfTable

A list of interface entries. The number of entries is given by ifNumber.

dbIfEntry

A collection of objects containing information for a given interface.

Entry	Type
dbIfIndex	INTEGER
dbIfReceiveBadFrames	Counter
dbIfTransmitErrorFrames	Counter

dbIfIndex

Identifies the Interface. The value of this is the same object as the ifIndex for this interface.

dbIfReceiveBadFrames

Number of frames received with a bad frame check sequence or the device received a frame loss indication from the hardware on the interface.

dbIfTransmitErrorFrames

Number of frames that were transmitted with an error on the line.

db90Char - Bridge Characteristics Group

This group consists of information about the bridge's hardware and firmware. It also displays information about hardware and software switches that control device operation.

db90LB100SpanningTreeVer

The version number of the Spanning Tree algorithm used by the bridge when in the LAN Bridge 100 Spanning Tree mode.

db90802SpanningTreeVer

The version number of the Spanning Tree algorithm used by the bridge when in the 802.1d Spanning Tree mode.

db90MaxForwardingDBEntries

The maximum number of address entries that the bridge can store in its volatile memory.

db90MaxNVForwardingDBEntries

The maximum number of permanent address entries that the bridge can store in its NVRAM.

db90MaxProtocolDBEntries

The maximum number of protocol entries that the bridge can store in its protocol database. These entries control the handling of frames based on their Ethernet PT, IEEE 802.2 DSAP or IEEE 802 SNAP Protocol ID.

db90MaxNVProtocolDBEntries

The maximum number of protocol entries that the bridge can store in its nonvolatile memory. These are retained after a power-down.

db90Stat - Bridge Status Group

This group consists of operational status of the bridge.

db90CurrProtocolDBEntries

The number of protocol entries that are stored in the bridge's protocol database. These control the handling of frames based on their Ethernet protocol type, IEEE 802.2 DSAP or 802 SNAP Protocol ID.

db90CurrNVProtocolDBEntries

The number of protocol entries in the bridge's NVRAM. These are retained after a power loss.

db90MgmtHeardPort

The port on which this command was received.

db90LB100BeingPolled

The address of the LAN Bridge 100 mode bridge that sent this bridge into the LAN Bridge 100 Spanning Tree mode. If this bridge is the Root, this object indicates the bridge that will be polled periodically to determine if it is still necessary to stay in this Spanning Tree mode.

db90TimeSinceLastHello

The number of seconds since the bridge last sent a Hello message.

db90HubManagement

Indicates whether the bridge is managing the hub (i.e., is capable of managing repeaters inserted in the hub).

db90CurrFdbEntries

The number of entries currently active in the DECbridge 90 forwarding database, or one more than the value of db90MaxForwardingDBEntries if the bridge has exceeded the size of the forwarding database hardware.

db90Coun - Bridge Counters Group

This group consists of counters that measure operational events and errors.

db90SpanningTreeModeChanges

The number of times that the bridge switched from the 802.1d Spanning Tree mode to the LB100 Spanning Tree mode.

db90Span - Bridge Spanning Tree Group

This group consists of Spanning Tree characteristics over and above what is required by the Bridge MIB.

db90BestRootAge

The age, in hundredths of seconds, of the Hello message that established the best root.

TopologyChangeFlag

A flag that indicates whether a Topology Change is currently in effect on the extended LAN.

db90TellParentFlag

A flag that indicates if the bridge is attempting to propagate a topology change towards the Root.

db90ForwardingDBShortAgingTime

The number of seconds that the bridge keeps learned entries active while a topology change is in effect.

db90BadHelloLimit

The number of hello intervals during which the bridge receives one or more bad hellos on a line, before the bridge performs a test on the link. A bad Hello message is one that contains inferior information that is received on a port on which this bridge is Designated.

db90BadHelloResetTimer

The number of Hello intervals without bad Hellos that the bridge will wait before it resets its bad Hello count to zero.

db90NoFrameInterval

The number of seconds of inactivity on a line that will cause the bridge to run a test on that line. The bridge considers a line to be inactive if it does not receive any frames on that line.

db90LB100PollTime

The number of seconds that a Root bridge in LAN Bridge 100 mode waits between polling the LAN Bridge 100 that is keeping it in this mode. This polling is done to determine whether the LB100 is still present on the extended LAN.

db90LB100ResponseTimeout

The number of seconds that a Root in LAN Bridge 100 Spanning Tree mode will wait for a response from the LAN Bridge 100 which is keeping it in this mode, before it will assume that the LB100 is no longer on the extended LAN.

db90LB100SpanningTreeCompat

A switch that controls the Spanning Tree mode used by the bridge. If in Auto-Select mode, the bridge will go into 802 Spanning Tree mode by default, but will switch to LB100 mode as soon as a LB100 is detected. In 802 Spanning Tree mode, the bridge will stay in 802 mode.

db90IfTable - Extended Bridge Port Table

This group consists of generic port objects for the bridge.

db90IfTable

A list of interface entries. The number of entries is given by ifNumber.

db90IfEntry

A collection of objects containing information for a given interface.

Entry	Type
db90IfIndex	INTEGER
db90IfReceiveDeviceFrames	Counter
db90IfExceededBadHelloLimits	Counter

db90IfIndex

Identifies the Interface. The value of this object is the same as the ifIndex for this interface.

db90IfReceiveDeviceFrames

The number of frames addressed to the bridge itself that were received on this line.

db90IfExceededBadHelloLimits

The number of times that the Bad Hellos on the line exceeded the Bad Hello limit.

db90IfEtherTable - Extended Bridge Ethernet Port Table

This optional group consists of Ethernet port objects for the bridge.

db90IfEtherTable

A list of interface entries for an Ethernet port.

db90IfEtherEntry

A collection of objects containing information for a given interface.

Entry	Type
db90IfEthIndex	INTEGER
db90IfEthFramingErrors	Counter
db90IfEthCarrierLosses	Counter
db90IfEthExceededCollisionLimits	Counter

db90IfEthIndex

Identifies the Interface. The value of this object is the same as the ifIndex for this interface.

db90IfEthFramingErrors

The number of times that a frame received on the line contained both a noninteger multiple of 8 bits and a CRC error.

db90IfEthCarrierLosses

The number of times that the bridge detected a loss of the carrier signal while transmitting a frame on the line.

db90IfEthExceededCollisionLimits

The number of times that the bridge failed to transmit a frame on this line after 16 attempts, the collision limit.

db90IfSpanTable - Extended Bridge Spanning Tree Port Table

This group consists of port Spanning Tree objects for the bridge.

db90IfSpanTable

A list of interface entries for a port.

db90IfSpanEntry

A collection of objects containing information for a given interface.

Entry	Type
db90IfSpIndex	INTEGER
db90IfSpDesigRootAge	INTEGER
db90IfSpForwardDelayTimer	INTEGER
db90IfSpBadHelloCounts	Counter
db90IfSpTopologyChangeAckFlag	INTEGER

db90IfSpIndex

Identifies the Interface. The value of this object is the same as the ifIndex for this interface.

db90IfSpDesigRootAge

The age, in hundredths of seconds, of the last Hello message received from the designated bridge on the line.

db90IfSpForwardDelayTimer

The time remaining, in hundredths of seconds, before the bridge will leave the Learning State of Preforwarding and enter the Forwarding State.

db90IfSpBadHelloCounts

The number of Hello intervals during which at least one Bad Hello was received.

db90IfSpTopologyChangeAckFlag

A flag that indicates whether a topology change notification received on a link that we are designated on needs to be acknowledged.

DECbridge 90 Protocol Database

db90ProtoFilterOther

Action taken by the bridge on Ethernet protocol types or the 5-byte SNAP SAPs other than those specified in the db90ProtoTable.

Action	Description
Forward (1)	Indicates that a filter all entry exists in the db90ProtoTable. This value also implies that no forward all entry may be added to the db90ProtoTable.
Filter (2)	Indicates that a forward all entry exists in the db90ProtoTable. This value also implies that no filter all entry may be added to the db90ProtoTable.
Default (3)	Indicates that an entry of either forward all or filter all (i.e. not just Multicast frames) may be added to the db90ProtoTable. This value also implies that all other protocol entries not listed in the db90ProtoTable will be forwarded.

db90ProtoFilterTable

A table that contains filtering information about Ethernet protocol types and 5 byte SNAP SAPs for the DECbridge 90.

db90ProtoFilterEntry

A table that contains filtering characteristics for Ethernet protocol types and 5 byte SNAP SAPs for the DECbridge 90.

Entry	Type
db90ProtoFilterProtocol	OCTET STRING
db90ProtoFilterType	INTEGER
db90ProtoFilterStatus	INTEGER
db90ProtoFilterMulticastFlag	INTEGER
db90ProtoFilterPortMask	INTEGER

db90ProtoFilterProtocol

The protocol type or SNAP SAP in a frame to which this entry's filtering information applies. The length is 2 for Ethernet protocol types and 5 for SNAP SAP.

db90ProtoFilterType

The type of this entry.

Type	Description
Unknown (1)	Specifies that this protocol filter is unused.
Invalid (2)	Deletes the filter.
Ethernet (3) or snap-sap (4)	Defines the size of db90ProtoFilterProtocol. Setting this variable to a value of delete.
AllProtocols (5)	Causes all protocols, both volatile and non-volatile, to be deleted.

db90ProtoFilterStatus

Action taken by bridge when it sees this Ethernet protocol type or SNAP SAP in a received frame. Note that this value must be identical to db90ProtoFilterOther unless db90ProtoFilterMulticastFlag is multicastOnly(2) or there are no other entries with db90ProtoFilterMulticastFlag=allFrames(1).

db90ProtoFilterMulticastFlag

Identifies whether or not filtering is based on the multicast bit in the MAC destination address. If allFrames(1), then filtering is performed as per db90ProtoFilterStatus. For multicastOnly(2), multicast frames for this protocol are always filtered and unicast frames are always forwarded; this is true regardless of the value of db90ProtoFilterOther.

db90ProtoFilterPortMask

Must be allPorts(3) unless db90ProtoFilterMulticastFlag is multicastOnly, in which case any specification is valid.

F

Documentation and Ordering

Introduction

This appendix lists documentation that is related to the HUBwatch for Windows application. Also included is the necessary ordering information.

Related Documentation

You can order the following documents from Digital:

Document Title	Order Number
DEChub 90 Owner's Manual	EK-DEHUB-OM
Open DECconnect Building Wiring Components and Application Catalog	EB-K2407-42
DECconnect System Planning and Configuration Guide	EK-DECSY-CG
DECagent 90 User Information	EK-DENMA-UI
DECbridge 90 Owner's Manual	EK-DEWGB-OM
DECrepeater 90C Owner's Manual	EK-DECMR-OM
DECrepeater 90T Owner's Manual	EK-DETMR-OM
DECserver 90L Owner's Manual	EK-DSRVD-OM
DECserver 90L+ Owner's Manual	EK-DSRVG-OM
HUBwatch Installation & Use for DECMcc	AA-PW4BA-TE
HUBwatch for Windows (Kit)	EK-478AA-DK
HUBwatch for Windows DECserver 90 Management	EK-485AA-UI
HUBwatch for Windows DECbridge 90 Management	EK-488AA-UI
HUBwatch for Windows DECrepeater 90 Management	EK-490AA-UI

Ordering Information

You can order options and documentation by mail, phone, or electronically.

Need Help?

If you need help deciding which documentation best meets your needs, please call 800-DIGITAL (800-344-4825) and press 2 for technical assistance.

Electronic Orders

To place an order through your account at the Electronic Store, dial 800-234-1998, using a modem set to 2400 or 9600 baud. You must use a VT terminal or terminal emulator set at 8 bits, no parity. If you need help, call 800-DIGITAL (800-344-4825) and ask for an Electronic Store specialist.

Telephone or Direct Mail Orders

You can order documentation by phone or direct mail.

If You Are From . . .	Call . . .	Or Write . . .
U.S.A.	DECdirect Phone: 800-DIGITAL (800-344-4825) FAX: (603) 884-5597	Digital Equipment Corporation P.O. Box CS2008 Nashua, NH 03061
Puerto Rico	Phone: (809) 781-0505 FAX: (809) 749-8377	Digital Equipment Caribbean, Inc. 3 Digital Plaza, 1st Street Suite 200 Metro Office Park San Juan, Puerto Rico 00920
Canada	Phone: 800-267-6215 FAX: (613) 592-1946	Digital Equipment of Canada Ltd. 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International	—	Local Digital subsidiary or approved distributor

Digital Personnel

You can order documentation by electronic mail. Contact the following organizations for instructions:

If You Need . . .	Call . . .	Contact . . .
Software documentation ¹	DTN: 241-3023 (508) 874-3023	Software Supply Business Digital Equipment Corporation 1 Digital Drive Westminster, MA 01473
Hardware documentation	DTN: 234-4325 (508) 351-4325 FAX: (508) 351-4467	Publishing & Circulation Services Digital Equipment Corporation NRO2-2/I5 444 Whitney Street Northboro, MA 01532

¹Call to request an Internal Software Order Form (EN-01740-07).

Glossary

Introduction

This glossary defines terms used in this manual.

Agent

A software system on a managed device that processes SNMP requests for management information. It uses its knowledge of the internal structure of the device to retrieve the information and return an SNMP response to the requester. Within HUBwatch, this is generally referred to as an SNMP agent.

Alarm icon

A traffic light symbol showing the network alarm status.

Alarm log

A record of alarm activity for the entire network. You can view it online, output it to a printer, or save it as a text or delimited ASCII file.

Auto Discovery

A HUBwatch utility that automatically finds all PING-only and SNMP-manageable devices within a specified range of IP addresses.

Background views

An .BMP file that is used as a backdrop to a HUBwatch.

Bridge

A protocol-independent network device that provides for the exchange of packets between the physical networks it connects. It can divide large networks into smaller, interconnected segments. It can also serve as a tool for fault isolation.

Buttons

A control in the HUBwatch windows for choosing operations or actions.

Clarkson Packet Driver

A public domain packet driver interface for inband communications with a Network Interface Card (NIC).

Coaxial cable

A single conductor surrounded by insulation and a conductive shield. This shield prevents the cable from picking up or emitting electrical noise.

Community string

A community string acts as a password for SNMP messages. When a device receives an SNMP request, it checks whether the community string in the request matches that in its configuration. If the community string matches, the device responds to the request. Otherwise, it ignores the request and issues an authentication failure trap.

DARPA

See ICMP.

Delimited ASCII file

A file with columns of information separated by commas or spaces.

Device

A generic SNMP element such as an agent, bridge, repeater or server.

Device driver

A program that translates requests for the attached hardware.

Device expert

A device expert provides for management and control of families of networking products. Runs under and as a part of the HUBwatch engine. It also provides information to the HUBwatch Engine.

Device view

A view within the HUBwatch application that is similar to the module view, but applies only to devices that are not a part of the hub.

Generic device

Any SNMP-manageable device for which a HUBwatch Device Expert does not exist. Device Experts exist only for devices in the DEChub 90 family of products.

Host

An end-system.

Hub

A generic term used to describe a multiport repeater.

Hub view

A view within the HUBwatch application that displays the entire hub with one or more devices in the slots. This view also provides access to MIB information.

HUBwatch engine

An engine within HUBwatch that provides network management facilities for SNMP- manageable network elements and Ping-only devices.

ICMP

Internet Control Message Protocol. A large collection of connected networks, primarily in the United States, running the Internet suite of protocols. Sometimes referred to as the DARPA Internet

Icons

Graphic representations of network elements such as bridges, routers and sites.

IEEE

A professional organization which, as a part of its services to the community, performs some pre-standardization work for OSI.

Institute of Electrical and Electronics Engineers (IEEE)

See IEEE.

Internet

A collection of local data networks connected by gateways and protocols that allow them to function as a single large network.

Internet address

IP address.

Internet Control Message Protocol (ICMP)

See ICMP.

Internet Protocol (IP)

See IP.

IP address

A 32-bit quantity used to represent a point of attachment in an Internet.

IP

Internet Protocol. A network layer protocol providing best-effort delivery within the Internet suite of protocols. It is responsible for providing transparency over both the topology of the Internet and the transmission media used in each physical network.

Jabber

A term used to denote continuous transmission of corrupted or random data onto a network. This is usually caused by network interface card failures.

LAN

Local area network. Any one of a number of technologies providing high speed, low-latency transfer and being limited in geographic size.

Local area network (LAN)

See LAN.

MAC

Medium Access Control. The lower half of the Data Link Layer of the OSI model that defines access to the transmission media.

Management Information Base (MIB)

See MIB.

Medium Access Control (MAC)

See MAC.

MIB

Management Information Base. A collection of objects that can be accessed by way of a network management protocol.

MIBI

An internet standard MIB (RFC 1156).

MIBII

An internet standard MIB (RFC 1213).

Module

A board that plugs into a hub or concentrator.

Module view

A view within the HUBwatch application that displays the individual devices displayed in the DEChub at the hub view along with the menu items that go along with the selected module.

NDIS

Network Driver Interface Specification. The specification for a generic LAN Manager-based device driver that is hardware-independent and protocol-independent. This was developed by Microsoft and 3Com.

Network

A collection of subnetworks connected by intermediate-systems and populated by end-systems. Also (Internet usage), a single subnetwork or a related set of subnetworks in the OSI sense.

Network administrator

A network administrator is the individual responsible for managing the network from the management station.

Network elements

The devices interconnected on the network.

Network management

A technology used to manage an Internet.

Network management station

A system responsible for managing the network.

Network map

A display showing the logical elements that compose the network.

Network view

The top-level view of the HUBwatch application.

ODI

Open Datalink Interface. Novell's media-and-protocol-independent communications specification providing a standard interface that allows transport protocols to share a network interface card without conflict.

Open Datalink Interface (ODI)

See ODI.

Open Systems Interconnection (OSI)

See OSI.

OSI

Open Systems Interconnection. An international effort to facilitate communications among computers of different manufacture and technology.

Packet

A collection of data bits and associated control information bits that is transmitted as a whole unit, the basic unit of information transfer.

Passwords

Regular printable keyboard characters, that when entered correctly, provide the user with access to a specific network. When passwords are entered, the characters are not echoed on the display. Passwords are not case sensitive and can be a maximum of 20 characters in length. Passwords are also optional, but recommended. You can change them through the HUBwatch for Windows Security menu.

PDU

Protocol Data Unit. A data object exchanged by protocol machines, usually containing both protocol control information and user data.

Ping

A program used to test IP-level connectivity from one IP address to another.

Port number

Identifies an application entity to a transport service in the Internet suite of protocols.

Protocol

A set of formats, rules and standards for network communications.

Protocol Data Unit (PDU)

See PDU.

Proxy

A term used to define software that fronts for another system. In the SNMP world, proxy agents are used to avoid implementing full protocol stacks in simple devices.

Repeater

A networking device that receives a digital signal from one cable, amplifies it, and then reconstructs the signal for transmission onto another cable. It does not filter packets or make routing decisions, but simply transfers packets from LAN to LAN.

Request for Comments (RFC)

See RFC.

RFC

Request for Comments. A document available on the Internet describing the Internet suite of protocols and related experiments.

Router

A networking device used to connect LANs by examining data addresses and choosing the most efficient path to the destination. Like a bridge, a router restricts a LAN's local traffic, only passing data on to the routed path when that data is specifically intended for it.

Simple Network Management Protocol (SNMP)

See SNMP.

Site

A network element that incorporates devices into single large groupings. For example, it may reflect a different corporate group or building in the network.

Site view

A view within the HUBwatch application that displays all the site icons. The organization of network elements into sites is arbitrary. Since sites are only convenient tools for planning and administering the network, for practical purposes, a site view is equivalent to a network view.

SNMP

Simple Network Management Protocol. The application protocol offering network management service in the Internet suite of protocols. It is part of the TCP/IP protocol suite and usually runs over the connectionless User Datagram Protocol (UDP).

Subnet mask

A 32-bit quantity indicating which bits in an IP address identify the physical network.

TCP

Transmission Control Protocol. The transport protocol offering a connection-oriented transport service in the Internet suite of protocols.

Token Ring

A network in which the computers are connected together in a ring. A special message called the token is passed from one machine to another around the ring and each machine can transmit only while it is holding the token.

Transmission Control Protocol (TCP)

See TCP.

UDP

User Datagram Protocol. The transport protocol offering a connectionless-mode transport service in the Internet suite of protocols.

User Datagram Protocol (UDP)

See UDP.

Variable

A pairing of an object instance name and associated value.

WAN

Wide area network. Any one of a number of technologies providing geographically distant transfer.

Wide area network (WAN)

See WAN.

X.25

A connection-oriented network facility.

Index

A

Adding

- connections, 5-13
- devices, 5-4
- icons
 - specialized network, 5-14
- objects at the hub view, 5-18
- Sites, 5-3

Adding objects, 5-3

Address Translation (AT) group, 6-26, E-5

Agent

- Accessing information, 8-3
- managing the configuration, 8-3
- managing the faults, 8-9
- managing the performance, 8-8

Agent Module report, 8-4

Alarm icons, 3-2

Alarm log, 6-14

Alarms option, 6-9

Async group, 6-46

AT group, E-5

Audible Alarms option, 6-10

Auto Discovery, 5-15

AUTOEXEC.BAT

- modifying, 2-4

B

Background maps, 3-7, D-1

Backing up a network, 5-27

Bridge groups, 6-37

Bridge icons, 3-3

Bridge MIB, E-22

Brief Configuration report, 6-52

Buttons, 4-3

C

Change Log, 6-54

Char-like groups, 6-43

Char-like MIB, E-34

Clarkson Packet Driver, C-1

CONFIG.SYS

- modifying, 2-5

Configuring

- network management stations, 2-3
- networks, 5-1

Connections

- adding, 5-13
- deleting, 5-19
- network, 3-6

Contents

- HUBwatch kit, 2-2

Conventions

- Manual, xiv

Creating networks, 5-1

D

DECagent

- managing the configuration, 8-3
- managing the faults, 8-9
- managing the performance, 8-8

- DECagent 90
 - Accessing information, 8-3
- DECagent 90 module
 - managing, 8-1
- DECagent Error report, 8-11
- DECagent Performance report, 8-8
- Defining connections, 5-14
- Deleting
 - connections, 5-19
 - devices, 5-19
 - networks, 5-30
 - objects at the hub view, 5-20
 - objects at the network or site view, 5-18
 - other icons, 5-20
- Deleting specialized network icons, 5-20
- Description
 - product, 1-1
- Device icon
 - colors, 3-2
- Device icons, 3-2
- Devices
 - adding, 5-4
 - deleting, 5-19
 - modifying, 5-22
- DH_90 MIB, E-45
- Documentation, F-1
 - ordering, F-2
- Dot1dBase group, 6-38
- Dot1dSr group, 6-40
- Dot1dStatic group, 6-42
- Dot1dStp group, 6-39
- Dot1dTp group, 6-41

E

- Edit Strings report, 8-6
- EGP group, E-16
- Error Statistics option, 6-8
- Exterior Gateway Protocol (EGP) group, 6-35, E-16

F

- Features, 1-1
 - of MIB devices, 6-22
- Find Me option, 4-7
- Find option, 4-5
- Full Configuration report, 6-53

G

- General group (Char-like), 6-43
- General groups (RS232-like), 6-45
- Generic device icons, 3-5
- Generic devices
 - accessing, 7-1
 - managing, 7-1
- Glossary, Glossary-1
- Graph option, 6-3

H

- Hardware requirements, 2-1
- Host icons, 3-4
- How to order
 - documentation, F-1
- Hub icons, 3-4
- Hubs
 - modifying, 5-22
- HUBwatch
 - buttons, 4-3
 - installing, 2-3
 - menus, 4-2
 - starting a session, 4-1
 - tools, 3-1
 - views, 4-1

I

- ICMP group, E-10
- Icons
 - adding
 - specialized network, 5-14
 - alarm, 3-2
 - bridge, 3-3
 - deleting

Icons

- deleting (cont'd)
 - specialized network, 5–20
- device, 3–2
- generic device, 3–5
- host, 3–4
- hub, 3–4
- mapping, 3–1
- repeater, 3–4
- router, 3–3
- site, 3–3
- specialized network, 3–5
- terminal server, 3–4

Installing

- HUBwatch, 2–3
- Interface (IFT) group, E–2
- Interface (ITF) group, 6–25
- Internet Control Message Protocol (ICMP) group, 6–32, E–10
- Internet Protocol (IP) group, 6–27, E–5
- IP group, E–5
- ITF group, E–2

K

- Kit contents, 2–2

L

- Local Configuration report, 6–51

M

Managing

- DECagent 90 modules, 8–1
- generic devices, 7–1
- MIB information, 6–23
- network faults, 6–7
- network performance, 6–1
- networks, 6–1
- Network security, 6–16
- Managing DECagent 90 configuration, 8–3
- Managing DECagent 90 faults, 8–9

- Managing DECagent 90 performance, 8–8

- Mapping networks, 3–1

- Menus, 4–2, A–1

MIB

- accessing bridge groups, 6–37
- accessing Char-like groups, 6–43
- accessing RS232-like groups, 6–45
- accessing the, 6–21
- Address Translation group, 6–26
- Bridge descriptions, E–22
- Char-like descriptions, E–34
- DH_90 descriptions, E–45
- Dot1dSr group, 6–40
- Dot1dStatic group, 6–42
- Dot1dStp group, 6–39
- Dot1dTp group, 6–41
- EGP group, 6–35
- features, 6–22
- ICMP group, 6–32
- Interface group, 6–25
- IP group, 6–27
- managing information, 6–23
- RS232-like descriptions, E–40
- SNMP group, 6–36
- System group, 6–24
- TCP group, 6–33
- UDP group, 6–34
- walking the, 6–50

MIB, Accessing

- MIBII group, 6–23
- MIBII, D–1, E–1
- Descriptions, E–1
- MIB Object Graph report - agent, 8–8
- MIB variables, 6–48

Modifying

- devices, 5–22
- hubs, 5–22
- networks, 5–30
- sites, 5–22
- views, 5–22

Moving

- objects, 5–21
- objects in same network, 5–21

N

Navigating

within HUBwatch, 4-4

NETDEV.SYS, 2-5

Network connections, 3-6

Network management stations

configuring, 2-3

Networks

backing up, 5-27

configuring, 5-1

creating, 5-1

deleting, 5-30

managing, 6-1

managing faults, 6-7

managing performance, 6-1

modifying, 5-30

opening established, 5-26

restoring, 5-29

Networks, Managing

Security, 6-16

NOS

with HUBwatch, B-1

Notes, 3-8

O

Objects

adding, 5-18

adding at network and site view, 5-3

deleting, 5-18, 5-20

moving, 5-21

in same network, 5-21

to another network, 5-21

Opening

established networks, 5-26

Opening networks, 4-1

Options

Alarms, 6-9

Audible Alarms, 6-10

Error Statistics, 6-8

Find, 4-5

Find Me, 4-7

Graph, 6-3

Options (cont'd)

ordering, F-2

Ping, 6-5

Set Thresholds, 6-11

Statistics, 6-1

Zoom In, 4-8

Zoom Out, 4-8

Zoom Top, 4-8

Ordering information, F-1

Organization

Manual, xiii

P

Packet drivers

Clarkson, C-1

Passwords

setting, 6-17

Ping option, 6-5

Printer setup, 4-8

Printing, 4-8

Product description, 1-1

R

Repeater icons, 3-4

Reports

accessing, 6-50

Agent Module, 8-4

Alarm Log, 6-14

Brief Configuration, 6-52

DECagent Error, 8-11

DECagent Performance, 8-8

Edit Strings, 8-6

Full Configuration, 6-53

Local Configuration, 6-51

MIB Object Graph - agent, 8-8

Printing, 8-13

viewing security, 6-18

Requirements

hardware, 2-1

software, 2-2

Restoring networks, 5-29

Router icons, 3-3
RS232-like groups, 6-45
RS232-like MIB, E-40

S

Security log, 6-18
Session group, 6-44
Set Thresholds option, 6-11
Setting
 passwords, 6-17
Simple Network Management Protocol (SNMP) group, 6-36, E-18
Site icons, 3-3
Sites
 modifying, 5-22
SNMP group, 6-36, E-18
Software requirements, 2-2
Specialized network icons, 3-5, 5-14
 deleting, 5-20
Starting a HUBwatch session, 4-1
Statistics option, 6-1
Sync group, 6-47
System group, 6-24, E-1

T

TCP group, 6-33, E-12
Terminal server icons, 3-4
Terms
 defined, Glossary-1
Thresholds
 Setting, 8-12
Tools
 HUBwatch, 3-1
Transmission Control Protocol (TCP) group, 6-33, E-12

U

UDP group, 6-34, E-15
User Datagram Protocol (UDP) group, 6-34, E-15

V

Views, 4-1
 modifying, 5-22

Z

Zoom In option, 4-8
Zoom Out option, 4-8
Zoom Top option, 4-8

