

PATHWORKS for DOS

digital

DECnet User's Guide



PATHWORKS for DOS

DECnet User's Guide

Order Number: AA-PAFGC-TK

September 1991

This guide describes how to set up and use DECnet-DOS, part of the PATHWORKS for DOS software. It also explains the DECnet-DOS commands you use to perform network tasks.

Revision/Update Information:	This is a revised guide.
Operating System/Version:	See the PATHWORKS for DOS <i>Software Product Description</i> for this information.
Software Version:	PATHWORKS for DOS V4.1

**Digital Equipment Corporation
Maynard, Massachusetts**

September 1991

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.

Copyright © 1989, 1991 by Digital Equipment Corporation
All Rights Reserved.
Printed in U.S.A.

The postpaid Reader's Comments form at the end of this document requests your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

ALL-IN-1, DEC, DECmate, DECnet, DECnet-DOS, DECnet-ULTRIX, DECrouter, DECsystem, DECstation, DECwrite, Ethernet, LAT, PATHWORKS, PCSA, RSX-11M, RSX-11M-PLUS, RSTS/E, Scholar, ThinWire, TOPS-10, TOPS-20, ULTRIX, VAX, VAXcluster, VMS, VT, WPS-PLUS, and the Digital logo.

The following are third-party trademarks:

3Com and EtherLink are registered trademarks of 3Com Corporation. COMPAQ is a registered trademark of COMPAQ Computer Corporation. Hayes is a registered trademark of Hayes Microcomputer Products, Inc. Hercules is a registered trademark of Hercules Computer Technology. IBM, PC/XT, and Personal Computer AT are registered trademarks of International Business Machines Corporation. Microsoft, MS, MS-DOS, and MS-NET are registered trademarks of Microsoft Corporation. Olivetti is a registered trademark of Ing. C. Olivetti. Tandy is a registered trademark of Tandy Corporation. Toshiba is a registered trademark of Kabushiki Kaisha Toshiba. UNIX is a registered trademark of UNIX System Laboratories. Zenith is a trademark of Zenith Electronics Corporation.

This manual was produced by Telecommunications and Networks Publications.

Contents

Preface	ix
1 Introducing DECnet	
1.1 What Is DECnet?	1-1
1.2 What Is DECnet-DOS?	1-2
1.3 Managing and Monitoring Your Network Software	1-4
1.4 Programming Task-to-Task Communications	1-5
1.5 Using Microsoft Windows with DECnet-DOS Utilities	1-5
2 Spawning Multiple Services	
2.1 Overview of the Job Spawner	2-1
2.2 Creating a Job Spawner Database File	2-2
2.3 Using the Job Spawner	2-5
3 Providing Remote File Access to Your Node	
3.1 Starting the File Access Listener Utility from a Command Line	3-1
3.2 Entering the File Access Listener Utility As an Object	3-3
3.3 How the File Access Listener Utility Accesses Files	3-3
3.4 Restrictions	3-4
3.5 The File Access Listener Utility Command Syntax	3-5
4 Accessing Local and Remote Files	
4.1 Running the Network File Transfer Utility	4-2
4.1.1 Command Line Method	4-2
4.1.2 Microsoft Windows Environment	4-3
4.2 Entering the Network File Transfer Utility Commands	4-3
4.2.1 Command Prompts	4-5
4.2.2 Command Switches	4-5

4.3	Using the HELP Command	4-6
4.4	How to Exit the Network File Transfer Utility	4-7
4.5	Defining Remote Node Access Information	4-7
4.5.1	Defining Access Control Information with Network File Transfer	4-8
4.5.2	Using Default Access Control Information	4-9
4.5.3	Changing the Default Access Control Information	4-10
4.5.4	Displaying Access Control Information	4-11
4.6	Using File Specifications in DECnet-DOS	4-11
4.7	Types of Files and Systems	4-12
4.8	Manipulating Files with the Network File Transfer Utility	4-14
4.8.1	Appending Files	4-15
4.8.2	Copying Files	4-15
4.8.3	Deleting Files	4-16
4.8.4	Displaying Directory Information	4-16
4.8.5	Printing Files	4-17
4.8.6	Displaying the Contents of a File	4-17
4.8.7	Running Command Files	4-18
4.8.8	Logging File Operations	4-18
4.9	Network File Transfer Utility Command Summary	4-19
	APPEND	4-20
	COPY	4-24
	DELETE	4-32
	DIRECTORY	4-33
	EXIT	4-35
	HELP	4-36
	PRINT	4-38
	SET	4-40
	SHOW	4-41
	SUBMIT	4-43
	TYPE	4-44

5 Accessing Remote Printers and Disks

5.1	Installing the Virtual Device Drivers	5-5
5.2	How to Run the Network Device Utility	5-5
5.3	How to Exit the Network Device Utility	5-6
5.4	Entering the Network Device Utility Commands	5-6
5.4.1	Command Verbs	5-7
5.4.2	Entities	5-8
5.5	Using the HELP Command	5-10

5.6	Special Considerations While Using Virtual Disks	5-11
5.7	Network Device Utility Command Summary	5-11
	CLOSE	5-12
	CREATE	5-14
	DELETE	5-17
	HELP	5-18
	OPEN	5-20
	SHOW	5-22

A File Specifications for Accessing Remote Files

A.1	Specifying Remote Files	A-1
A.2	Using Wildcards	A-2

B Network File Transfer Utility Error Messages

B.1	Types of Network File Transfer Utility Error Messages	B-1
B.2	Command Line Error Messages	B-3
B.2.1	Warnings	B-6
B.3	File Input/Output Error Messages	B-7
B.4	Network Error Messages	B-9
B.4.1	Secondary Error Messages	B-11
B.5	Miscellaneous Error Messages	B-14
B.5.1	Warnings	B-25

C Network Device Utility Error Messages

C.1	Virtual Disk Error Messages	C-2
C.1.1	Command Line Error Messages	C-2
C.1.2	Network Errors	C-3
C.1.3	Execution Error Messages	C-5
C.2	Virtual Printer Error Messages	C-10
C.2.1	Command Line Error Messages	C-10
C.2.2	Network Errors	C-11
C.2.3	Execution Error Messages	C-12

D Data Access Protocol Error Messages

D.1	Overview	D-1
D.1.1	Maccode Field	D-1
D.1.2	Miccode Field	D-2

Glossary

Index

Examples

2-1	Spawner Receiving a Request	2-6
-----	-----------------------------------	-----

Figures

1-1	A Sample DECnet Network	1-3
4-1	Network File Transfer Command Example	4-4
4-2	Copying a Local File to a Remote Node	4-12
5-1	Using Virtual Disks on DECnet-DOS	5-2
5-2	Using Remote Virtual Printers on DECnet-DOS	5-3
5-3	Parts of a Network Device Utility Command	5-7

Tables

1-1	What You Can Do with DECnet-DOS	1-4
2-1	Predefined Object Numbers	2-3
4-1	Copying Files from a Remote Node	4-13
4-2	Copying Files to a Remote Node	4-14
4-3	Network File Transfer Commands	4-19
5-1	Network Device Utility (NDU) Commands for Virtual Disks	5-7
5-2	Network Device Utility Command Entities	5-8
A-1	Operating Systems and File Specifications	A-1
D-1	Data Access Protocol Maccode Field Values	D-1
D-2	Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11	D-3

D-3	Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7	D-11
D-4	Data Access Protocol Miccode Values for Use with Maccode Value 12	D-21

Preface

The DECnet-DOS™ software, part of the PATHWORKS™ for DOS communication product, enables individual computer systems to communicate with one another in a network.

This guide provides the information you need to operate a DECnet-DOS system within a DECnet™ network. This guide:

- Defines the basic DECnet network utilities.
- Explains the terms and concepts related to DECnet networks.
- Defines and explains the command set for each basic utility of DECnet-DOS Version 4.1.

In this guide:

- The term *DOS* refers to MS-DOS™ or PC-DOS™ running on supported personal computers.
- The term *DECnet-DOS* refers to DECnet-DOS Version 4.1 running on supported industry-compatible systems that run the DOS operating system.
- The term *personal computer* refers to all industry-compatible personal computers.

For a list of all supported personal computers and supported versions of the DOS operating system, see the *PATHWORKS for DOS Software Product Description*.

Objectives of This Guide

This guide describes the DECnet utilities:

- Job Spawner (SPAWNER)
- File Access Listener (FAL)
- Network File Transfer (NFT)

- Network Device Utility (NDU)

An overview section describes DECnet network concepts and defines DECnet network terms. The network management utilities and terminal emulation utilities are included in separate guides. However, this guide includes an overview of networking terms, network management concepts, and terminal emulation.

The PATHWORKS for DOS, *User's Handbook* provides an overview of DECnet Client for DOS. It also includes an explanation of the product set, a brief description of DECnet-DOS utilities and the tasks that you perform with them, and an introduction to frequently used DECnet-DOS commands.

Intended Audience

This guide is intended for users who want to expand the capabilities of their personal computers in order to share data and resources with other DECnet network systems. You should be familiar with personal computers, the DOS operating system, networking concepts, and network systems.

Structure of This Guide

This guide consists of the following chapters and appendixes:

Chapter 1	Introduces the capabilities of DECnet-DOS. It also defines some basic concepts within a DECnet environment and introduces the DECnet-DOS utilities.
Chapter 2	Describes how to use the DECnet-DOS Job Spawner to detect incoming connection requests from other nodes in the network.
Chapter 3	Describes how to use the File Access Listener utility for providing (or limiting) remote file access to your node from other nodes in the network.
Chapter 4	Describes how to use the Network File Transfer utility to access files on other nodes in the network.
Chapter 5	Describes how to use the Network Device Utility to set up disk drives and use printers on remote nodes.
Appendix A	Describes file specifications in detail.
Appendix B	Lists the Network File Transfer utility error messages.
Appendix C	Lists the Network Device Utility error messages.
Appendix D	Lists the Data Access Protocol error messages.
Glossary	Contains a list of terminology specific to this documentation set.

Conventions Used in This Guide

The following conventions are used in this guide:

Convention	Meaning
Special type	Indicates examples of system output or user input. System output is in black; user input is in teal.
COMMAND	Bold characters represent acceptable abbreviations for commands. For example, DELETE indicates that the acceptable abbreviation for the DELETE command is DEL.
UPPERCASE	In commands and examples, uppercase lettering indicates that you should enter the exact characters shown. However, you can enter them in either uppercase or lowercase.
<i>italics</i>	In commands and examples, italics indicates a value that either the system supplies as default or you should supply.
boldface	In text, boldface wording identifies a glossary item.
{ }	Braces indicate that you are required to specify one (and only one) of the enclosed options. Do not type the braces when you enter the command.
[]	Brackets indicate that the enclosed data is optional. (If a vertical list of options is enclosed, you can specify only one option.) Do not type the brackets when you enter the command.
()	Parentheses enclose a set of options that must be specified together.
vertical list of options	A vertical list of options not enclosed within braces, brackets, or parentheses indicates that you can specify any number of options (or if defaults apply, none).
<code>key</code>	Indicates that you should press the specified key. For example, <code>Ctrl/x</code> indicates that you should hold down the Control key while you press the <i>x</i> key, where <i>x</i> is a letter.
<code>Return</code>	Indicates that you should press the key that executes commands. This key is <code>Enter</code> , <code>Return</code> , or <code>↵</code> , depending on your keyboard.

Related Documents

It is important that you also read the contents of the file `README.TXT`, which is included on the first floppy diskette of your installation kit.

The following is a list of associated documents in the PATHWORKS for DOS documentation set:

- *Client Commands Reference*
- *DECnet Network Management Guide*
- *DECnet Programmer's Reference Manual*
- *Installing and Configuring (with Diskettes)*
- *Microsoft Windows Support Guide*
- *SETHOST Terminal Emulation Guide*
- *Software Product Description*
- *User's Handbook*

You should also have the installation guide and manuals for the DOS operating system and for your personal computer.

The postage-paid Reader's Comments form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

Introducing DECnet

This chapter provides an overview of DECnet and covers the following topics:

- What is DECnet?
- What is DECnet-DOS?
- Managing and monitoring your network.
- Programming task-to-task communications.
- User interfaces.
- Using Microsoft Windows with DECnet-DOS utilities.

1.1 What Is DECnet?

DECnet is Digital's family of software and hardware communications products that let users of various Digital and other vendor computer systems participate in a computer network.

The network environment of DECnet allows any computer or node to communicate with every other node in the network, without consulting a central controlling node. In this environment, each node is equally responsive to user requests, allowing network users to easily gain access to applications and facilities on other nodes. This simplifies communications and data handling, and provides flexibility when configuring a network.

DECnet software, located on each system in a DECnet network, provides you with an interface that extends your computer's operating system. This network interface lets users communicate and share resources with other users in the DECnet network. Users in a network can exchange information, files, and programs. Refer to the *PATHWORKS for DOS, Systems Support Addendum (SSA)* for a list of supported personal computers and operating systems.

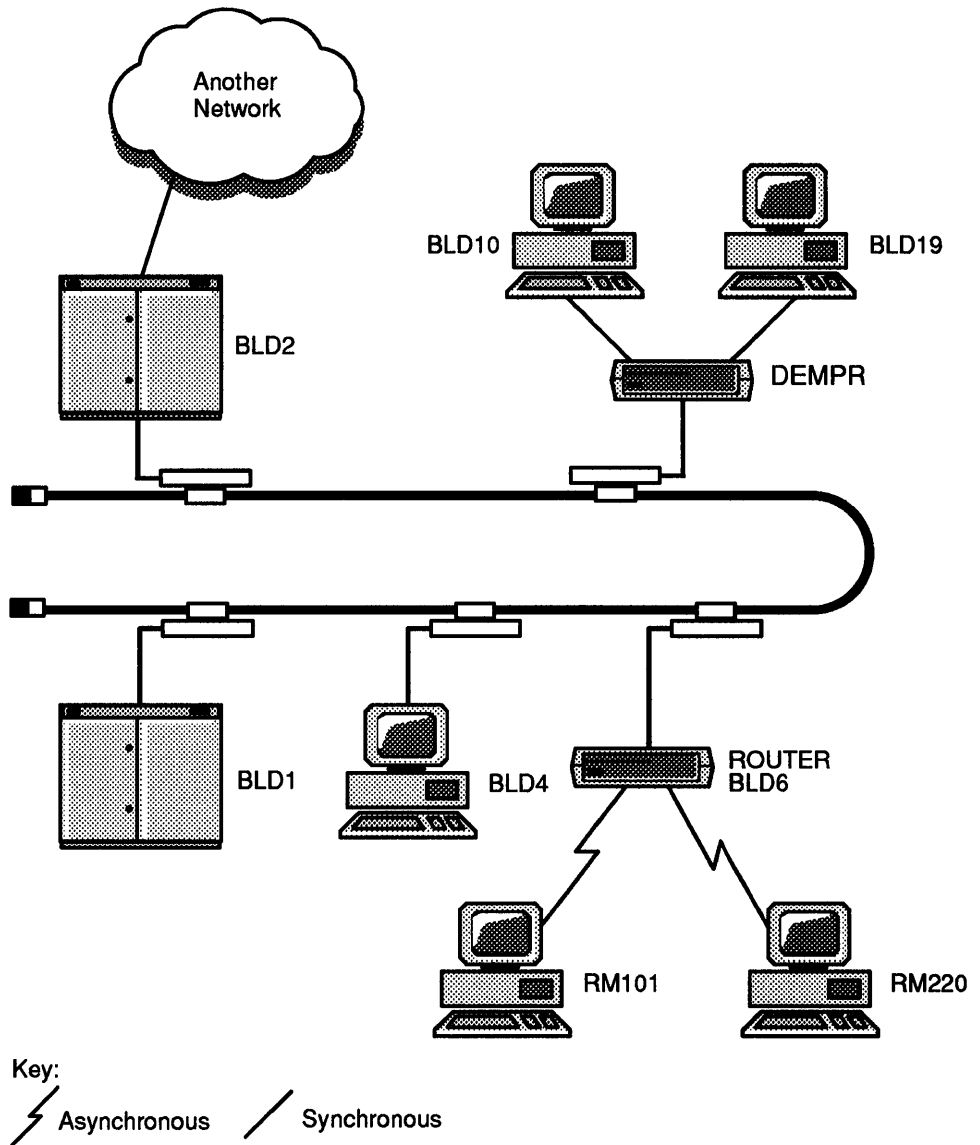
Throughout this guide, several terms are used to describe networks. The term **node** refers to any processor, intelligent terminal, or other computer system capable of functioning independently within a DECnet network environment. The term **DEMPR** refers to the Digital ThinWire™ Ethernet™ Multiport Repeater. Figure 1–1 displays different types of nodes in the context of a DECnet network.

- An **adjacent node** is physically connected to your node by a single communications line, such as BLD1 and BLD4 in Figure 1–1.
- An **end node**, such as BLD10, BLD19, BLD1, BLD4, RM101, or RM220, can receive information for its own use only; it cannot receive messages and subsequently route them to another node.
 - Your personal computer is an end node. An end node is connected to the network by a single line.
 - An end node may be physically connected to the adjacent routing node or directly to the Ethernet local area network (LAN).
- A **routing node**, such as the MicroVAX™ and DECrouter™ in Figure 1–1 (BLD2 and BLD6), can receive and forward information from one node in the DECnet network to another node or network.
- An **executor node** is the node that is executing the network management commands.
- A **local node** is the node on which you are physically working when you type in commands. For instance, if you are typing commands at the node RM101, RM101 is your local node.
- A **remote node** is any node in the network other than your local node. For example, if your local node is RM101, all other nodes in the sample network are remote nodes.

1.2 What Is DECnet–DOS?

DECnet–DOS consists of a set of software components or modules that work with the DOS operating system to enable your personal computer to participate in a DECnet network. Table 1–1 lists some of the tasks you can accomplish using DECnet–DOS utilities.

Figure 1-1 A Sample DECnet Network



LKG-4481-89I

Table 1–1 What You Can Do with DECnet–DOS

If You Want to	Use This Utility
Emulate VT320™, VT220™, VT100™, VT52™, terminals to directly access a host system	Network Virtual Terminal (SETHOST). See the PATHWORKS for DOS, <i>SETHOST Terminal Emulation Guide</i> for directions on how to use the SETHOST utility.
Test and maintain your network configuration	Network Control Program (NCP). See the PATHWORKS for DOS, <i>DECnet Network Management Guide</i> for more information on using NCP.
Enable another system to monitor your personal computer	Network Management Listener (NML). Refer to PATHWORKS for DOS, <i>DECnet Network Management Guide</i> for more information on how to use NML.
Exchange files with other users	File Access Listener (FAL), Network File Transfer (NFT), and Transparent File Access (TFA). See Chapters 3 and 4 of this manual for more details on NFT and FAL. Refer to PATHWORKS for DOS, <i>DECnet Programmer's Reference Manual</i> for TFA.
Share remote disks and printers	Network Device Utility (NDU). See Chapter 5 of this manual.
Allow automatic remote access to your local node	Job Spawner (SPAWNER). File Access Listener (FAL). See Chapters 2 and 3 of this manual.
Create applications that communicate with applications on remote nodes	Refer to the PATHWORKS for DOS, <i>DECnet Programmer's Reference Manual</i> for information on DECnet programming interfaces.

1.3 Managing and Monitoring Your Network Software

The Network Control Program (NCP) lets you manage your network software. You can use NCP to:

- Set up your personal computer as a DECnet end node.
- Maintain and update your node's DECnet databases.
- Monitor your node's operation in the network.
- Diagnose and solve communication problems.

Refer to the PATHWORKS for DOS, *DECnet Network Management Guide* for more information about when and how to use NCP commands.

1.4 Programming Task-to-Task Communications

DECnet-DOS offers programming interfaces that let you enhance the task-to-task communication capabilities of DECnet. You can create network task-to-task applications that suit your specific needs. Programmers, for example, can take advantage of the VMS™ operating system's computational power by passing problems to a subroutine on the VMS system, with results returned to the PC-based program.

DECnet-DOS provides a programming library of subroutines a programmer can use to write task-to-task applications. A program can access these subroutines through C or MACRO Assembly system calls. Your PATHWORKS for DOS kit provides source files you can use to build a programming library.

The DECnet-DOS programming interface makes it easy for programmers to create network programs. The interface you choose depends upon several factors, including:

- The level of control over the communications process that you want your program to have
- The programming language you want to use
- Your programming knowledge of DOS and of DECnet networking concepts

DECnet-DOS also provides a NETBIOS emulation interface for network applications. NETBIOS is an industry-standard session-layer interface developed by IBM for network applications that are written to use IBM's PC LAN program. With the DECnet-DOS NETBIOS emulation installed on your personal computer, your industry-standard NETBIOS applications can communicate over a DECnet network.

Refer to the PATHWORKS for DOS, *DECnet Programmer's Reference Manual* for programming information.

1.5 Using Microsoft Windows with DECnet-DOS Utilities

You can start any of the DECnet-DOS utilities from the Microsoft® Windows environment. Two of the utilities operate as fully functional windows applications. Seven of the utilities operate as standard applications. They work the same as if you started them from the command line. In standard mode, they run in a full-screen environment. In enhanced mode, they can run in a window or full-screen environment.

There are two utilities that can run in a windows environment:

- The File Access Listener.
- The Network File Transfer utility.

For more information about using these utilities from the command line, refer to Chapters 3 and 4 of this guide. For more information about running these utilities in a windows environment, refer to *PATHWORKS for DOS, Microsoft Windows Support*.

Spawning Multiple Services

This chapter describes the DECnet–DOS Job Spawner. The Job Spawner is a DECnet–DOS utility that runs as a foreground task while waiting for incoming connection requests from other nodes in the network. With the Job Spawner utility running, your personal computer acts as a server, performing multiple service functions for remote systems.

This chapter covers the following topics:

- An overview of the Job Spawner (see Section 2.1).
- Creation of the Job Spawner database file (see Section 2.2).
- How to use the Job Spawner (see Section 2.3).

2.1 Overview of the Job Spawner

The Job Spawner supports the File Access Listener (FAL), the DECnet Test Receiver (DTR), and user-written servers. By using the Job Spawner, your personal computer can process different connection requests at different times without any need for user intervention. When a connection request arrives, the Job Spawner looks up the requested object name or number in its database (DECOBJ.DAT), then runs the program that services that type of request. When the program finishes, the Job Spawner resumes control and again waits for incoming connection requests.

As an example, if the Job Spawner runs and it detects a request for file transfer, the Job Spawner initiates the FAL utility and causes FAL to run. FAL continues to run until the requested activity is complete. The Job Spawner then continues to listen for other requests. (The Job Spawner cannot run two service programs simultaneously.)

The Job Spawner accesses a database file to determine the programs (or objects) that run for each kind of connection that the Job Spawner detects. DECOBJ.DAT is the object database that the the Job Spawner uses.

The installation procedure creates the DECOBJ.DAT database. However, if you installed DECnet-DOS manually, you might not have created this database. You must use the Network Control Program (NCP) command, `DEFINE OBJECT`, to create the DECOBJ.DAT database. You can use the NCP `DEFINE OBJECT`, `LIST OBJECT`, and `PURGE OBJECT` commands to change, review, or delete the objects in your database file. Refer to the *PATHWORKS for DOS, DECnet Network Management Guide* for more information about using NCP commands.

2.2 Creating a Job Spawner Database File

To create a DECOBJ.DAT file, first start the NCP utility.

```
C:\>NCP 
```

```
NCP>
```

To define the objects, use the `DEFINE` command, which has the following format:

```
DEFINE OBJECT obj-nam NUMBER n FILE file-nam [ARGUMENTS x]
```

Where:

<i>obj-nam</i>	Is the name of the DECnet object. The object name can have a maximum length of 16 characters.
<i>n</i>	Is the number of the DECnet object. The range for this number is 0 to 255.
<i>file-nam</i>	Is the name of a command file or program to be started by the Spawner. Command file names must end with the .BAT extension. The file name can include a path specification.
<i>x</i>	Specifies the command line arguments for the program specified with FILE. To store argument information in uppercase letters, type the characters: abc is stored in the database as ABC . To store argument information exactly as typed, place the string in double quotation marks: "Aab" is stored in the database as Aab .

Digital Equipment Corporation reserves the numbers 1–127 for Digital-supplied objects. Each number is predefined for a network program. The object numbers 128–255 are available for user-written or user-supplied applications. Table 2–1 lists the DECnet–DOS predefined object numbers:

Table 2–1 Predefined Object Numbers

Object #	Object Name	Object File Name
0	Any object	Any object file name
17	FAL	FAL.EXE
63	DTR	DTR.EXE
128–255	Any user-written/ user-provided network object	Any user-defined name

For example, to define FAL as object 17 with an argument that includes the command file T.CMD, use the following command:

```
NCP>DEFINE OBJECT FAL NUMBER 17 FILE FAL.EXE ARGUMENTS "/LOG:T.CMD
/ASCII" 
```

Use the LIST command to display the objects you have defined. For example:

```
NCP>LIST KNOWN OBJECTS 
```

DECnet Objects

#	Taskname	File	"Arguments"
17	FAL	FAL	"/LOG:T.CMD /ASCII"
63	DTR	DTR	" "

```
NCP>
```

You can also use the PURGE OBJECT command to delete one object or all the objects in the database. For more information about PURGE and other NCP commands, refer to the PATHWORKS for DOS, *DECnet Network Management Guide*.

The object number 0 has a special meaning. An unlimited number of objects can be defined as object 0. When DECnet receives a connection request of object number 0, it makes the connection based on object name.

To write your own server programs, note that the server program is run by the Job Spawner as if the following command were typed:

```
PROGRAM -use n
```

Where:

The variable *n* is the number of a socket on which an incoming connect request has been accepted by the Job Spawner in deferred mode. The server program is responsible for accepting or rejecting incoming connection requests. The server program is also responsible for closing all sockets used for data communications. This includes the socket that Job Spawner passed to the server program.

2.3 Using the Job Spawner

To start the Job Spawner, simply type SPAWNER at the system prompt and press **Return**. To exit from the Job Spawner, type an exclamation mark (!).

You can also start the Job Spawner from a window, if you are using Microsoft Windows. The Job Spawner runs under Microsoft Windows as a standard application. (There is a SPAWNER.PIF file included in your kit that allows you to run the Job Spawner as a standard application.)

Format

```
SPAWNER [ /LOG  
         /HELP ]
```

Where:

/LOG	Logs messages about Spawner and incoming connections to the file SPAWNER.LOG in the DECnet directory.
/HELP	Prints help information on the screen.

Example 2-1 illustrates the Spawner running and receiving an incoming connect request for FAL:

Example 2-1 Spawner Receiving a Request

```
C:\>SPAWNER Return
press '!' to abort
SPAWNER (V4.0) listening... on Wed Jul 15 1990 at 11:22:46
Connect request from node RM204 for object #17 name fal on Wed Jul 15 1990
at 11:26:57
Executing: FAL -use 2... on Wed Jul 15 1990 at 11:26:57
          FAL - File Access Listener - Version 4.0

Network Driver Version 4.0
Current working directory:C:\DECNET
Files will be sent as either ASCII or binary.
Existing files will be overwritten.
No access checking will be done (world has read/write privileges).
FAL running...
DIRECTORY access from RM204 for LOCAL"::C:\DECNET\*.EXE;*
SPAWNER (Version 4.0) listening... on Wed Jul 15 1990 at 11:27:01
SPAWNER exiting... on Wed Jul 15 1990 at 11:34:22
```

DECnet Objects

#	Taskname	File	"Arguments"
0	SHOWME	C:\SHOWME.BAT	"a1 a2 a3"
17	FAL		"/l:fal.log /a"
63	DTR		
129		C:\BATCH.BAT	"a1 a2"

The command line argument for FAL in this example is "-use 2." This means that the incoming connect request is pending on socket number 2 (in accept deferred mode).

In the previous example, several batch files are included that are associated with object names or object numbers. The Job Spawner executes each batch file as specified, passing the arguments that are contained in the DECOBJ.DAT database. The Job Spawner always attempts to close the socket that was created for the batch file's incoming connect request.

If you start the Job Spawner and it cannot locate the necessary database to use for answering a request, it displays the following help information:

```
Error: Could not open file C:\DECNET\DECOBJ.DAT
```

- . Abort job Spawner by pressing '!'
- . Job Spawner does not support multiple links.
- . Database file for Spawner in DECnet database path.
- . Job Spawner can spawn batch files.
- . Job Spawner can pass command line arguments to servers.
- . Name of database file is DECOBJ.DAT.
- . Use the DEFINE and LIST commands in NCP to configure DECOBJ.DAT.

If you start the Job Spawner and a connection request arrives at the node, the following message appears:

```
Connect request from node BLD3 for object #17 name FAL on Dec  
16, 1990 at 14:12:49.
```

This example shows that a user on node BLD3 was using NFT. FAL is responsible for handling remote NFT requests, so the Job Spawner runs a FAL process for the request.

Providing Remote File Access to Your Node

This chapter describes the File Access Listener (FAL) utility. The FAL utility provides a way for you to let other nodes access files on your node. The purpose of this utility is to listen for and receive remote file access requests from the network. Then, from a remote terminal or personal computer, you can use the Network File Transfer to copy files, list directories, and type files that reside on the personal computer.

This chapter covers the following topics:

- Starting FAL (see Section 3.1).
- Entering FAL as an object (see Section 3.2).
- Accessing files (see Section 3.3).
- Restrictions (see Section 3.4).
- The FAL syntax (see Section 3.5).

3.1 Starting the File Access Listener Utility from a Command Line

You must have the FAL utility running on your system before any exchanges of file data can take place. In addition, because DOS is a single-tasking operating system, you must run FAL to the exclusion of all other application programs, unless you use Microsoft Windows.

You can use FAL in three different ways:

1. Invoked from the command line
2. Spawned by the Job Spawner
3. Run as a fully functional Microsoft Windows application

Command Line

To start FAL from the command line, type the command at the system prompt. For example:

```
C:\>FAL 
```

FAL runs until you cancel it by pressing the "!" key. If you use , FAL does not finish the current task. Instead, it exits immediately and closes down all links with the remote node. If you use any other key, FAL first finishes its current task and then exits.

Job Spawner

If you invoke FAL through the Job Spawner, the Job Spawner runs in the foreground, waiting for incoming connection requests. When a connection request is made for a FAL connection, the Job Spawner dispatches FAL. FAL returns control to the Job Spawner when it completes its connection.

Microsoft Windows Application

Use the file FAL.EXE if you want to run FAL as a Microsoft Windows application. This image file contains both windows and nonwindows versions of FAL. If you are not running windows, use the file FALNOWIN.EXE to save disk space.

The Microsoft Windows version of FAL provides the same functions as command line FAL, but the appearance is very different. This chapter describes how to use FAL from the command line. For information on how to use FAL as a windows application, see PATHWORKS for DOS, *Microsoft Windows Support*.

3.2 Entering the File Access Listener Utility As an Object

If you ran the PATHWORKS for DOS configuration procedure to install DECnet-DOS, the procedure installs FAL as an object in the DECOBJ.DAT database. However, if you installed DECnet-DOS manually, you can define it as an object now.

To enter FAL as an object, invoke NCP and use the DEFINE OBJECT command.

Objects are run by the Job Spawner, as needed. The Job Spawner listens for connection requests from remote systems. For more information on the Job Spawner and the NCP utility, refer to Chapter 2.

Format

```
DEFINE OBJECT FAL NUMBER 17 FILE FAL.EXE [ARGUMENTS xxx]
```

Where:

xxx Specifies the command-line arguments for the program specified by *file-name*. You can use one to three arguments for each file.

For example, to define FAL as object #17 with an argument that includes the command file T.CMD, use the following command:

```
C:\>NCP DEFINE OBJECT FAL NUMBER 17 FILE FAL.EXE ARGUMENT "/LOG:T.CMD /ASCII"
```

3.3 How the File Access Listener Utility Accesses Files

When FAL detects a request to copy a file to or from your node, it first determines whether access checking has been enabled on your local system. (Access checking is enabled if the file DECACC.DAT exists in the DECnet database path.) NCP creates the file the first time you use the NCP DEFINE ACCESS command. For more information about using NCP and NCP commands, refer to the PATHWORKS for DOS, *DECnet Network Management Guide*. If you do not enable access checking, your system's files are subject to all incoming access requests while FAL is active.

Note

If you issue the NCP command PURGE KNOWN ACCESS, you clear the contents of the DECACC.DAT file. The DECACC.DAT file will no longer exist. **If this happens, FAL will not check access and will allow access to any user.**

If access checking is enabled, then FAL checks the type of access privilege contained in the incoming request. If the request has the proper privilege for access to your node, FAL sends or receives the file data to or from the requesting node.

MS-DOS systems do not retain file attributes with files. As a result, when a remote node asks FAL for the attributes of a file on the personal computer, it cannot determine what those attributes are.

3.4 Restrictions

This section describes some restrictions when working with the FAL utility on a DOS system.

- FAL does not perform file locking. If you are using FAL for simultaneous access to the same file, ensure that all users of that specific file open the file for read only. If two users attempt to open the same file for simultaneous write, unpredictable results can occur.
- The VMS operating system cannot use remote path specifications that are not surrounded by double quotation marks unless they conform to VMS syntax. Some restrictions apply when using file specifications enclosed in quotation marks with VMS. Also, VMS does not separate the parts of a file specification when it uses wildcards. You cannot make wildcard copies of files from a personal computer to VMS by using VMS DCL with DOS file syntax. To solve all these problems, FAL accepts VMS syntax for file specifications. For example:

```
COPY *.whatever MSDOS:."\ DIR\*.*" fails
COPY *.whatever MSDOS:.[DIR]*.* works
COPY MSDOS:.[WORK]*.*.* works
```

- The VMS COPY command cannot be used to copy binary files between VMS and MS-DOS systems. (DDIF, DTIF, and DOTS files are exceptions.) Instead, use the VMS EXCHANGE/NETWORK command to copy binary files.
- The symbol *\whatever* is sometimes a file and sometimes a directory. To avoid the problems that this causes, always specify file names or use VMS syntax or append **.** if it is a directory.

The following format returns information on the directory file itself:

```
$DIR MSDOS:."\SUBDIR"
```

The following format returns a directory listing of the specified subdirectory:

```
$DIR MSDOS: : "\SUBDIR\*.*"
```

The following format returns a directory listing of the specified subdirectory:

```
$DIR MSDOS: : [SUBDIR]
```

From ULTRIX use this format using single quotation marks:

```
$dls MSDOS: : '\SUBDIR\*.*'
```

3.5 The File Access Listener Utility Command Syntax

While FAL is running, file data can pass between your node and the requesting node. You can use switches to control the way FAL operates.

FORMAT

```
FAL [ /ASCII  
      /BINARY  
      /ERROR  
      /LOG:file-name ]
```

Where:

- | | |
|---------|--|
| /ASCII | Transfers all files as ASCII files. An ASCII file is a file whose records end with end-of-text or line-terminating characters, such as a carriage return/line feed (CR/LF) pair.
Binary files can lose data if they are copied in the ASCII format. |
| /BINARY | Transfers all files as binary files. A binary (or image) file is a file whose data is copied exactly as it appears with no interpretation of CR/LF characters.
Using this switch ensures that any file copied to a remote node and then back to your personal computer will not lose any data. However, ASCII files copied from the personal computer when the /BINARY switch is used might not be in the proper format for use on the remote node. |
| /ERROR | Reports an error to the remote node if the requesting node is attempting to overwrite an existing file. If you do not set this switch, an existing file is deleted and a new one created with the same name. The default is NO ERROR. |

/LOG:file-name Logs the type of access for every request it receives. The type of access includes the following information:

- The command or request such as an NFT DIRECTORY or DELETE command
- The requested directory name and file name
- The node name
- The access control information for that node user name and account

This information displays on the screen. If you do not want to see the information on the screen, you can redirect it to an output file when you first issue the FAL command.

file-name Names the file to use for storing the access information.

To use these switches, type them after the FAL command on the same line. Leave a space between the command and the switch. If you use more than one switch, be sure to separate them with a space.

Note

When you start FAL without either switch (/ASCII or /BINARY), it automatically checks each file. FAL looks for any CR/LF character sequence in the first 512 bytes of the file. If FAL finds CR/LF character sequence, it sends the file as ASCII.

Examples

1. The following command starts FAL, requesting it to report errors to the remote node that requested the transfer and then log all file transfer requests in the LOGFILE.TXT file:

```
C:\>FAL/ERROR/LOG:LOGFILE.TXT 
```

2. The following command enters FAL as an object in the DECOBJ.DAT database file:

```
NCP>DEFINE OBJECT FAL NUMBER 17 ARG "/ERROR/LOG:LOGFILE.TXT"
```

The Job Spawner runs this object if a connection request is made for a file transfer. When a remote node makes this request, FAL runs the same as in the previous example. The only difference is when the connection request completes, FAL exits until another remote node requests a file transfer.

Accessing Local and Remote Files

File transfer is a basic utility of networks. Without it you could not easily share files with other nodes on the network.

The Network File Transfer (NFT) utility is a network program that allows you to access remote files. NFT provides access to files on any DECnet system that is running a File Access Listener (FAL). NFT also supports numerous file attributes.

The NFT utility allows you to:

- Append two or more files
- Copy files between the local and remote nodes
- Delete local and remote files
- List files located in a local or remote directory
- Define and display access control information
- Run command files on remote nodes
- Display the contents of a local or remote file on your screen
- Print files on remote printers

This chapter describes:

- How to run NFT (see Section 4.1)
- Entering NFT commands (see Section 4.2)
- Using the HELP command (see Section 4.3)
- Exiting NFT (see Section 4.4)
- Defining remote node access information (see Section 4.5)
- Stating file specifications (see Section 4.6)
- Types of files and systems (see Section 4.7)

- Manipulating file operations (see Section 4.8)
- The NFT commands and their syntax (see Section 4.9)

You can use NFT in two different environments:

1. From the command line (in the PATHWORKS for DOS environment)
2. In a windows environment (if you are using Microsoft Windows)

4.1 Running the Network File Transfer Utility

This section describes the ways you can invoke the Network File Transfer utility.

4.1.1 Command Line Method

You perform NFT functions by using one of the following command line methods:

- Single command method:

```
C:\>NFT command   
C:\>
```

Note

When you run NFT in single command mode, an **ERRORLEVEL** of 0 indicates success. An **ERRORLEVEL** of 1 represents failure.

- Single command method using a command file:

A command file contains all the NFT commands that you want to use to complete a series of NFT operations. By using a command file, you only need to issue one command to process all other commands.

To use a command file, you must enter a left angle bracket (<) followed by the file name. For example:

```
C:\>NFT < COMMANDS.DAT   
C:\>
```

In this example, the **COMMANDS.DAT** file contains all of the NFT commands you want to process. The angle bracket is known as a redirect symbol.

- Multiple command method:

```
C:\>NFT   
NFT>command   
NFT>command   
NFT>EXIT   
C:\>
```

4.1.2 Microsoft Windows Environment

The NFT utility runs under Microsoft Windows as a fully functional windows application. Use the file NFT.EXE if you want to run NFT as a windows application. This file contains both windows and nonwindows versions of NFT. If you are not running windows and you want to save disk storage space, you should use the file NFTNOWIN.EXE.

The windows version of NFT provides the same functions as command line NFT, but the appearance is very different. This chapter describes how to use NFT from the command line. For information on how to use NFT as a windows application, see the PATHWORKS for DOS, *Microsoft Windows Support Manual*.

4.2 Entering the Network File Transfer Utility Commands

The NFT commands consist of three parts:

1. The command verb and switch
2. The source file specification
3. The destination file specification

Figure 4–1 shows the correct syntax to use.

A file specification can be local or remote. If the file is local, you need not include the node name in the file specification. If the file is remote, you must include the node name in the file specification.

Figure 4–1 Network File Transfer Command Example

```
NFT> COPY/PRINT FILE.TXT RM10::FILE1TXT ;1 Return
```

Verb and Switch	Source File Specification	Destination File Specification
--------------------	------------------------------	-----------------------------------

LKG-4377-901

You must separate each part of the command with a space or tab. Notice in Figure 4–1 that the command verb and switch make up one part of the command line and so need not be separated. The node name is part of the file specification and should not be separated from the rest of the file specification.

You can abbreviate command verbs and switches to a character string, of three or more characters that is unique to that verb. For example, you can type **SHO** for **SHOW**. For clarity and consistency, all examples in this manual use the full command format.

If you need help with a command or switch, remember to use the **HELP** command (see Section 4.3).

4.2.1 Command Prompts

In many cases, if you type a command verb alone, NFT prompts you for further information. For example, if you type the command APPEND without qualifiers, NFT prompts you for the required information:

```
NFT>APPEND 
File(s)? BLD3"SMTH OPN"::WRT:[SMTH]FLB.TXT;3 
To? FLA.TXT 
Appending file:BLD3"SMTH"::WRT:[SMTH]FLB.TXT;3 to FLA.TXT[1322 bytes at 3626 bytes/second]
NFT>
```

4.2.2 Command Switches

Some operating systems store files with attributes that indicate the type of file and the format of the file's records. A DOS file does not include attributes. Therefore, when you copy a file from the local system to another operating system that stores attributes (such as VMS), you can use switches to indicate the type of file you are copying. NFT provides defaults for file attributes. You can use switches to change these defaults.

You can also use switches to modify command verbs. Switches are global in effect and must immediately follow the command verb in the command line. The same switch can have a different effect, depending on whether you are copying a file to or from a remote node.

A valid switch for several NFT commands is /NOLOG. This switch requests that NFT not print a notification line after the APPEND, COPY, DELETE, PRINT, and SUBMIT operations. It is ignored for all other commands.

If you use a switch that NFT does not accept, such as /PRINT with a remote input file, NFT displays an error message.

4.3 Using the HELP Command

If you need assistance in selecting NFT commands and switches, use the **HELP** command. Type:

```
NFT>HELP 
```

The system responds with:

Help is available on the following commands:

APPEND	COPY	DELETE	DIRECTORY
EXIT	HELP	PRINT	SET
SHOW	SUBMIT	TYPE	

switches:

/ALLOCATION	/ASCII	/BLOCK	/BRIEF
/CC	/DELETE	/FIXED	/FULL
/IMAGE	/LSA	/MACY11	/MRS
/NOCONVERT	/NOLOG	/NOSPAN	/PRINT
/STREAM	/STREAM_CR	/STREAM_LF	/SUBMIT
/UNDEFINED	/VARIABLE	/VFC	

General command format is:
command/switches file file

Please note: Switches **MUST** always immediately follow the command.
Wildcards and file lists are supported for most commands.

```
NFT>
```

To obtain information about the **SHOW** command, for example, type:

```
NFT>HELP SHOW 
```

```
SHOW
```

The **SHOW** command displays defaults for remote file access.

Example:

```
NFT> SHOW node::
```

```
NFT>
```

To obtain information about the **/ASCII** switch, type:

```
NFT>HELP /ASCII 
```

```
/ASCII
```

The **ASCII** switch indicates that a file contains ASCII text.
If neither the **ASCII** nor the **IMAGE** switch is used, **NFT** attempts to determine the file type automatically.

This switch is valid only with the **COPY** and **APPEND** commands.

4.4 How to Exit the Network File Transfer Utility

You can use `Ctrl/Z` `Return` to exit from NFT, or you can type EXIT followed by `Return` at the NFT prompt.

4.5 Defining Remote Node Access Information

Access control information is security information that allows you to access a specified remote node with the privileges of a specific user. This information includes:

- **User name**
A character string consisting of 1 to 39 alphanumeric characters that identifies the user at the remote node.
- **Password**
A character string consisting of 1 to 39 alphanumeric characters that identifies the user's password.
- **Account**
A character string consisting of 1 to 39 alphanumeric characters that identifies the user's account.

You can follow the node name or node address (*area.node*) with the user name, password, and account enclosed in quotation marks (" "). You must separate each field with a space. You can then follow the entire string with a double colon (::). This example shows a node name with access control information:

```
BLD3"SMITH OPEN"::
```

For extra security, you can eliminate the password from the string and have NFT prompt you for it. When you enter the password at the prompt, NFT does not echo the password on the screen. Because your password does not display on the screen, you have control over who gets to see it. This example eliminates the password:

```
BLD3"SMITH":: Return  
Password for BLD3/SMITH?
```


4.5.1 Defining Access Control Information with Network File Transfer

There are two ways to specify access control information during an NFT operation:

1. By specifying access control information on the command line
2. By using the SET command

The NFT utility temporarily stores access control information for up to ten nodes.

In the following example, you access a remote node by requesting a directory listing of files located on the VMS node BLD3, user name SMITH, password OPEN:

```
NFT>DIRECTORY BLD3"SMITH OPEN":: [Return]
```

The NFT utility displays the list of files located in the requested directory. This includes the block size of each file, and the time and date the file was last modified or created. For example:

```
Directory of: BLD3"SMITH password"::SYS$SYSROOT:[SMITH]
```

```
FILE1.TXT;1      12      01-MAR-85  16:01:51
FILE2.TXT;1      34      02-SEP-83  14:20:35
FILE3.TXT;1      22      09-JUL-77  12:15:22
NFT>
```

Note

The password information is shown in output as "password" to protect it from casual observation.

The NFT utility stores the access control information for node BLD3 in its temporary table.

You use the SET command to specify and then save a user name, password and account, as well as specific disk and/or directory information. In the following example, you specify the user name and password for node BLD12. In addition, you specify the name of the subdirectory ARENA.FILES.

```
NFT>SET BLD12"ARENA BULL"::[ARENA.FILES] [Return]
```

The NFT utility stores the access information for node BLD12 in its temporary table. When you specify access information during an NFT operation and you are using the multiple command method, NFT temporarily stores the information for use in any following command dealing with the same node.

The NFT utility deletes access information when you exit from it. However, if you saved the access information using NCP, it is not deleted.

If you often access a certain node and account, you can specify default access control information using the NCP command `DEFINE NODE`. For more information about NCP commands, refer to the *PATHWORKS for DOS, DECnet Network Management Guide*.

4.5.2 Using Default Access Control Information

After you specify access control information by using either method, you can use the node name alone in subsequent file operations. NFT uses the access control information you last specified for the node name. This is the **default** access control information.

For example, if you request a directory listing of node BLD3 by using only the node name, NFT displays the same list of files as the first time you accessed the node. When you type a node name alone, you must follow the name with a double colon. For example:

```
NFT>DIRECTORY BLD3:: 
Directory of: BLD3"SMITH password": :SYS$SYSROOT: [SMITH]

FILE1.TXT;1          12    01-MAR-85  16:01:51
FILE2.TXT;1          34    02-SEP-83  14:20:35
FILE3.TXT;1          22    09-JUL-77  12:15:22
NFT>
```

When you type a node name alone, NFT follows this procedure to determine the access control information:

1. First, NFT checks to determine if you specified the access control information earlier in the same NFT session. If you did, NFT uses that access control information.
2. If you did not specify access control information earlier in the same session, NFT checks to determine if you specified it using NCP. If you did, NFT uses that access control information.
3. If you did not specify access control information, either using NCP or earlier in the same NFT session, NFT assumes that the remote node does not require the information and attempts the current operation.

4.5.3 Changing the Default Access Control Information

You can change the default access control information by typing different access control information for the same node name in an NFT operation. In this case, NFT replaces the original access control information in its temporary table with the new information.

For example, if you request a directory listing of files on remote node BLD3, but specify a different user name and password, NFT displays a list of files for the new user:

```
NFT>DIRECTORY BLD3"DOC PEN":: 
Directory of: BLD3"DOC password"::SYS$SYSROOT:[DOC]

ABC.TXT;1          10      07-SEP-85  12:10:49
DEF.TXT;1          29      12-JAN-86  15:20:32
GHI.TXT;1          47      03-MAR-86  14:10:05
NFT>
```

Now if you request a directory listing using the node name alone, NFT displays this list of files associated with the most recent access control information for the node BLD3.

```
NFT>DIRECTORY BLD3:: 
Directory of: BLD3"DOC password"::SYS$SYSROOT:[DOC]

ABC.TXT;1          10      07-SEP-85  12:10:49
DEF.TXT;1          29      12-JAN-86  15:20:32
GHI.TXT;1          47      03-MAR-86  14:10:05
NFT>
```

You can also change the access control information by using the SET command. In fact, when you use the SET command to specify default access control information, you can replace only the information in the NFT table by using another SET command. For example, to change the access control information for node BLD12, type:

```
NFT>SET BLD12"SMITH DANCE":: 
```

Now if you request a directory listing by using the node name alone, NFT displays the new list of files associated with the most recent access control information for the node BLD12.

```
NFT>DIRECTORY/BRIEF BLD12:: 
Directory of: BLD12"SMITH password"::SYS$SYSROOT:[SMITH]

LAW.DOC;1          RULE.DOC;2
NFT>
```

4.5.4 Displaying Access Control Information

The **SHOW** command displays access control information that you specify during an NBT session. For example:

```
NBT>SHOW   
BLD3"DOC password"::  
BLD12"SMITH password"::
```

4.6 Using File Specifications in DECnet-DOS

A complete file name is called a **file specification**. A file specification provides the computer system with all the information it requires to identify a unique file. Each operating system in the network has its own set of rules for naming files. Refer to Appendix A for more information about file specifications for other operating systems. For example, when you name a local DOS file, you must follow the standard DOS operating system format:

- A **drive name**. (This is optional if you are using the default drive name.)
- A **path name**. (This is optional.)
- A **file name** of up to eight alphanumeric characters.
- A **file type** of up to three alphanumeric characters, separated from the file name by a period (this is optional).

The following example specifies a file named SURVEY.CRD. The file is located on drive A in the subdirectory SUE, which is located in the subdirectory USERS:

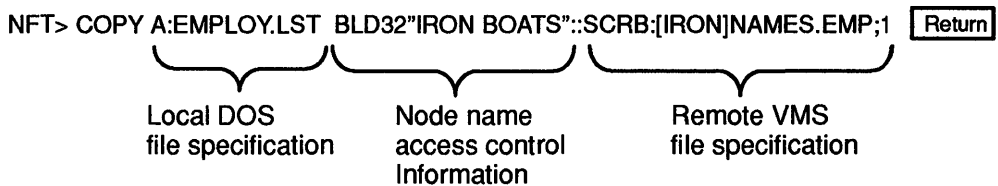
```
A:\USERS\SUE\SURVEY.CRD
```

When you access a file on a remote node, you must use a file specification that conforms to the conventions that a remote node requires. For example, some operating systems accept a version number as part of the file specification:

```
BLD3::CHAP1.DOC;2
```

The example in Figure 4-2 shows how to copy a local DOS file called EMPLOY.LST to a remote VMS node named BLD32. The file is stored on the remote node as NAMES.EMP;1 (version 1).

Figure 4–2 Copying a Local File to a Remote Node



LKG-4539-901

4.7 Types of Files and Systems

There are two types of data files:

- **Image** - A file whose data is copied without any interpretation or data change. The file is copied and received as a copy of itself. The most common example of this type of file is an executable file. An image file is the same as a binary file.
- **ASCII** - A file whose records end with a carriage return/line feed (CR/LF) pair.

The algorithm that NFT uses to determine whether a file is image or ASCII (when copying the file from your personal computer to a remote node without switches) is the following: if the first 512 bytes contain a carriage return/line feed (CR/LF character sequence), the file is ASCII. If the CR/LF character sequence does not appear in the first 512 bytes, the file is an image. When you use this rule, some image files might appear as ASCII files.

File systems can be one of two types:

- **Stream**
With a stream system, a file is a series of continuous characters. Most stream systems, including the DOS operating system, do not support such attributes as fixed- or variable-length records. If a file with these attributes is copied to the local system, the attributes are lost.
- **Nonstream**
With a nonstream system, file data is in specific record formats. Some examples of these formats are fixed length, variable length, and variable with fixed-length control (VFC).

Fixed-length records are all the same size. The size is fixed when you create the file, and you cannot change it.

Variable-length records can be of different lengths, up to a maximum size that you specify. The maximum size is fixed when you create the file and you cannot change it.

VFC records include a fixed-length control field that precedes the variable length data. This format allows you to add data that labels the contents of the variable length portion of the record.

Refer to Table 4–1 for the changes that occur when you copy a file from a remote node.

Table 4–1 Copying Files from a Remote Node

Data Type	File System Type	Record Attributes	How Stored
IMAGE	Not applicable	Ignored	As received from the remote node
ASCII	Stream	Ignored	As received from the remote node with embedded carriage control
ASCII	Nonstream	Other than implied CR/LF, PRN, or FTN	As received from the remote node
ASCII	Nonstream	Implied CR/LF pair	CR/LF added to each record
ASCII	Nonstream	PRN or FTN	Data converted correctly
DDIF	Any	N/A	Block mode
DTIF	Any	N/A	Block mode
DOTS	Any	N/A	Block mode

The NFT utility converts embedded carriage control characters by default. You can cancel this conversion with the `/NOCONVERT` switch.

Although NFT can usually determine whether a file is ASCII, you should use the `/IMAGE` switch when you copy a binary file to a remote node. Refer to Table 4–2 for the changes that occur when you copy a file to a remote node.

Table 4–2 Copying Files to a Remote Node

Data Type	File System Type	File Attributes	How Sent
ASCII	Stream	None	Unchanged. Records are determined by LFs.
ASCII	Nonstream	Variable, implied CRLF	Carriage return/line feed pair is dropped. Records are determined by LFs.
IMAGE	Stream	None	Unchanged.
IMAGE	Nonstream	Fixed: 128 bytes	Unchanged.
DDIF	Any	DDIF	Block mode.
DTIF	Any	DTIF	Block mode.
DOTS	Any	DOTS	Block mode.

4.8 Manipulating Files with the Network File Transfer Utility

You can use the NFT utility to manipulate files on accessible DECnet nodes.

You can include lists of up to ten file specifications in the APPEND, COPY, and TYPE commands. For example, you can append one to ten input files to a single output file. When you specify a list of files, the following rules apply:

- You cannot include more than 10 file specifications in a list.
- You cannot use wildcards (asterisk or question mark) in a list of files.
- You must separate items in the list with commas.
- When you enter a command line that contains more than one input file specification, these commands use temporary defaults. Temporary defaults are used for:
 - Node name
 - Device name
 - Directory name
 - File name and file type

If a file specification includes a node, device, or directory name, these names are applied to subsequent file specifications within the list. For example, the following command copies three test files from directory ADAMS on the device DBA1.

```
NFT>COPY RM2::DBA1:[ADAMS]TEST1.DAT,TEST2.DAT,TEST3.DAT Return
```

4.8.1 Appending Files

To copy one or more files to the end of an existing file, use the APPEND command. You can append the following combinations of files:

- One or more local files to a remote file
- One or more remote files to a local file

Note

You cannot append a local file to a local file, or a remote file to a remote file.

For example, to append the remote file B.TXT, located on the VMS node BLD3, to the local file A.TXT, type:

```
NFT>APPEND BLD3"SMITH OPEN": :WRIT:[SMITH]B.TXT;3 A.TXT 
```

The two files are now:

- The local file, A.TXT, which includes the contents of A.TXT and B.TXT.
- The remote file, B.TXT;3, which is the original copy of B.TXT on node BLD3.

If you are appending more than one file to the end of another file, separate the file specifications with a comma. For example:

```
NFT>APPEND BLD3"SMITH OPEN": :WRIT:[SMITH]B.TXT;3,C.TXT;2 A.TXT 
```

4.8.2 Copying Files

To copy files between your node and a remote node, use the COPY command.

When you copy a remote file to the local node, you must use the correct remote file specification. The following example shows how to copy a remote file named NEWS.DOC;10 from a VMS node called BLD3 to the local node. The remote file is located in the directory SMITH on device WRIT. The file is given the new name FLASH.DOC when it is copied to the local node.

```
NFT>COPY BLD3"SMITH OPEN": :WRIT:[SMITH]NEWS.DOC;10 FLASH.DOC 
```

You can copy a file from a remote node to a local node without specifying a local file name. For example:

```
NFT>COPY BLD3"SMITH OPEN": :WRIT:[SMITH]NEWS.DOC;10 
```

By default, the local file is named NEWS.DOC, the same as the original file.

When you copy a local file to a remote node, you must specify the local file name and the remote node name. For example:

```
NFT>COPY FILEA.TXT BLD3::
```

You can also specify the remote file name in a format supported by the remote operating system. For example, to copy a DOS file called FILEA.TXT from drive B on the local node to directory SMITH on the BLD3 node, type:

```
NFT>COPY B:FILEA.TXT BLD3"SMITH OPEN"::WRIT:[SMITH]FILEA.TXT;3 
```

The NFT utility truncates long file names (to reduce file name size to eight characters and file type size to three characters) when copying files to your personal computer.

4.8.3 Deleting Files

To delete a file or a group of files, use the DELETE command. The following example deletes all versions of a file named TEST1.TST from a remote VMS node named RM104.

```
NFT>DELETE RM104::TEST1.TST;* 
```

To delete only one file, enter the name and the type for the file. If there is only one version of that file on the remote node, NFT deletes it. You do not need to include the version number. If there are multiple versions of the same file, NFT deletes only the latest (or highest numbered) version of the file.

To delete a file that is not the latest version of that file, you must indicate the specific version number you want to delete. If the files FINDER.TXT;3 and FINDER.TXT;4 exist in your directory, you can delete FINDER.TXT;3, as follows:

```
NFT>DELETE RM102::FINDER.TXT;3 
```

This command deletes version 3 of the file FINDER.TXT on the remote node RM102.

4.8.4 Displaying Directory Information

To display a list of local or remote file specifications on your screen, use the DIRECTORY command. The names are displayed in the format used by the specified node. For example, to list file specifications from a directory on the remote VMS node BLD3, type:

```
NFT>DIRECTORY/BRIEF BLD3:: [Return]
Directory of: BLD3 "SMITH password"::SYS$SYSROOT: [SMITH]
APPXA.DOC;3      CHAP_1.DOC;2      CHAP_2.DOC;13     DATA.DAT;9
NEWS.DOC;10     MEMO.TXT;1       TEST2.TST;6       TEST3.TST;2
TEXT.DOC;8
```

You can either list a single file specification or use wildcards to specify a group of file specifications. The following example lists the group of files with a file type of TST:

```
NFT>DIRECTORY/BRIEF BLD3::*TST [Return]
Directory of: BLD3 "SMITH password"::SYS$SYSROOT: [SMITH]
TEST2.TST;6     TEST3.TST;2
```

4.8.5 Printing Files

To print a local file on a remote printer, use the /PRINT switch with the COPY or APPEND command. For example, to copy FILEA.TXT from the local node to the remote node BLD3 and then print the file at the remote node, type:

```
NFT>COPY/PRINT FILEA.TXT BLD3::FILEA.TXT;1 [Return]
```

Be aware that you cannot print a remote file on the local printer by using the /PRINT switch. If you try, the file is copied to the local node, and NFT displays the following warning message:

```
Warning: Cannot print files on local printer.
```

However, you can print a remote file on the local printer by copying it to the printer device.

To print a remote file on a remote printer, use the PRINT command. You must specify the file name that exists on the remote node. For more information about the PRINT commands, refer to Section 4.9.

4.8.6 Displaying the Contents of a File

To display the contents of a file on your screen, use the TYPE command. For example:

```
NFT>TYPE RM6::PS:[SMITH]FILE1.DOC;2 [Return]
```

4.8.7 Running Command Files

A command file contains a list of command strings. To execute the commands within a remote command file on the remote node, use the NFT SUBMIT command followed by the name of the remote command file. By typing commands in one file, you can run the file many times without retyping the commands.

You can also use the /SUBMIT switch to execute commands in a local file on a remote node. This switch is valid only with the APPEND and COPY commands. When you use /SUBMIT with either of these commands, the copied file is queued to the remote node's batch system (after the copy operation is complete).

For example, the following commands are in a command file called DUMP.CTL.13 on the VMS remote node RM6. The file is located on the device PS: in the directory JONES.

```
@PRINT PS:[SMITH]FILE1.DOC;2
@PRINT PS:[SMITH]*.BAK
```

This command file:

1. Prints the file called FILE1.DOC;2 located in the SMITH directory.
2. Prints all files with a file type of .BAK in the same directory.

To run this command file, type:

```
NFT>SUBMIT RM6::PS:[JONES]DUMP.CTL.13 
```

The file is placed in the batch queue on node RM6.

4.8.8 Logging File Operations

When you enter an NFT copy command (such as COPY or PRINT), NFT displays a notification that it has opened the specified files. In the following example, NFT logs the copy operation and notifies you as it opens the specified files:

```
NFT>COPY INTRO.DOC BLD3"SMITH OPEN": 
Copying file: INTRO.DOC to BLD3"SMITH password":DISK01:[SMITH]INTR
O.DOC;1 [148 bytes at 9289 bytes/second]
NFT>
```

There are two parts to the notification line:

- The first part includes all the text up to the record count in brackets. The NFT utility displays the notification line as soon as it successfully opens both the remote and local files.

- The second part is the byte count (x bytes at x bytes per second) at the end of the line. The NFT utility displays the byte count when it completes the copy operation.

You can use the /NOLOG switch to suppress the logging message for the APPEND, COPY, DELETE, PRINT, and SUBMIT commands.

4.9 Network File Transfer Utility Command Summary

The NFT utility provides you with command verbs that allow you to manipulate files located on local and remote nodes. Table 4–3 lists each NFT command and its function.

Table 4–3 Network File Transfer Commands

Command	Function
APPEND	Appends files from the local node to an existing file on the remote node, or copies and appends files from the remote node to an existing file on the local node.
COPY	Copies files from the local node to the remote node or from the remote node to the local node.
DELETE	Deletes a local or remote file.
DIRECTORY	Lists files located in a specified local or remote directory.
EXIT	Exits from an NFT operation and returns control to the DOS operating system. (You can also press Ctrl/Z Return to exit.)
HELP	Displays information about NFT commands and switches.
PRINT	Allows you to queue a file that exists on a remote node to be printed at that remote node.
SET	Allows you to set remote file access defaults for the current NFT session.
SHOW	Displays access control information for remote files.
SUBMIT	Submits a command file to run on a remote node.
TYPE	Displays the contents of a local or remote file on the screen.

The remainder of the chapter discusses each command in alphabetical order.

APPEND

APPEND

The APPEND command adds the contents of one or more input files to the end of an existing output file. You can append either ASCII or binary files to or from remote nodes. You can append the following combination of files:

- One or more local files to a remote file.
- One or more remote files to a local file.

You cannot append a local file to a local file, or a remote file to a remote file.

Depending on its position in a command, a file specification is either input (source) or output (destination).

Format

```
APPEND [ /ASCII:src-charset:dest-charset ]  
        /DELETE  
        /IMAGE  
        /NOLOG  
        /PRINT  
        /STREAM  
        /SUBMIT  
        /UNDEFINED ] input-file output-file
```

Where:

/ASCII:
src-charset:dest-charset

Indicates that the file has records that end with a carriage return/line feed pair.

You can also use the ASCII switch to perform character set conversion:

/ASCII:src-charset:dest-charset

The character sets are optional and you may also specify just the first character set. For more information on character set files see the *PATHWORKS for DOS, SETHOST Terminal Emulation Guide*.

Where:

src-charset and *dest-charset* are one of the following character set values:

DUTCH	FINNISH	FRCAN
FRENCH	GERMAN	ISO
ITALIAN	MCS	NORDAN
PORTUGUESE	SPANISH	SWEDISH
SWISS	UK	US

If you supply only one character set, it applies to the remote system. The local system uses the current character set.

- /DDIF** Tells the remote system that the file is a DDIF file. (Files created by DECwrite are of this type.) You rarely need to use this switch because NFT can detect DDIF files.
- /DELETE** Requests that NFT delete the copied file from the remote node after the file is copied and printed. This switch is valid only with the /PRINT switch and only when appending files from a remote node.
- /DOTS** Tells the remote system that the file is a DOTS file. You rarely need to use this switch because NFT can detect DOTS files.
- /DTIF** Tells the remote system that the file is a DTIF file. You rarely need to use this switch because NFT can detect DTIF files.
- /IMAGE** Requests that NFT copy the file to the remote system as it is (binary data) with no conversion of any kind. The default record format is **FIXED**, and the default maximum record size (MRS) is 128 bytes. You can change the defaults with the **VARIABLE** (or **/VFC**) and **/MRS** switches. The last record can be shorter than the previous records.
- /NOLOG** Requests that NFT not print a notification line after the operation is complete.
- /PRINT** Allows you to print a file on the remote node's default printer after the file copy is complete.

APPEND

<code>/STREAM</code>	Notifies the receiving system that the stream file sent has records terminated with a CR/LF character sequence.
<code>/STREAM_CR</code>	Notifies the receiving system that you sent a stream file whose records terminate with a CR character.
<code>/STREAM_LF</code>	Notifies the receiving system that you sent a stream file whose records terminate with a LF character.
<code>/SUBMIT</code>	Allows you to queue a command file to execute at the remote node after NFT completes the file copy.
<code>/UNDEFINED</code>	Indicates that the file has an undefined record format.
<i>input-file</i>	Specifies one or more input files to copy. The file name consists of up to eight characters. The file type consists of up to three characters. If you specify multiple input files, you must insert a comma between the file specifications. NFT then appends the multiple files to the output file.
<i>output-file</i>	Specifies the file to which the file(s) are to be appended.

A file specification has one of two forms. In its longest form, it consists of a drive name, directory path, file name, and file type. In its shortest form, it consists of a file name and file type.

If you append more than one input file to create a single output file, the attributes of the output file are determined by the attributes of the first input file that you specify in the command. If the attributes of the input files differ, the append operation seems to succeed, but the output file might be incorrect.

Examples

1. The following command appends the local file `FILE1.COM` to the remote file `FILE2.COM;1`, located at node `BLD14`. The resulting output file prints.

```
NFT>APPEND/PRINT FILE1.COM BLD14::FILE2.COM;1 Return
```

You can also use the `/SUBMIT` switch with this command. The `/SUBMIT` switch causes the file you just created to queue to the remote node's batch system after NFT completes the append operation.

2. This example shows how to convert from a personal computer character set to a Digital MCS character set when moving a text file to a remote system.

```
NFT>COPY/ASCII:MCS BONJOUR.TXT PARIS:: Return
```

3. This example shows how to convert a file on the personal computer that is in the French NRC to ISO on the remote system.

```
NFT>COPY/ASCII:FRENCH:ISO BONSOIR.TXT PARIS:: Return
```


COPY

COPY

The COPY command creates a new file or a new version of a file at the destination node. You can use the COPY command to copy files from the local node to the remote node and from the remote node to the local node.

When using the COPY command, you should note the following:

- If you issue a COPY command to copy a file from a remote ULTRIX system to a personal computer (using a default output file specification) and the ULTRIX file specification contains directories, then the resulting personal computer file is named incorrectly.
- If you copy a file from a remote system by using a wildcard character in the command, and if the wildcard matches a directory name, the COPY command also copies directory file. However, it contains no useful information.

Format for Copying Files to a Remote Node

```
COPY [ /ALLOCATION=number
      /ASCII:[src-charset:dest-charset]
      /BLOCK
      /CC=string
      /DELETE
      /FIXED=number
      /IMAGE
      /LSA
      /MACY11
      /MRS=number
      /NOLOG
      /NOSPAN
      /PRINT
      /STREAM
      /STREAM_CR
      /STREAM_LF
      /SUBMIT
      /UNDEFINED
      /VARIABLE
      /VFC=number ] input-file output-file
```

Format for Copying Files from a Remote Node

```
COPY [ /ASCII:[src-charset:dest-charset]
      /BLOCK
      /IMAGE
      /NOCONVERT
      /NOLOG ] input-file output-file
```

Where:

/ALLOCATION=*number* Requests that the remote system set the allocation quantity (in blocks) for the file to *number* when creating a new file on a remote system. The default is 0, which causes allocation as needed. The valid range is 0 to 2147483647. This switch is valid when copying files to a remote system.

/ASCII:
src-charset:dest-charset

Indicates that the file has records that end with a carriage return/line feed pair.

You can also use the /ASCII switch to perform character set conversion:

/ASCII

/ASCII:remote-charset

/ASCII:src-charset:dest-charset

The character sets are optional and you may also specify just the first character set. For more information on character set files see the *PATHWORKS for DOS, SETHOST Terminal Emulation Guide*.

Where:

src-charset and *dest-charset* are one of the following Digital National Replace Character (NRC) set values:

DUTCH	FINNISH	FRCAN
FRENCH	GERMAN	ISO
ITALIAN	MCS	NORDAN
PORTUGUESE	SPANISH	SWEDISH
SWISS	UK	US

Note

The character set values ISO (International Standards Organization) and MCS (Digital Multinational Character Set) are not Digital replacement character sets. The Digital character sets conform to the industry standards.

Refer to PATHWORKS for DOS, *SETHOST Terminal Emulation Guide*, for complete information on setting general operating characteristics.

If you supply only one character set, it applies to the remote system. The local system uses the current character set.

/BLOCK

Copies files as image mode files, regardless of record structure. This allows you to move files that have undefined formats or files with very long record lengths. When you indicate **BLOCK**, it forces the following attributes: **IMAGE** mode, **FIXED** length, and an **MRS** of 512.

/CC=string

Sets record attributes for files you are copying to RMS file systems only. If you are copying files to a stream system, such as TOPS-20, the record attributes are ignored. The variable *string* represents one of the following:

None	No record attributes.
FTN	Records contain FORTRAN carriage control.
CR	(Default) Records have an implied carriage return/line feed.

PRN Records contain a fixed header with print carriage control. This is used with the **/VFC** switch.

Assuming you specify an explicit record attribute for the output file, the file data already conforms to that specification.

- /DDIF** Tells the remote system that the file is a DDIF file. (Files created by DECwrite are of this type.) You rarely need to use this switch because NFT can detect DDIF files.
- /DELETE** Requests that NFT delete the copied file from the remote node after the file is copied and printed. It is valid only with the **PRINT** switch. Use this switch only when copying a file to a remote node.
- /DOTS** Tells the remote system that the file is a DOTS file. You rarely need to use this switch because NFT can detect DOTS files.
- /DTIF** Tells the remote system that the file is a DTIF file. You rarely need to use this switch because NFT can detect DTIF files.
- /FIXED=*number*** Indicates that the records within the file are all the same length. By default, the length is 128 bytes, but you can change the length with the **/MRS** switch. Use this switch when copying the file to a remote node.
- /IMAGE** Requests that NFT copy the file to the remote system with no conversion of any kind. It remains binary data. The default record format is **FIXED**, and the default maximum record size (**MRS**) is 128 bytes. You can change these defaults with the **/VARIABLE** (or **/VFC**) and **/MRS** switches. The last record can be shorter than the previous records.
- /LSA** Indicates to a remote node that records are line-sequenced ASCII. This switch is valid only when copying files to a remote node.
- /MACY11** Indicates that the remote file is in MACY11 format.

COPY

<code>/MRS=<i>number</i></code>	Sets the maximum record size, where <i>number</i> is 0 to 32767. The default MRS for variable records is 0. This means there is no maximum for variable record file copies. The default MRS for fixed record files is 128. This switch is valid only when copying files to a remote node.
<code>/NOCONVERT</code>	Requests that NFT not convert FTN or PRN carriage control characters in remote input files from RMS systems. If you do not use this switch, NFT does not convert the carriage control characters. Use this switch only with the COPY and APPEND commands.
<code>/NOLOG</code>	Requests that NFT not print a notification line after the operation is complete.
<code>/NOSPAN</code>	Requests that records not span blocks. This switch is valid only for copying to RMS file systems.
<code>/PRINT</code>	Allows you to print a file on the remote node's default printer after the file copy is complete. This switch is valid only when copying files to remote nodes.
<code>/STREAM</code>	Notifies the receiving system that the stream file records terminated with a CR/LF character sequence.
<code>/STREAM_CR</code>	Notifies the receiving system that the stream file records terminate with a CR character.
<code>/STREAM_LF</code>	Notifies the receiving system that the stream file records terminate with a LF character.
<code>/SUBMIT</code>	Allows you to queue a command file to execute at the remote node after the file copy operation completes.
<code>/UNDEFINED</code>	Indicates that the file has an undefined record format.
<code>/VARIABLE</code>	Resets the record format (RFM) to variable-length records with a maximum record size of 0 when copying files to RMS file systems. You can change the maximum record size with the /MRS switch. This switch is valid only when copying files to a remote system.

<i>/VFC=number</i>	Indicates that the file contains variable-length records with fixed control headers. The variable <i>number</i> (0 to 255 bytes) is the size of the control header when copying a file to an RMS file system. The default header size is 2. This switch is valid only when copying files to a remote node.
<i>input-file</i>	Specifies the input file(s) to copy. If you specify more than one input file, you must first specify the node, disk, drive, and pathname, and then the file names separated by commas.
<i>output-file</i>	Specifies the name(s) of the output file(s).

A file specification has one of two forms. In its longest form, it consists of a drive name, directory path, file name, and file type. In its shortest form, it consists of a file name and file type. This form assumes that the file is in the current default DOS directory.

You can use the COPY command to:

- Copy a single file—This operation copies a single file from one node to another. For example:

```
NFT>COPY SAM.TXT BLD3::SAM.TXT Return
```
- Copy a list of files—This operation copies more than one input file to the same number of output files. For example:

```
NFT>COPY BLD3::SAM.TXT,SHAM.TXT SAM.TXT,SHAM.TXT Return
```
- Copy multiple files by using wildcards—This operation also copies more than one input file to the same number of output files. However, using this method, you can specify more than one file without typing more than one file specification. For example, the following command line copies all files with a file type of SUM from the local node to node BLD3. NFT copies the files with the same file names and file types.

```
NFT>COPY *.SUM BLD3:*.SUM Return
```

If you are copying a local file to a remote node, you can include the /PRINT switch or the /SUBMIT switch. These switches cause the file you just copied to queue to the remote node's printer or batch system, after the NFT utility completes the copy operation.

If you specify multiple names, you must use commas to separate the file names. There is a limit of 10 file names for each list.

Remember that you cannot use wildcards when you use file name lists.

COPY

The NFT utility truncates file names to eight characters and file types to three characters. For example, NFT copies a remote file named THISISMYFILENAME.RIGHT;1 as THISISMY.RIG.

The DOS operating system overwrites existing files of the same name. If you copy a file named THISISMYOTHERFILENAME.RIGHT;1, NFT truncates it to THISISMY.RIG. DOS then overwrites the first file with the same name. When you copy multiple versions of the same file, DOS overwrites all versions.

Examples

1. The following command copies the local file PAGE1.TXT to the remote node BLD3. The file prints on the remote printer and then deletes from the remote node:

```
NFT>COPY/PRINT/DELETE PAGE1.TXT BLD3::PAGE1.TXT;1 
```

2. You can omit the output file specification. In this case, NFT produces the file specification the same as the input file specification. For example:

```
NFT>COPY/PRINT/DELETE PAGE1.TXT BLD3:: 
```

3. The following command copies your local file (DOS) to a remote node using a wildcard (*):

```
NFT>COPY YOURS.* JERRY::YOURS.* 
```

4. The following example copies all files from the directory on an ULTRIX system from a remote system to a personal computer subdirectory:

```
NFT>COPY VINCE"welnick keyboards":: * 
```

5. This example copies a VMS file to a personal computer:

```
NFT>COPY GUITAR"BOB"::[WIER]ESTIMATED_PROPHET.DOC;* 
```

6. The following example uses the wildcard to copy a Macintosh® file:

```
NFT>GARCIA::[JGB_FOLDER]CATS_UNDER_THE_* 
```

Note

The Macintosh treats the underscore (...CATS_UNDER_THE*) as a space and the asterisk (*) as a wildcard.

7. This example shows how to convert from a personal computer character set to a Digital MCS character set when moving a text file to a remote system.

```
NFT>COPY/ASCII:MCS BONJOUR.TXT PARIS:: 
```

8. This example shows how to convert a file on the personal computer that is in the French NRC to ISO on a remote system.

```
NFT>COPY/ASCII:FRENCH:ISO BONSOIR.TXT PARIS:: 
```


DELETE

DELETE

The **DELETE** command deletes one or more specified local or remote files. Be aware that some remote systems require a different file specification syntax from DOS. See Appendix A for details of foreign file specifications.

Format

DELETE [/NOLOG] *file-spec*

Where:

/NOLOG Requests that NPT not print a notification line after the **DELETE** operation.

file-spec Is any valid local or remote file specification.

Examples

1. The following command deletes the file **TAX.LST;3** from the remote VMS node **BLD3**:

```
NPT>DELETE BLD3::TAX.LST;3 Return
```

2. The following command deletes all versions of all files with a file type of **DOC** located on node **BLD6**:

```
NPT>DELETE BLD6::* .DOC;* Return
```

You can delete multiple files only by using wildcards. You cannot use a list of file specifications in the **DELETE** command.

DIRECTORY

The **DIRECTORY** command displays a list of local or remote file names, including the size (in blocks) and the time and date the file was last modified or created.

When using the **DIRECTORY** command, you should note the following:

1. If you use the **DIRECTORY** command to specify multiple subdirectories and some of those directories are protected, you see an error message. However, the error message does not display the volume or the directory name.
2. When you use the **DIRECTORY** command with a remote node designation and no other information (such as **DIRECTORY REMNODE:**), **NFT** uses the default file specification of *.*. This produces the correct results on all remote systems except for **ULTRIX™**. On an **ULTRIX** system, this command lists only the files that have a period in their names. To avoid this, issue the command using the following format:

```
DIRECTORY REMNODE:.*
```

Format

```
DIRECTORY [ /BRIEF ] [ /FULL ] [file-spec]
```

Where:

/BRIEF	Causes NFT to display file names only.
/FULL	Causes NFT to display file names and file attributes.
<i>file-spec</i>	Is any valid local or remote file specification. If you omit the file specification, NFT assumes the local disk and the current default directory.

Examples

1. The following command displays the file named **TAX.LST;4** located on node **BLD3**. The display includes the file's size and the time and date the file was last modified or created:

```
NFT>DIRECTORY BLD3::TAX.LST;4 Return
Directory of: BLD3 "SMITH password":.SYS$SYSROOT:[SMITH]
TAX.LST;4          20      09-JUL-90 12:30:52
```

DIRECTORY

2. This command example displays complete information for the file REPORT.DAT;4 on the remote node BLD3:

- The file name
- The size of the file
- The owner
- The type of file organization
- The type of record format used in the file
- The type of record attributes contained in the file
- The type of protection assigned to the file

The NFT utility displays all of this information when you use the /FULL switch for files on a remote node. If you request a DIR/FULL listing for a file on the local node, you see only the size of the file and the time and date the file was last modified or created.

```
NFT>DIRECTORY/FULL BLD3::REPORT.DAT;4 Return
Directory of: BLD3 "SMITH password"::SYS$SYSROOT:[SMITH]
REPORT.DAT;4
Size:      8/9                Owner:   [910,20]
Created:   22-MAR-90 15:38:36
File organization:      Sequential
Record Format:          Variable length, maximum 128 bytes
Record Attributes:     Carriage return carriage control
File protection:       System: RWED, Owner:RWED, Group:RW, World:R
```

3. The following command lists all file names on the node BLD3. The list does not include sizes, times, or dates. If a file name consists of more than 19 characters, NFT truncates the name to 19 characters in the directory listing.

```
NFT>DIRECTORY/BRIEF BLD3:: Return
Directory of: BLD3"SMITH password"::SYS$SYSROOT:[SMITH]
APNDXA.DOC;3   CHAP1.DOC;2   CHAP2.DOC;13  DATA.DAT;9
NEWS.DOC;10   MEMO.TXT;1    TEST2.TST;6   TEST3.TST;2
TEXT.DOC;8
```

EXIT

The **EXIT** command exits from the NFT utility and returns control to the DOS operating system.

Format

EXIT

Example

The following command causes you to exit from the NFT utility:

```
NFT>EXIT 
```

You can also enter and then to exit from NFT.

HELP

HELP

The HELP command displays information on your screen about NFT commands and switches.

Format

```
HELP { command-verb }  
      { /switch }
```

Where:

command-verb Is any valid NFT command verb.

/switch Is any switch acceptable to NFT command verbs. It must be separated from the command by a space.

Examples

1. The following command displays a summary of all NFT commands:

```
NFT>HELP Return
```

```
Help is available on the following
```

```
commands:
```

```
APPEND      COPY      DELETE     DIRECTORY  
EXIT        HELP      PRINT      SET  
SHOW        SUBMIT    TYPE
```

```
switches:
```

```
/ALLOCATION  /ASCII    /BLOCK     /BRIEF  
/CC         /DELETE   /FIXED     /FULL  
/IMAGE     /LSA      /MACY11    /MRS  
/NOCONVERT /NOLOG    /NOSPAN    /PRINT  
/STREAM     /STREAM_CR /STREAM_LF /SUBMIT  
/UNDEFINED  /VARIABLE /VFC
```

```
General command format is:
```

```
command /switches file file
```

```
Please note: Switches MUST always immediately follow the command.
```

```
Wildcards and file lists are supported for most commands.
```

```
NFT>
```

2. This command displays a summary of the SHOW command:

```
NFT>HELP SHOW Return
```

```
SHOW
```

The SHOW command displays defaults for remote file access.

Example:

```
NFT>SHOW node::
```

```
NFT>
```

3. The following command displays a summary of the effects of the /ASCII switch:

```
NFT>HELP /ASCII Return
```

```
/ASCII
```

The ASCII switch indicates that a file contains ASCII text.

If neither the ASCII nor the IMAGE switch is used, NFT attempts to determine the file type automatically.

This switch is valid only with the COPY and APPEND commands.

```
NFT>
```

PRINT

PRINT

The PRINT command queues a remote file to print on a remote printer.

Note

The remote node cannot be another DECnet-DOS node and must support printing.

You must indicate the file to print. Print the file on the system's default printer.

Format

PRINT { /DELETE
/NOLOG } *file-spec*

Where:

/DELETE	Requests that NFT delete the copied file from the remote node after the file copies and prints.
/NOLOG	Requests that NFT not print a notification line after completing the operation.
<i>file-spec</i>	Is any valid remote file specification.

Examples

1. The following command queues the file TAX.LST;3 (located on the remote VMS node BLD3) to print on the default system printer of that node:

```
NFT>PRINT BLD3::TAX.LST;3 Return
```

2. The following command queues the file CHAPTER.MEM on node BLD6 to print on BLD6 node's default system printer. You cannot use wildcards or lists with the PRINT command.

```
NFT>PRINT BLD6::CHAPTER.MEM Return
```

Note

On a VMS system you can redefine SYS\$PRINT in your LOGIN.COM file and change the default printer queue. When redefined, the NFT utility sends the print queue to the new destination.

3. This command example modifies a VMS default print queue to a print queue named DOS_LN03:

```
$DEFINE SYS$PRINT DOS_LN03
```


SET

SET

The SET command allows you to set default access control information as well as disk and directory information for up to ten nodes. After you set this information, you can type only the node name, followed by two colons. The NFT utility checks its default access control information table for a record matching the specified node name. If it finds a match, NFT uses this information for network access.

Format

SET *node-spec*::

Where:

node-spec:: Is a remote node name and access control information in quotes, followed by two colons.

Examples

1. The following command sets default access control information for node BLD3:

```
NFT>SET BLD3"SMITH OPEN":: 
```

2. The following command sets default access control information for node BLD1. This information includes only the user name JONES and the password NEW:

```
NFT>SET BLD1"JONES NEW":: 
```

SHOW

The **SHOW** command displays the temporary default table of remote access information for the specified node. The **SHOW** command does not display the password on the screen. Instead, it displays the character string *password* in its place.

Format

SHOW [*node-name::*]

Where:

node-name:: Is a valid remote node name. If you do not specify a node name, NFT displays the contents of the access control information table for all nodes known to the NFT temporary default table.

If NFT does not find the specified node name in the table, it checks to see if you typed default remote access control information by using NCP. If you did, NFT displays the access control information. If NFT does not find the node name in its own table or with NCP, NFT displays:

No defaults match: node-name::

SHOW

Examples

1. This example displays the contents of the access control information table for all nodes known to the NFT temporary default table (the nodes you accessed during this NFT session). Your NCP defaults do not display.

```
NFT>SHOW   
BLD3"tower password"::  
RM4"strtrfd password"::
```

2. The following command displays the access control information for node BLD3:

```
NFT>SHOW BLD3::   
BLD3"TOWER password"::
```

SUBMIT

The **SUBMIT** command requests that the specified command file run on the remote node. The specified remote node must support command file submission and execution.

Format

SUBMIT [/NOLOG] *remote-filespec*

Where:

/NOLOG

Requests that NFT not print a notification line after the **SUBMIT** operation.

remote-filespec

Has two formats. In its longest form, it consists of node and access control information followed by a file specification required by the remote node. (See Section 4.5 for details on access control information.) In its shortest form, it consists of a node name followed by a file specification appropriate to the remote node. For example, **NODE::FILE.TYP;3**.

Command files contain one or more commands that are recognized and run by the remote node's operating system. This means that the format of the commands within the file must conform to the standards of the remote system.

Example

This example requests that NFT runs the file **BACKUP.COM** on the remote node **BLD1**. You cannot use wildcards or a list of file specifications in the **SUBMIT** command.

```
NFT>SUBMIT BLD1::BACKUP.COM 
```

TYPE

TYPE

The **TYPE** command displays the contents of a local or remote file on your screen. You should use this command for ASCII files only.

Format

TYPE *file-spec*

Where:

file-spec Is any valid local or remote file specification.

Example

The following command displays the contents of the file **PREFACE.DOC** located on the remote node **BLD3**:

```
NFT>TYPE BLD3:;PREFACE.DOC Return
```

Accessing Remote Printers and Disks

The Network Device Utility (NDU) lets you access remote printers and disks as if they were directly connected to your personal computer. This connection access includes both wide area networks and local area networks. The file or printer resident on the remote node appears to be a local device except for differences in access time. These differences depend on the type of communication services for the remote node you are using.

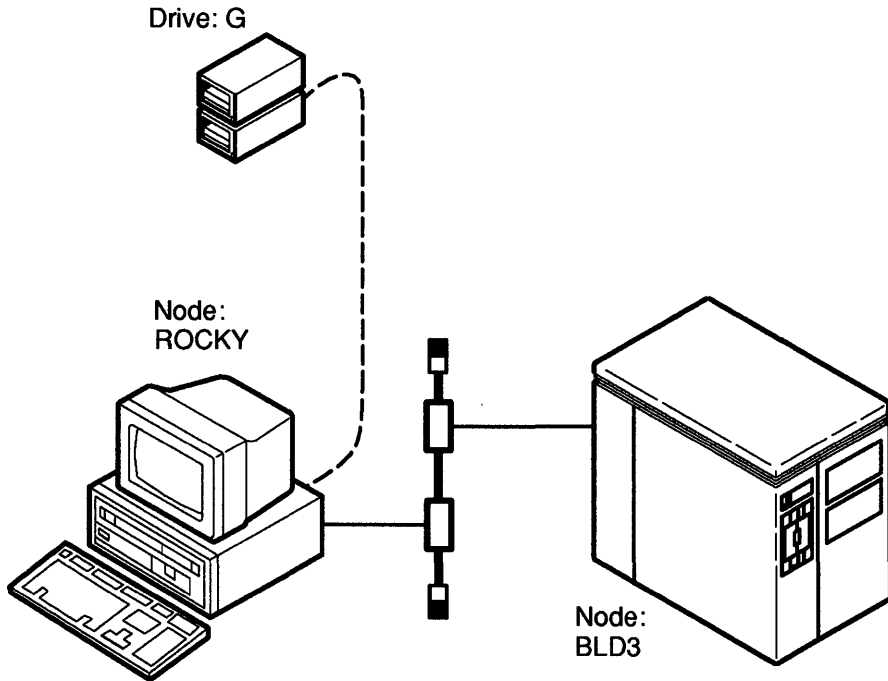
The following topics are covered in this chapter:

- Installing the virtual device drivers (see Section 5.1).
- How to run the NDU (see Sections 5.2 and 5.3).
- Entering the NDU commands to control virtual disk volumes (see Section 5.4).
- Using the HELP command (see Section 5.5).
- Special considerations while using the NDU for virtual disks (see Section 5.6).
- An NDU command summary followed by a description of each NDU command and its syntax. (see Section 5.7)

Accessing Remote Printers and Disks

You can assign a remote disk file a volume name (such as G:). Your applications software or DOS can use the disk file as if it were a local hard-disk volume (see Figure 5-1).

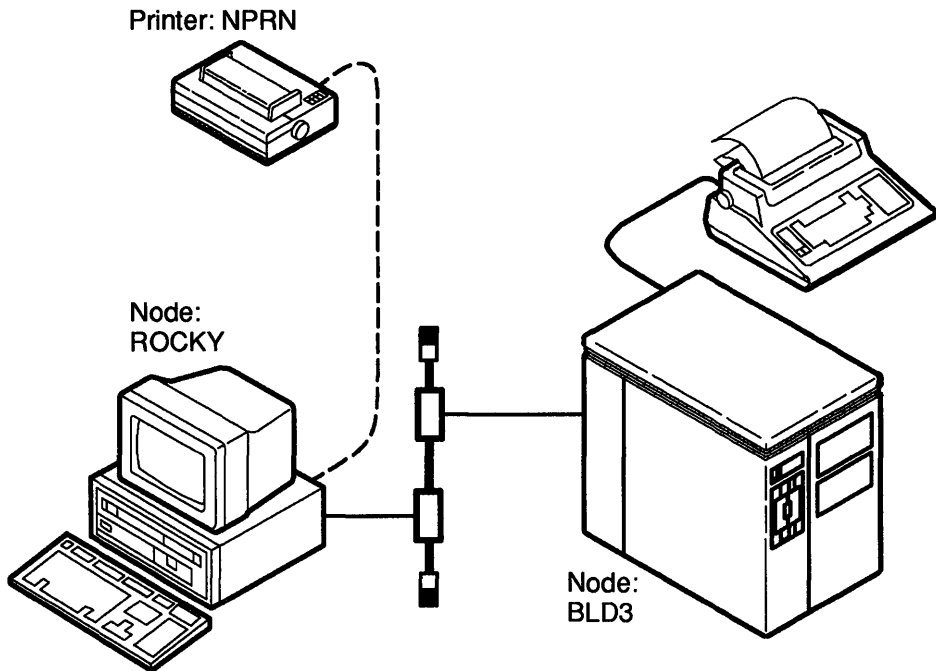
Figure 5-1 Using Virtual Disks on DECnet-DOS



LKG-4661-901

You can assign remote printers a device name (such as NPRN, PRN, LPT1, LPT2, or LPT3) and use them as if they were local printers. Because the devices only appear to be resident at the local node, they are called virtual devices (see Figure 5-2).

Figure 5-2 Using Remote Virtual Printers on DECnet-DOS



LKG-4662-901

The NDU provides virtual disk support at any remote DECnet node with file access server, such as the DECnet-VAX File Access Listener (FAL). The NDU works directly with the standard DECnet FAL and does not use the file server on the remote system. Virtual disks are single remote files that appear to be entire DOS volumes. You can use a single virtual disk privately for read and write access or you can share it for read only access. The ability to create virtual disks is useful for providing extra storage or a target for backing up local files.

Accessing Remote Printers and Disks

Any output directed to the specified printer device (such as NPRN) is forwarded to a remote system and retained until you issue a NDU command to close the file and queue it for printing.

Using the NDU, you can:

- Assign a hard disk volume name to a new or existing remote file to be used as a virtual disk volume.
- Stop the use of a remote file as a virtual disk volume.
- Delete a remote disk file that was used as a virtual disk volume.
- Assign the printer device name NPRN to a disk file at a remote node.
- Stop the use of the virtual printer at the remote node and direct the NDU to queue the save text file for printing.
- Obtain status information about all assigned hard disk drives and the printer device.

The NDU controls the use of both virtual disks and virtual printers in the following ways:

- **Virtual Disk Volumes**

The NDU allows you to access up to four virtual disk volumes at a time. You can access four volumes on the same remote node, one volume on each of four different remote nodes, or any other combination totaling up to four volumes. Each remote volume is a file that is treated as if it were a hard disk. The file is a binary file on the remote node that contains DOS directories and files.

- **Virtual Printers**

The NDU allows you to access one virtual printer at a time. You can change the virtual print device NPRN by specifying one of the following device names on the command line in your CONFIG.SYS file:

```
PRN
LPT1
LPT2
LPT3
```

The remote node providing the virtual printer saves the output to the specified printer in a temporary file on disk. When you close the virtual printer, the NDU queues the temporary file to the default printer on the remote node.

5.1 Installing the Virtual Device Drivers

Although the NDU allows you to access virtual devices, two other programs called device drivers actually perform the input and output. The NDDRV.SYS device driver performs operations for virtual disks; the NPDRV.SYS device driver performs operations for virtual printers. You must install these device drivers on the DOS operating system to use the NDU. If you do not, the NDU displays an error message indicating that the disk driver or the printer driver is not installed. For example, the following message appears when the printer driver is not installed:

```
Network printer driver not installed
```

You install the device drivers when you first install the PATHWORKS for DOS software. If you select this component as part of your configuration, the PATHWORKS for DOS installation procedure automatically installs the device drivers. Refer to PATHWORKS for DOS, *Installing and Configuring (with Diskettes)* for more details.

5.2 How to Run the Network Device Utility

When using the NDU, you can create a disk file on a remote node for use as a virtual disk volume. You can create an unlimited number of virtual disks, but you can only use four at any one time.

The NDU allows you to direct text to a remote node to queue for printing. Use one of the following methods to enter the NDU utility:

- Enter a command string that includes the NDU followed by the name of the specific NDU command, then press **[Return]**.

```
C:\>NDU command [Return]
```

After each NDU command executes, it returns the DOS prompt.

```
C:\>
```

- Supply the name of the utility, NDU, and press **[Return]**.

```
C:\>NDU [Return]
```

The NDU responds with a start-up message, then displays its own prompt. If you opened any virtual disk drives, the NDU displays the status of the drives. For example:

Network Device Utility (NDU) 4.0
Network Disk Driver Version 4.0
Network Printer Driver is not installed

Disk Drive	Status/ Ndisk	Access	Node/ User	Socket	Reads	Writes
G	CLOSED					
H	CLOSED					
I	CLOSED					
J	CLOSED					

NDU>

Enter the NDU command after the NDU prompt, and press **Return**. Continue this procedure until you have entered all the NDU commands you need.

```
NDU>command Return  
NDU>command Return  
NDU>command Return  
NDU>
```

5.3 How to Exit the Network Device Utility

To exit from the NDU, enter the **EXIT** command and press **Return**. The system returns with the DOS prompt.

```
NDU>EXIT Return
```

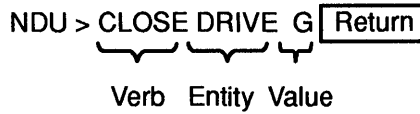
C:\>

You can use **Ctrl/Z** **Return** to exit from the NDU.

5.4 Entering the Network Device Utility Commands

The NDU commands consist of three parts: the command verb, the entity and a value. Figure 5-3 shows the command syntax.

Figure 5–3 Parts of a Network Device Utility Command



LKG-4379-901

The NDU commands act on entities and their values. When using a NDU command, you must provide the command verb, the entity, and the entity value.

5.4.1 Command Verbs

You can abbreviate command verbs to a character string that is unique to that verb. For example, you can type CRE for CREATE. For clarity and consistency, all examples in this chapter show the full command format.

The NDU provides you with command verbs that allow you to access the remote files as virtual disk volumes. It also provides you with commands that allow you to direct text to a remote node to be queued for printing. Table 5–1 lists each NDU command and its function.

Table 5–1 Network Device Utility (NDU) Commands for Virtual Disks

Command	Function
CLOSE	Requests that an existing connection to a virtual hard disk be terminated. The CLOSE command also requests that NDU end an existing connection to a remote print file and queues the file for printing.
CREATE	Instructs the NDU to create and open a new remote file and format it to be treated as a virtual hard disk or to create and open a remote printer.
DELETE	Deletes a remote file that had been used as a virtual hard disk.
HELP	Displays a message explaining the use of the NDU command and the meaning of the command parameters.
OPEN	Requests a connection to an existing remote file.

(continued on next page)

Table 5–1 (Cont.) Network Device Utility (NDU) Commands for Virtual Disks

Command	Function
SHOW	Displays the activity and current state of all virtual hard disks and the virtual printer. A status report appears at the completion of other function requests.

5.4.2 Entities

Table 5–2 lists the entities that NDU commands act upon when controlling virtual disks or printers. When you specify more than one entity in a command line, use either a space or a tab to separate them.

Table 5–2 Network Device Utility Command Entities

Entity	Function
ACCESS <i>access mode</i>	Uses the OPEN and CREATE commands to select the operating mode used to access the virtual hard disk. RW access indicates that the disk can be read from or written to, but it cannot be shared. RO access indicates that the disk can be shared but only for the purpose of reading.
ALLOCATION <i>n</i>	Specifies the initial disk file size of the virtual disk file (in blocks) on the remote node. The value of <i>n</i> is a decimal number. This entity is valid only with the CREATE command.
DRIVE <i>drive</i>	Specifies a drive name to be treated as a hard disk. The name is one alphabetic character. The drive name must be one of the names listed by the SHOW STATUS command. When you start the NDU, it displays the drives that are available by indicating they are CLOSED. You can then select G, for example, as the drive name to be used in the DOS commands and functions that address a drive.
MAX <i>xxx</i>	Specifies the maximum size of the virtual disk in megabytes. The value of <i>xxx</i> can be .360, .720, 1, 1.2, 1.44, 10, 20, 32, 64, 128, 256, or 512. The default is 32 megabytes. However, DOS Version 4.0 supports virtual disk sizes greater than 32 megabytes. The DECnet maximum is 32 megabytes.

(continued on next page)

Table 5–2 (Cont.) Network Device Utility Command Entities

Entity	Function
NDISK <i>file-name</i>	Identifies the file name associated with a drive name. For example, NDISK MYDISK1.DSK associates the file name MYDISK1.DSK with the drive name that is addressed by a DOS command. The file name can be any valid remote file name. It can include device and directory fields, provided it is properly formatted for the remote node's system type. If you include information for the device and directory fields, the total number of characters you can use in the command line is 127.
NODE <i>node-name</i> or <i>node-address</i>	Identifies the virtual disk. The node name is from 1 to 6 alphanumeric characters in length and must contain at least one alphabetic character. You can also specify a node address for the remote node. The node's address includes a unique area number and a node number, separated by a period: <i>area.node</i> Where: <i>area</i> is a number in the range of 1 to 63. <i>node</i> can be in the range of 1 to 1023.
NPRINT <i>file-name</i>	Specifies the name of the file that contains the text sent to the virtual printer before it queues for printing on the remote node. The file name can be any legal remote file name. It can include device and directory fields, provided it is properly formatted for the remote node's system type. If you include information for the device and directory fields, the total number of characters you can use in the command line is 127.
USER <i>access info</i>	Specifies access control information for a specific remote node. This is the identification information that you enter during log-on at the hard disk's resident DECnet node. Each item in the identification string must be between 1 to 39 alphanumeric characters. The access control string has the following format: USER <i>user-name</i> [/password [/account]] Where:

(continued on next page)

Table 5-2 (Cont.) Network Device Utility Command Entities

Entity	Function
	<p><i>user-name</i> states the name of the user on the specified remote node. If you do not specify USER, NCP uses the default access control information. You can set the default information by using the NCP DEFINE NODE command. For more detailed information about NCP commands, refer to the PATHWORKS for DOS, DECnet Network Management Guide.</p> <p><i>password</i> specifies the password you need to access files or programs on the specified node.</p> <p><i>account</i> specifies a 1 to 39 alphanumeric character string that consists of account information.</p> <p>You cannot supply account information unless you supply a specific password. If you supply a user name but you do not enter a password, the utility prompts you for the password.</p> <p>To store the characters representing the access information in uppercase letters, type the characters. To store characters exactly as typed, place the string in double quotation marks. Character strings that are not within quotation marks are forced to uppercase.</p>

5.5 Using the HELP Command

If you need assistance in selecting the NDU commands, use the **HELP** command. Enter the **HELP** command, then press **[Return]**. For example:

```
NDU>HELP [Return]
```

The **HELP** command displays information about the NDU commands and parameters

The format of the **HELP** command is:

```
HELP topic
```

Where topics include:

```
OPEN   CLOSE   DELETE  CREATE  SHOW   EXIT   HELP
NODE   USER   DRIVE   NDISK  NPRINT MAX   ACCESS ALLOCATION
REDIRECTED
```

For information about one of the subjects listed, enter **HELP** plus a subject. Then press **[Return]**. For example:

```
NDU>HELP SHOW [Return]
```

5.6 Special Considerations While Using Virtual Disks

While using the NDU to access virtual disks, be aware of the following information:

- When you issue the NDU CLOSE command for a remote virtual disk file residing on a personal computer that is running FAL, FAL reports an error message. This is only a warning; no error has occurred.
- The DELETE command fails if you create a virtual disk file with one NDISK file specification and then attempt to delete the virtual disk file with the following command:

```
DELETE NODE node NDISK file
```

You can delete the virtual disk, by using the file specification that you used to create the virtual disk or using the drive letter.

- After you issue the NDU DELETE command, the NDU prompts you for confirmation in the DOS style:

```
Are you sure (Y/N)?
```

5.7 Network Device Utility Command Summary

The remainder of this section discusses each command in alphabetical order.

CLOSE

CLOSE

The **CLOSE** command ends the connection between the specified drive and the virtual disk file on the remote node. The **CLOSE** command also ends the connection between the printer device and the file on the remote node. When you close a printer device, the remote file is queued to print on the remote printer. After the file is printed, it is deleted.

Format

```
CLOSE [ DRIVE drive
      NODE node-name NDISK file-name
      PRINTER ]
```

Where:

<i>drive</i>	Specifies the virtual drive to close.
<i>node-name</i>	Identifies and locates the virtual hard disk's resident node. The node name must be from 1 to 6 alphanumeric characters, including at least 1 alphabetic character. You can also specify a node-address for the remote node. Refer to Table 5-2 to see how the node address is formatted.
<i>file-name</i>	Specifies the name of the file that is the virtual disk on the remote node. It can include device and directory fields, provided it is properly formatted for the remote node's system type. If you include information for the device and directory fields, the total number of characters you can use in the command line is 127.

Examples

1. The following command ends the connection between the local node and the remote hard disk drive G:

```
NDU>CLOSE DRIVE G 
```

Any subsequent operations to drive G result in the following error message:

```
Not ready error reading drive G
```

```
Abort, Retry, Ignore?
```

2. The following command ends the connection between the printer device and the printer on the remote node:

```
NDU>CLOSE PRINTER Return
```

The file on the remote node is printed and deleted. Any subsequent operations sent to the printer device result in the following error message:

```
Not ready error writing device NPRN
```

```
Abort, Retry or Ignore?
```

CREATE

CREATE

The **CREATE** command creates and formats a new remote data file, which is treated as a hard disk. The **CREATE** command then establishes a connection between a volume name at your local node and the file.

For printers, the **CREATE** command creates a new virtual printer file and opens it for use. The printer file is created and then linked to the device **NPRN**.

Format

```
CREATE { NODE node-name NDISK file-name [ DRIVE drive  
      [ USER access info  
      ACCESS { RW }  
      ALLOCATION number  
      MAX xxx ] ]  
      PRINTER NODE node-name [ NPRINT file-name  
      [ USER access info ] ] }
```

Where:

NODE <i>node-name</i>	Identifies and locates the virtual hard disk's resident node.
NDISK <i>file-name</i>	Identifies the file name associated with a drive name.
DRIVE <i>drive</i>	Specifies the name for a drive. The name consists of one alphabetic character. You should use the NDU SHOW command to list available names for the disk. Only the names for closed drives can be used. The default is the next available drive.
USER <i>access info</i>	Indicates access control information for a specific remote node. If the access control information is omitted, the access data specified with NCP is used instead. (See the entry under USER in Table 5-2 for a description of access control information.)
ACCESS RW or RO	Identifies the operating mode for accessing the virtual disk. RO specifies that you can share the disk, but only for the purpose of reading. RW specifies that you can read from or write to the disk, but you cannot share it. The default is RW.

- ALLOCATION** *number* Specifies the initial disk file size in blocks of the virtual disk file on the remote node. The value of *number* is a decimal number greater than the minimum virtual disk size. The default is the minimum size.
- MAX** *xxx* Specifies the maximum size of the virtual disk in megabytes. The value of *xxx* can be .360, .720, 1, 1.2, 1.44, 10, 20, 32, 64, 128, 256, or 512. The default is 32 megabytes. DOS Version 4.0 supports virtual disk sizes greater than 32 megabytes. The DECnet maximum is 32 megabytes.

The NDU displays the following message while creating a new virtual hard disk file. It is an informational message only. The number of blocks displayed on the screen changes during the process.

```
CREATE DISK Writing block 0041
```

Disk Drive	Status/ Ndisk	Access	Node/ User	Socket	Reads	Writes
G	OPEN	RW	BLD1	2	0	0
	[sam]temp.dsk		sam			

Examples

- The following command creates a new remote file named ACCTDEPT.DSK to be treated as a virtual hard disk, drive G:

```
NDU>CREATE DRIVE G NODE RM6 NDISK ACCTDEPT.DSK USER JONES Return
```

You can restrict future access to this hard disk to USER JONES, depending on the specified access control information. If DRIVE G already existed and is open, the following error message displays:

```
Drive "G" is already OPEN: CREATE function not done.
```

CREATE

2. The following example defaults to the first available NDU drive and uses your NCP access control information:

```
NDU>CREATE NODE RM6 NDISK LAB.DSK 
```

3. The following command creates a file named DEMO.TXT to print at the remote DECnet node RM6:

```
NDU>CREATE PRINTER NODE RM6 NPRINT DEMO.TXT USER JONES 
```

DELETE

The DELETE command deletes an existing virtual disk file on a remote node.

Format

```
DELETE { NODE node-name NDISK file-name [USER access info] }
        { DRIVE drive }
```

Where:

NODE <i>node-name</i>	Identifies and locates the virtual hard disk's resident node.
NDISK <i>file-name</i>	Identifies the hard disk's file name.
USER <i>access info</i>	Indicates access control information for a specific remote node. If you omit the access control information, the system defaults to the access data specified with NCP. (See the entry under USER in Table 5-2 for a description of access control information.)
DRIVE <i>drive</i>	Specifies the virtual drive to delete.

After a DELETE request, the NDU displays status information about all the drives. It also displays a message requesting verification that you want to delete the specified drive. For example:

```
Are you sure (y/n)?
```

The information is followed by the NDU> prompt.

Example

The following command deletes a file named TEMPDISK.DSK at node BLD9. If a drive opens with the specified virtual disk, this command closes the drive. The drive is now available for use with another virtual disk.

```
NDU>DELETE NODE BLD9 NDISK TEMPDISK.DSK 
```

HELP

HELP

The HELP command displays information about NDU commands and entities.

Format

HELP { *command-verb* }
 { *entity* }

Where:

command-verb Identifies the NDU command for which you want information.

entity Identifies the NDU entity for which you want information.

Examples

1. The following command displays a list of all the NDU commands and entities that you can get help on:

```
NDU>HELP Return
```

The HELP command displays information about NDU commands and parameters.

The format of the HELP command is:

```
HELP topic
```

Where topics include:

```
OPEN   CLOSE   DELETE   CREATE   SHOW   EXIT   HELP
NODE   USER   DRIVE   NDISK   NPRINT  MAX   ACCESS  ALLOCATION
REDIRECTED
```

2. This example explains how the system displays help information for the CREATE command:

NDU>HELP CREATE

THE CREATE command makes a new virtual disk and makes it ready for use. The format of the CREATE command for disks is:

```
CREATE DRIVE drive NODE name NDISK name USER user_string
ACCESS access ALLOCATION blocks MAX bytes
```

If the DRIVE parameter is omitted, the first free drive is used.
If the USER parameter is omitted, the string stored by NCP is used.
If the ACCESS parameter is omitted, RW (read/write) is used.
If the ALLOCATION parameter is omitted, the minimum size file is created.
If the MAX parameter is omitted, a value of 32 (32 million bytes) is used.

The CREATE command also makes a virtual printer ready for use. The format of the CREATE command for the printer is:

```
CREATE PRINTER NODE name NPRINT name USER user_string
```

If the NPRINT parameter is omitted, a file name will be created.
If the USER parameter is omitted, the string stored by NCP is used.

OPEN

OPEN

The OPEN command establishes a connection between your local node and an existing virtual hard disk or printer at a remote DECnet node. If you open a printer, the system creates a printer file and then links to a printer device. You can only open one printer at a time. If you open a disk, the existing file links with the device specified by DRIVE.

Format

$$\text{OPEN} \left\{ \begin{array}{l} \text{NODE } \textit{node-name} \text{ NDISK } \textit{file-name} \left[\begin{array}{l} \text{DRIVE } \textit{drive} \\ \text{USER } \textit{access info} \\ \text{ACCESS } \left\{ \begin{array}{l} \text{RW} \\ \text{RO} \end{array} \right\} \end{array} \right] \\ \text{PRINTER NODE } \textit{node-name} \left[\begin{array}{l} \text{NPRINT } \textit{file-name} \\ \text{USER } \textit{access info} \end{array} \right] \end{array} \right\}$$

Where:

NODE <i>node-name</i>	Identifies and locates the virtual hard disk's resident node.
NDISK <i>file-name</i>	Identifies the file name associated with a drive name.
NPRINT <i>file-name</i>	Identifies the file name associated with the printer device.
DRIVE <i>drive</i>	Specifies a drive name. The default is the next free drive.
USER <i>access info</i>	Indicates access control information for a specific remote node. If the access control information is omitted, the access data specified with NCP is used instead. (See the entry under USER in Table 5-2 for a description of access control information.)
ACCESS RW or RO	Identifies the operating mode for accessing the virtual disk. RO specifies that the disk can be shared, but only for the purpose of reading. RW specifies that the disk can be read from or written to, but it cannot be shared. The default is RW.

Following an OPEN request, NDU displays status information about the hard disk followed by the NDU> prompt.

Examples

1. The following command opens a connection between the local node and the remote DECnet node RM102. At that node, an existing file named ACCTDEPT.DSK opens for READ WRITE access. All subsequent operations directed to the hard disk drive G go to ACCTDEPT.DSK at node RM102:

```
NDU>OPEN DRIVE G NODE RM102 NDISK ACCTDEPT.DSK 
```

If the file does not exist, the OPEN operation fails, and the NDU displays the following error message:

```
OPEN function failed: the remote server could not find that file.
```

If the drive was already in use, NDU displays the following error message:

```
Drive name is already OPEN, cannot OPEN it again.
```

If the file on the drive was already open for a create function, NDU displays the following error message:

```
OPEN function failed: requested ACCESS conflicts with another's  
use of the file.
```

2. The following command creates a file named LETTER.TXT to print at the remote DECnet node RM202:

```
NDU>OPEN PRINTER NODE RM202 NPRINT LETTER.TXT USER SAM 
```

SHOW

SHOW

The SHOW command causes the NDU to report on the status of all virtual disks and the virtual printer.

Format

SHOW STATUS

You can also display system status information through the use of DOS commands, such as DIR and CHKDSK.

Example

The following command shows the status of all virtual disks and virtual printers:

```
NDU>SHOW STATUS 
```

Sample output from this command appears as follows:

Disk Drive	Status/ Ndisk	Access	Node/ User	Socket	Reads	Writes
E	REDIRECTED					
F	OPEN	RW	RM102	6	241	68
	ND01.NVD		JONES/PASSWORD			
G	CLOSED					
H	CLOSED					

Printer Device	Status/ Nprint	Node/ User	Socket	Writes
NPRN	CLOSED			

This display includes the following information:

- The name of the disk drive.
- The drive's status
 - OPEN—The drive is available for use.
 - CLOSED—The drive is closed and not available for use.
 - REDIRECTED—The drive can no longer be accessed by the NDU because it has been reassigned for an alternate use.
- The file name.
- The access mode: read or write.

- The node name, the user name, and any account information.
- The number of the socket to which you are connected (DECnet assigns you to a socket).
- The number of completed read operations.
- The number of completed write operations.
- The name of the printer.
- The printer's status: open or closed.

The NDU displays the user and account information if you have specified them in the **CREATE** or the **OPEN** command. If you do not specify this information and it is obtained from the NCP database, the NDU does not display it.

A

File Specifications for Accessing Remote Files

A complete file name is called a file specification. A file specification provides your computer system with all the information it requires to identify a unique file. Each operating system in the network has its own set of rules for naming files.

A.1 Specifying Remote Files

PATHWORKS for DOS can copy files to and from nodes running different operating systems. Table A-1 provides file specifications for a variety of operating systems.

Table A-1 Operating Systems and File Specifications

Operating System	File Specification
DOS	<i>dev:\dir\filename.typ</i>
Macintosh®	<i>dev:[dir] [filename.type]</i> - This translates the Macintosh file specifications.
OS/2®	<i>dev:[dir]filename</i>
RSTS/E™	<i>dev:[dir]filename.typ</i>
RSX™	<i>dev:[dir]filename.typ;ver</i>
TOPS-10™	<i>dev:[p,pn,subdir]filename.typ<prot></i>
TOPS-20™	<i>dev:<dir>filename.typ.gen;att</i>
ULTRIX™	<i>dev:dir/filename.typ</i>
VMS™	<i>dev:[dir]filename.typ;ver</i>

Note

Most systems accept square brackets ([]) or angle brackets (< >) to delimit a directory name, and a period (.) or a semicolon (;) to delimit version. NFT accepts all these delimiters.

If you omit any portion of these file specification formats, NFT assumes the default. File specifications of any format other than those listed in Table A-1 are considered foreign to DECnet-DOS NFT. When you type a foreign file specification, you must enclose it in quotation marks (" "). For example:

```
NFT>COPY RM2::"DK2:[100,100]NAMES.DAT" 
```

This directs NFT to copy the RSTS/E file from the remote node RM2 to the local node, allowing unacceptable characters (such as a comma) in the directory name.

If you are using a DECnet/SNA gateway to access files on IBM systems, refer to *DECnet/SNA Data Transfer Facility Use* for information on transferring files.

A.2 Using Wildcards

Wildcards allow you to specify more than one file at a time. A wildcard is a character that takes the place of a number or letter in an MS-DOS file name or file extension. They give MS-DOS commands greater flexibility.

The wildcards discussed in this section include:

- A question mark (?)
- A percent sign (%)
- An asterisk (*)

Question Mark

A question mark in a file name or file extension indicates that any single character, can occupy that position. For example, suppose you have a directory that contains these files:

```
TEST1.DOC  
TEST2.DOC  
TESTA.DOC  
TESTB.DOC  
TESTING.TXT
```

If you type this command:

```
C:\>DIR TEST?.DOC
```

Your results would be:

Volume in drive C is DEC

Directory of C:

TEST1	DOC	5981	01-10-90	09:00a
TEST2	DOC	1890	01-01-90	10:33a
TESTA	DOC	766	01-03-90	11:21a
TESTB	DOC	1400	01-05-90	07:30a

Percent Sign

The percent sign (%) symbol is the single character wildcard for VMS and TOPS-20 systems.

Asterisk

An asterisk in a file name or file extension indicates that any number of alphanumeric characters can occupy the specified position. For example, if you use `TES*.*`, your resulting directory would be:

```
Volume in drive C is DEC
Directory of C:
```

```
TEST1   DOC      5981   01-10-90  09:00a
TEST2   DOC      1890   01-01-90  10:33a
TESTA   DOC       766   01-03-90  11:21a
TESTB   DOC      1400   01-05-90  07:30a
TESTING TXT     32544   01-03-90  11:44a
```

It is also possible to combine question marks with asterisks or question marks with asterisks. Consider these possibilities:

```
C:\>DIR TEST?.*
C:\>DIR ?E*.DOC
C:\>DIR T%ST*.TXT
```

Note

Some remote systems do not support all of these wildcards. Also, there may be wildcards which are supported on some remote systems but not supported by NFT.

Local and remote files can contain question marks, percent signs, or asterisks as wildcards. NFT cannot change the names of the files when wildcards are used. As an example, you can type:

```
COPY TEST?.* RM2:.*.*
```

To rename the files included in the remote file specification, you cannot use wildcards. For example, you do not type:

```
COPY TEST?.* RM2::FARA?.*
```

You can use wildcards with the `DELETE`, `TYPE`, and `DIRECTORY` commands. For example:

```
DELETE BLD3:.*.*.*
```

This command deletes all files in the user's default directory on the node `BLD3`.

B

Network File Transfer Utility Error Messages

When an NFT error occurs, you should receive one or more of the error messages listed in this appendix.

If you are accessing a remote system and the remote node reports an error for which no text has been defined, the NFT displays a Data Access Protocol (DAP) message similar to the following:

DAP error reported by remote node: *error-code / error-code*

The error message consists of a pair of DAP error codes in octal. The first error code indicates a specific error type. The second error code identifies the specific reason for that error. To determine the meanings of the DAP error codes, refer to Appendix D of this manual.

If a local error occurs for which there is no defined text, NFT displays a message similar to the following:

Unexpected network error, ERRNO value: *n*

Where: *n* is a decimal.

If you receive either of these messages, see your network coordinator.

B.1 Types of Network File Transfer Utility Error Messages

There are three general categories of NFT error messages:

1. Command line error messages
2. File input/output (I/O) errors
3. Network errors

Command line error messages

The NFT utility displays these messages when you mistype part of a command line (such as the verb or a switch). For example:

```
NFT>COPI PAGE1.TXT BLD3::PAGE1.TXT;1 Return
```

```
Error: Unrecognized command: COPI
```

The COPY command is mistyped.

For a complete listing refer to Section B.2.

File input/output (I/O) errors

The NFT utility displays these messages when you are accessing a local or remote file or submitting a remote file. The NFT displays remote file I/O error messages on two lines:

- The first line states the problem.
- The second line explains why the problem occurred, and the line begins with the word "Because." These messages are *secondary messages* (see Section B.4.1).

For example:

```
Error Opening File: BLD3::MYSTAT.DAT  
Because cannot assign address - possibly node name is undefined.
```

You typed a remote node name that was not previously defined.

For a complete listing see Section B.3.

Network errors

The NFT utility displays these messages when you try to run it on a system where network support or some part of it is missing or working incorrectly.

The messages are alphabetically listed within each category. See Section B.4.

When a command line specifies multiple operations and an error is encountered during one operation, NFT tries to execute subsequent operations even after it detects the error.

B.2 Command Line Error Messages

Syntax of the command line is important. Incorrect syntax can produce the messages that follow in this section.

Error: Cannot APPEND to list of files.

Explanation: You tried to append one or more files to the end of more than one file. You can have only one output file.

User Action: Append only one output file at a time.

Error: Cannot APPEND to wildcard file name.

Explanation: You included a wildcard in the output file name when you tried to append an input file to an output file.

User Action: Reenter the file name without a wildcard symbol.

Error: Cannot COPY from wildcard specification to explicit files.

Explanation: You tried to copy multiple files by using a wildcard while you wanted the output files to have specific names or file types.

User Action: Retry the COPY command without using a wildcard.

Error: Cannot SUBMIT list of files.

Explanation: You tried to submit more than one remote command file (in a list) to run on the remote node.

User Action: Submit only one file.

Error: Command string too long.

Explanation: The command line you typed was longer than 256 characters.

User Action: Shorten the command string.

Error: File lists not supported for DELETE command.

Explanation: You tried to delete a list of files. You can delete multiple files using wildcards.

User Action: Use a wildcard to delete multiple files or enter the files as separate commands.

Error: File lists not supported for DIRECTORY command.

Explanation: You tried to display a directory of a list of files. You can display multiple file names using wildcards.

User Action: Use the wildcard to display multiple files.

Error: Illegal ALLOCATION quantity: *text*

Explanation: You used the /ALLOCATION switch with the COPY command and specified a nonnumeric value.

User Action: Reenter the switch and specify a numeric value.

Error: Illegal Maximum Record Size: *value*

Explanation: You used the /MRS switch with the COPY command and specified either a nonnumeric value or a value outside of the range of 0 to 1024.

User Action: Use either a numeric value or keep the value within the specified range.

Error: Illegal VFC amount: *value*

Explanation: You used the /VFC switch and specified either a nonnumeric value or a value that is less than 0 or greater than 255.

User Action: Reissue the /VFC switch using the appropriate value.

Error in allocation amount.

Explanation: You used the /ALLOCATION switch without a value.

User Action: Reissue the /ALLOCATION switch with an appropriate value.

Error in allocation quantity.

Explanation: You used the /ALLOCATION switch with the COPY command, and the remote node did not accept this value.

User Action: Reenter a different value and try again.

Error in maximum record size.

Explanation: You used the /MRS switch with the COPY command, and the remote node did not accept the value you specified.

User Action: Reenter a different value and try again.

Error in VFC size.

Explanation: You used the /VFC switch with the COPY command and specified a fixed length header, which by the remote node did not accept.

User Action: Reenter a different value and try again.

Error: Quoted local files are illegal.

Explanation: You enclosed a local file specification in quotation marks (" "). Quotation marks are used only for remote file specifications.

User Action: Omit the quotation marks.

Error: Too many file names in list. Limit is 10.

Explanation: You included more than ten file names in a list (for example, with the APPEND command).

User Action: Enter a maximum of 10 files.

Error: Two paths in a row.

Explanation: You specified more than one path name without specifying the corresponding file names in between.

User Action: Use complete path names.

Error: Unrecognized command: *command*

Explanation: You mistyped an NFT command or tried to use a command that NFT could not recognize.

User Action: Reenter the command correctly.

Error: Unrecognized /CC option: *option*

Explanation: You used the /CC switch and specified an incorrect record attribute.

User Action: Use one of the following attributes:

- None - No attributes
- FTN - FORTRAN carriage control
- CR - Implied carriage return/line feed
- PRN - Fixed header with carriage control

Error: Unrecognized switch: *switch*

Explanation: You mistyped the name of a switch or used a switch that NFT could not recognize.

User Action: Reenter the name of the switch correctly.

Error: Wildcards are not supported in file name lists.

Explanation: You entered more than one source file specification and at least one wildcard.

User Action: Break the command up into several NFT commands.

B.2.1 Warnings

The following messages are warnings. When NFT displays these, your current operations continue.

Print switch not supported with this command.

Explanation: You used the /PRINT switch with a command that does not support the switch (for example, DELETE/PRINT).

User Action: Do not use the /PRINT switch with that command.

Warning: Cannot DELETE both remote and local files.

xxx file name(s) will be ignored.

Explanation: You specified a local and a remote file to delete. The NFT utility only deletes the file you listed first.

User Action: Enter only one file name.

Warning: Cannot perform DIRECTORY for both local and remote files.

xxx file name(s) will be ignored.

Explanation: You requested a directory listing of both remote and local file names. The NFT utility displays the directory you listed first.

User Action: Reenter the name of only one directory.

Warning: Cannot print file(s) on local printer.

Explanation: You included the /PRINT switch in a COPY operation from a remote node to the local node. You cannot print remote files on the local printer. (However, you can print local files on a remote printer.)

User Action: Do not use the /PRINT switch in this operation.

Warning: Cannot TYPE from one file to another.

Explanation: You included two file specifications with the TYPE command.

User Action: Use only one file specification with this command.

Warning: Renaming files with wildcard specifications not supported.
xxx file name(s) will be ignored.

Explanation: You tried to rename a file with the COPY command and used a wildcard in both the input file name and the output file name.

User Action: Do not use wildcards to rename files.

Warning: File(s) will NOT be deleted after copy.

Explanation: The /DELETE switch was used as an invalid switch with the COPY command. /DELETE is valid only with /PRINT.

User Action: Use the DELETE command correctly.

Name is too long for local file. Limit is 12 characters.

Enter local file name:

Explanation: You entered a file name that is too long for DOS to accept. The local file name must be 12 characters or less.

User Action: Reenter a shorter file name.

Unable to make local file name from remote name: *node-name::file-name.typ*

Enter local file name:

Explanation: You omitted the local file name with the COPY command, and NFT was not able to create a local file name from the specified remote file name.

User Action: Retry the COPY command using an explicit local file name.

B.3 File Input/Output Error Messages

The messages listed in this section are all related to accessing local and remote files or submitting a remote command file. If you receive a remote node error message, refer to the documentation for that operating system, or see your network coordinator for instructions on how to correct the problem.

Examples of file I/O errors are improper file specifications or a file read error. Remember that the NFT utility displays remote file I/O messages on two lines. The second line is a secondary message that describes why the error occurred. Many of the secondary messages can display with more than one file I/O message.

Cannot delete file: *file-name.typ*

Explanation: The NFT utility cannot delete the local file you specified because it is a directory file or has a READ ONLY protection.

User Action: Select another file name to delete.

Error Closing File: *node-name::file-name.typ*

Explanation: The NFT utility cannot close the remote file you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error Deleting File: *node-name::file-name.typ*

Explanation: The NFT utility cannot delete the remote file you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error In Directory of *node-name::*

Explanation: There is a problem with the remote directory you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error Opening File: *node-name::file-name.typ*

Explanation: The NFT utility cannot open the remote file you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error Reading from: *node-name::file-name.typ*

Explanation: The NFT utility cannot copy from the remote file you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error Submitting File:*node-name::file-name.typ*

Explanation: The NFT utility has a problem running the remote command file you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error Printing File: *node-name::file-name.typ*

Explanation: The NFT utility has a problem queuing the remote file to a printer. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error Writing to: *node-name::file-name.typ*

Explanation: The NFT utility cannot copy to the remote file you specified. The secondary message explains the specific reason.

User Action: Observe the secondary message.

Error: Cannot SUBMIT local file.

Explanation: You tried to run a local command file and only remote submissions accepted.

User Action: Do not submit a local file.

Error: Unable to type non-ASCII file: *file-name.typ*

Explanation: You tried to type a file that has a non-ASCII format. The NFT utility displays the name of the file you tried to type.

User Action: Avoid typing non-ASCII files.

File not found: *file-name.typ*

Explanation: You tried to access a file that The NFT utility could not find. The file might not exist, or you typed the file name incorrectly.

User Action: Reenter the correct file name.

Record too big - Try /BLOCK or /IMAGE

Explanation: The usual cause is attempting to transfer a binary file as ASCII. Use the /BLOCK switch to transfer stream files that were put on the remote system with DECnet-DOS, and use the /IMAGE switch to create fixed record length files on the remote system.

User Action: Retry the transfer using the appropriate (/BLOCK or /IMAGE) switch.

Unable to open file: *file-name.typ*

Explanation: The NFT utility cannot open the local file you specified.

User Action: Reenter the file name correctly.

B.4 Network Error Messages

The error messages listed in this section are related to network software, including temporary files that NFT creates.

Network errors pertain to network operation failures or rejections. For example, the network can reject connection requests by the network because of insufficient network resources or if you specify an invalid node name.

Allocation quantity too large.

Explanation: The disk on the remote system is too full to process your request.

User Action: Free up some space on that disk or select another destination.

Connection lost.

Explanation: The remote system broke the connection.

User Action: Try the connection later.

Error opening file {remote file spec} because local node is off.

Explanation: The local node is off.

User Action: Check to make sure that the state of the local node, circuit, and line is ON.

Host is down.

Explanation: You are trying to access a remote host, and it is not in operation.

User Action: Try to reconnect later.

Insufficient network resources.

Explanation: The remote system could not accept any more connections.

User Action: Try to reconnect later.

Network is down.

Explanation: You are trying to perform network activity, and the network is not currently available.

User Action: Try to reconnect later.

Network not installed.

Explanation: You must start DECnet to use NFT.

User Action: Start DECnet.

No route to host.

Explanation: This message is an indication of one of the following conditions:

- The network driver is not loaded, or you are not using the correct version.
- The line state is OFF.

- The remote host is currently unreachable.
- There is a problem with the temporary file for remote file names.
- You are copying a file from a remote node, and you included a wildcard in the command. For example, COPY NODE:.*.*. The NFT utility first requests a directory list of the remote files. The NFT utility then stores these file names in a temporary local file on the default disk. For example, if a problem occurs with the temporary file because of lack of disk space, the NFT displays this message.

User Action: Take the appropriate action as indicated by the messages listed.

Remote system DAP buffer size < 256.

Explanation: Two programs on different systems have such different buffer sizes that they cannot communicate with one another.

User Action: Upgrade the network software on the remote system.

B.4.1 Secondary Error Messages

The NFT utility displays secondary error messages with remote file I/O messages. Many of the secondary messages can occur with more than one file I/O message. Also, NFT can display a file I/O message with one of several secondary messages depending on the cause of the problem.

Because bad record size.

Explanation: The specified record size is either invalid or illegal for the specified operation. Fixed length records must be the length specified by the MRS and must be nonzero.

User Action: Specify MRS correctly and use correct attributes.

Because cannot assign address, possibly node name is undefined.

Explanation: The network rejected an attempted connection because the remote node name did not correspond to any node name defined at the local node.

User Action: Define the node using NCP.

Because cannot open file.

Explanation: An error occurred on a file open operation.

User Action: Try a different file or try again later.

Because cannot position to end of file.

Explanation: The NFT utility cannot append the specified input file to the end of the specified output file.

User Action: Reenter the file name.

Because device is write locked.

Explanation: NFT cannot create the specified output file because the output device is write locked.

User Action: Change the write protection on the output device or try another device.

Because directory full.

Explanation: You tried to copy a file to a remote node, and the remote directory is already full.

User Action: Specify another directory.

Because directory not found.

Explanation: The specified directory does not exist on the node/device specified or defaulted in the file specification.

User Action: Specify the correct directory.

Because disk quota exceeded.

Explanation: You tried to perform a remote file operation, and you have exceeded your disk storage quota.

User Action: Delete some files on the remote disk to make room for your operation. You can also increase your disk quota.

Because error in directory name.

Explanation: The specified directory does not conform to the syntax of the target system.

User Action: Use the correct syntax.

Because error in file name.

Explanation: The specified file name does not conform to the syntax of the target system.

User Action: Use the correct syntax.

Because error in record attribute.

Explanation: You specified remote file record attributes (such as FORTRAN, implied line feed/carriage return, embedded or VMS print file) that are not valid.

User Action: Use correct remote file attributes.

Because error in record format.

Explanation: You specified a remote file record format (such as fixed, variable, VFC, or stream) that is not valid.

User Action: Use correct remote file attributes.

Because file locked by other user.

Explanation: Another user currently has the specified file locked or open for writing. You can also receive this message in a full directory listing.

User Action: Retry the operation later.

Because file not found.

Explanation: The specified file or files do not exist.

User Action: Check the file name and reenter the correct file name.

Because file read error.

Explanation: An irrecoverable error has occurred.

User Action: This may be a hardware error. Contact your network system manager.

Because file write error.

Explanation: An irrecoverable error has occurred.

User Action: Select another destination.

Because illegal record attributes.

Explanation: The file's record attributes (that is, FORTRAN, implied CR/LF, embedded, or VMS print file) are invalid or unsupported by NFTP or the remote system.

User Action: Use supported file record attributes.

Because incorrect user access information.

Explanation: The network rejected an attempted connection because the specified access control information (user ID, password, and account) does not match a valid account on the remote node.

User Action: Retry using the correct user access information.

Because invalid record format.

Explanation: The file's record format (that is, fixed, variable, VFC, or stream) is invalid or unsupported by NFT or the remote FAL.

User Action: Use the appropriate file record format.

Because invalid wildcard operation.

Explanation: The remote system rejects the specified wildcard as inappropriate for the specified operation.

User Action: Reenter without using a wildcard symbol.

Because privilege violation.

Explanation: An operation was specified for which you do not have privileges.

User Action: Specify the correct access information to access that specific file.

B.5 Miscellaneous Error Messages

Here is a list of other related error messages you can encounter.

Address already in use.

Explanation: There is an internal error that should not occur.

User Action: None.

Address family not supported by protocol family.

Explanation: There is an internal error that should not occur.

User Action: None.

Argument list too long.

Explanation: There is an internal error that should not occur.

User Action: None.

Argument too long.

Explanation: There is an internal error that should not occur.

User Action: None.

Attribute read error.

Explanation: The remote system reported an error.

User Action: None.

Attribute write error.

Explanation: The remote system reported an error.

User Action: None.

Bad address.

Explanation: There is an internal error that should not occur.

User Action: None.

Bad block on device.

Explanation: The remote system reported an error.

User Action: None.

Bad file number.

Explanation: There is an internal error that should not occur.

User Action: None.

Bad terminator or bad value for switch:

Explanation: There is a syntax error in the command line.

User Action: Reenter using the correct syntax.

Bad version number.

Explanation: The remote system reported an error.

User Action: Correct the version number.

Cannot close file.

Explanation: The remote system reported an error.

User Action: Try again later or try another destination.

Cannot get JFN for file.

Explanation: The remote system reported an error.

User Action: None.

Cannot open local file with DOS reserved names:

Explanation: You cannot open a local with a DOS reserved name.

User Action: Retry opening a local file using a different name.

Cannot PRINT list of files.

Explanation: The PRINT command does not accept print lists.

User Action: Break the command up into separate commands for each file.

Cannot PRINT local file.

Explanation: The PRINT command can only print remote files.

User Action: Use the DOS PRINT command to print your files.

Connection refused.

Explanation: The remote system rejected the connection.

User Action: None.

Connection reset by peer.

Explanation: The remote system disconnected the link.

User Action: Retry to make the connection.

Connection timed out.

Explanation: FAL on the remote system did not respond.

User Action: Try again later.

Destination address required.

Explanation: There is an internal error that should not occur.

User Action: None.

Device not available.

Explanation: The remote system reported an error.

User Action: Select another device.

Device not found.

Explanation: The remote system reported an error.

User Action: None.

Device not ready.

Explanation: The remote system reported an error.

User Action: Try again later.

Device or file full.

Explanation: The remote system reported an error.

User Action: Use another device.

Disk full or other error in closing file.

Explanation: The local disk drive is full.

User Action: Try using another disk drive.

Disk full or other error in writing file.

Explanation: The local disk drive is full.

User Action: Try using another disk drive.

Disk usage exceeds quota.

Explanation: The remote system reported an error.

User Action: Use another disk device or delete some files and try again.

Error in data type.

Explanation: The remote system does not support that required data type.

User Action: None.

Error in file type extension.

Explanation: You have a syntax error.

User Action: Reenter the correct file.

Failure to get transmit buffer.

Explanation: There is an internal error that should not occur.

User Action: Try again later.

F11-ACP could not access file.

Explanation: The remote system reported an error.

User Action: Try again later.

F11-ACP could not create file.

Explanation: The remote system reported an error.

User Action: Try again later.

F11-ACP could not mark file for deletion.

Explanation: The remote system reported an error.

User Action: Try again later.

File activity precludes operation.

Explanation: The remote system reported an error.

User Action: Try again later.

File already open.

Explanation: The remote system reported an error.

User Action: Try again later.

File exists.

Explanation: The remote system reported an error.

User Action: Select another destination file name.

File extend failure.

Explanation: The remote system reported an error.

User Action: The remote disk may be full or fragmented. Try another destination.

File is currently in an undefined state.

Explanation: The remote system reported an error.

User Action: Try again later or contact your network system manager.

File name syntax error.

Explanation: The remote system reported an error.

User Action: Reenter the correct file name.

File name too long.

Explanation: The remote system reported an error.

User Action: Reenter the correct file name.

File not found.

Explanation: The remote system reported an error.

User Action: Reenter the correct file name.

FSZ field invalid.

Explanation: The remote system reported an error.

User Action: Specify a correct value for FSZ.

Illegal record encountered.

Explanation: The remote system reported an error.

User Action: None.

Invalid DAP message type received.

Explanation: There is an incompatibility between NFT and the remote system implementation of FAL. This message could also mean that there are hardware problems on the network.

User Action: Upgrade the remote system or check your hardware.

Invalid DAP message format received.

Explanation: There is an incompatibility between NFT and the remote system implementation of FAL. It could also mean that there are hardware problems on the network.

User Action: Upgrade remote system or check your hardware.

Invalid file options.

Explanation: The remote system rejected the file options requested.

User Action: Try again using different options.

Invalid object name format.

Explanation: The remote system reported an invalid object name.

User Action: There is an internal error that should not occur.

Invalid wildcard context value.

Explanation: The remote system reported an invalid wildcard context value.

User Action: There is an internal error that should not occur.

Logical name error.

Explanation: The remote system reported an error.

User Action: Check the logical name or use another.

LPT page limit exceeded.

Explanation: The remote system reported an error.

User Action: Break file up before printing.

Message too long.

Explanation: There is an internal error that should not occur.

User Action: None.

Network capacity exceeded.

Explanation: The remote system reported an error.

User Action: Try again later.

Network operation failed at remote node.

Explanation: The remote system reported an error.

User Action: Check the remote node.

Network operation not supported.

Explanation: The remote system does not support the requested operation.

User Action: None.

Network operation timed out.

Explanation: The remote system attempted a network operation and timed out.

User Action: Try again later.

NFAR error, FFIRST with dir in progress.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, file not open.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, FNEXT with no dir in progress.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, illegal access.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, invalid function.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, invalid RATs.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, missing ::.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, missing quote.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, no * or *.*.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, no more room.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, record buffer too small.

Explanation: There is an internal error that should not occur.

User Action: None.

NFAR error, socket not found.

Explanation: There is an internal error that should not occur.

User Action: None.

No buffer space available.

Explanation: The network has no more buffer space available.

User Action: Run the following:

- SHOW EXEC COUNTERS
- SHOW LINE COUNTERS

- SHOW CIRCUIT COUNTERS

Refer to the PATHWORKS for DOS *Troubleshooting Guide*.

Node name cannot be longer than 6 characters.

Explanation: You entered a node name longer than 6 characters.

User Action: Retry entering a node name of appropriate length.

Node is unreachable.

Explanation: The node specified is not reachable at this time.

User Action: Try again later.

No more sockets available.

Explanation: There are no sockets available.

User Action: Close some file service connections to release some sockets or close some other network applications. Use NCP to increase the number of logical links available. Use the NCP DEFINE EXECUTOR MAXIMUM LINKS command to modify the parameters.

No such device.

Explanation: There is no device as specified.

User Action: Reenter the device name correctly.

Not supported by remote.

Explanation: The specified operation is not supported by the remote system.

User Action: None.

Operation illegal or invalid for file organization.

Explanation: The specified operation is either invalid or illegal.

User Action: None.

Operation not supported on socket.

Explanation: The socket does not support the specified operation.

User Action: None.

Parity error on device.

Explanation: There is a hardware problem.

User Action: Contact your network system manager.

Path and file names not valid for SET, SHOW, or EXIT commands.

Explanation: These commands do not accept file names.

User Action: Retry using without path names.

Protocol not available.

Explanation: There is an internal error that should not occur.

User Action: None.

Protocol not supported.

Explanation: There is an internal error that should not occur.

User Action: None.

Remote object is too busy.

Explanation: The remote object is busy.

User Action: Try again later.

Result too large.

Explanation: There is an internal error that should not occur.

User Action: None.

Socket is already connected.

Explanation: There is an internal error that should not occur.

User Action: None.

Socket is not connected.

Explanation: The remote system's FAL exited or the remote system disconnected the link prematurely.

User Action: Try again later.

Socket operation on nonsocket.

Explanation: There is an internal error that should not occur.

User Action: None.

Socket type not supported.

Explanation: There is an internal error that should not occur.

User Action: None.

Spool or submit command file error.

Explanation: The remote system was unable to accept the PRINT or SUBMIT commands.

User Action: Check the remote system.

Switches not valid for SET, SHOW, or EXIT commands.

Explanation: The specified switches are invalid.

User Action: Retry the operation without switches.

Task not available.

Explanation: The specified task is not available.

User Action: Retry later.

Too many open files.

Explanation: You have too many files or logical links opened.

User Action: Close some files to free up file handles.

Unexpected DAP message received.

Explanation: The remote system displays an unexpected message.

User Action: There is a problem on the remote system or there is a hardware problem. Fix the hardware or upgrade the remote system software.

Unmatched quotes:

Explanation: You omitted a quotation mark, or you entered too many.

User Action: Reenter the operation using matched quotation marks.

Unrecognized local file specification, remote-to-remote operations not supported.

Explanation: The file specification is invalid.

User Action: Reenter the file specification using the correct file information.

Unrecognized object.

Explanation: There is a syntax error.

User Action: Reenter the object using the correct syntax.

Unrecognized remote file specification:

Explanation: There is a syntax error.

User Action: Reenter the remote file specification using the correct syntax.

Unrecognized remote file specification, local-to-local operations not supported.

Explanation: There is a syntax error.

User Action: Reenter the file specification using the correct syntax.

Unsupported DAP flag field received.

Explanation: There is a syntax error.

User Action: Reenter the DAP flag field using the correct syntax.

VFC value must be less than 256.

Explanation: You entered a value larger than 256.

User Action: Retry using a value less than 256.

B.5.1 Warnings

Here is a list of warnings you may encounter.

Extra local file names will be ignored.

Explanation: Your file name is too long.

User Action: Reenter the operation using a shorter file name.

Extra remote file names will be ignored.

Explanation: You entered too many file names.

User Action: Enter only one file name.

Warning: Destination file name list longer than source list.

Explanation: There is a syntax error.

User Action: Reenter the file name list correctly.

Network Device Utility Error Messages

This appendix lists the error messages you can encounter when using the Network Device Utility (NDU). Section C.1 explains Virtual Disk error messages. Section C.2 lists Virtual Printer error messages. There are three categories of NDU error messages:

- **Command line error messages** - The NDU displays these messages when you mistype part of a command line. The error message includes a circumflex (^) which appears directly under the error, followed by explanatory text. For example:

```
NDU>CLOSE DRIVE ONE Return
```

```
      ^ Disk DRIVE must be a single character.
```

- **Network error messages** - The NDU displays these messages when you try to run NDU on a system where network support or some part of it is missing or not working properly.
- **Execution error messages** - You can encounter these messages when you attempt to perform an NDU function, such as CREATE or OPEN.

The remainder of this appendix lists, explains, and provides a user action for all of the NDU error messages.

C.1 Virtual Disk Error Messages

The following error messages can occur when you perform operations on virtual disks:

C.1.1 Command Line Error Messages

The error messages listed in this section are related to the syntax of a command line.

ACCESS must be either RO or RW.

Explanation: You can specify only RO or RW as valid ACCESS options.

User Action: Reenter using the correct response.

DRIVE *name* is too long. It must be a single character.

Explanation: A disk drive's name cannot exceed one alphabetic character.

User Action: Retry using a single letter.

Initial ALLOCATION must be a decimal number between 0 and 65535.

Explanation: You can specify a decimal value in the range 0 to 65535. However, because of internal requirements, NDU always sets this value greater than 30 when it creates the file.

User Action: Reenter using another value.

Unexpected syntax error.

Explanation: You mistyped a command.

User Action: Reenter the command using correct syntax.

Unrecognized command.

Explanation: You mistyped an NDU command or tried to use a command that NDU could not recognize.

User Action: Reenter the command using correct syntax.

Unrecognized HELP option; try typing just HELP.

Explanation: You typed an invalid HELP option. Just type HELP for assistance.

User Action: Type HELP again to retry.

Unrecognized NDISK file name.

Explanation: A valid file name consists of 1 to 126 alphanumeric characters. The file name can include device and directory fields, but it must be in the proper format for the remote node's system type.

User Action: Reenter the operation using the correct format.

Unrecognized node name.

Explanation: A valid node name consists of 1 to 6 alphanumeric characters. It must contain at least one alphabetic character. The node name can also be a unique numeric address in the format of *area.number*.

User Action: Reenter the operation using the correct format.

Unrecognized NPRINT file name.

Explanation: A valid file name consists of 1 to 126 alphanumeric characters. The file name can include device and directory fields, but it must be in the proper format for the remote node's system type.

User Action: Reenter the operation using the correct format.

Unrecognized Parameter.

Explanation: You supplied an invalid parameter with an NDU command. Type HELP for assistance.

User Action: Type HELP.

Unrecognized SHOW option.

Explanation: NDU could not recognize the option. The only valid option for the SHOW command is STATUS.

User Action: Retry the command using the STATUS option.

Unrecognized USER name.

Explanation: NDU could not recognize the USER name. A valid USER name consists of access control information: *user-id*, *password*, and *account*. Each field can include 1 to 39 alphanumeric characters. Enter a valid USER name.

User Action: Reenter the user name correctly.

C.1.2 Network Errors

A network error consists of a pair of messages separated by a colon. The message format is:

ndu-specific-message : *dap-specific-message*

The first message can be one of the following NDU messages:

CLOSE during DELETE function failed.

Explanation: The operation failed.

User Action: Look up the DAP specific error message in Appendix D of this manual.

CLOSE function failed.

Explanation: The CLOSE function failed.

User Action: Look up the DAP specific error message in Appendix D of this manual.

CREATE function failed.

Explanation: The operation failed.

User Action: Look up the DAP specific error message in Appendix D of this manual.

DELETE function failed.

Explanation: The operation failed.

User Action: Look up the DAP specific error message in Appendix D of this manual.

OPEN function failed.

Explanation: The second half of the message string can be one of the following DAP messages:

A received DAP message was poorly formed.

Explanation: During the dialog with the File Access Server on the remote node (which was necessary to execute your request), a message was received of the proper type, but it did not contain the proper contents.

User Action: For an explanation of the DAP message, refer to the *PATHWORKS for DOS, DECnet Programmer's Reference Manual*. For additional assistance, see the person responsible for your network.

An unexpected DAP message was received.

Explanation: During the dialog with the File Access Server on the remote node (which was necessary to execute your request), a message was received of the wrong type.

User Action: For an explanation of the DAP message, refer to the *PATHWORKS for DOS, DECnet Programmer's Reference Manual*. For further assistance, see the person responsible for your network.

Internal error.

Explanation: An internal error message was detected.

User Action: See the person responsible for your network for assistance.

The remote server's buffer is too small.

Explanation: During the dialog with the File Access Server on the remote node, that was necessary to execute your request, a message was received which included the maximum possible buffer size for the remaining messages. This size is too small to support the communication.

User Action: See the person responsible for your network for assistance.

The remote server could not find that file.

Explanation: The NDISK file that you specified was not found at that remote node.

User Action: You must specify a file that exists.

The remote server reports DAP error *xx*.

Explanation: The remote File Access Listener is relaying a DAP error message to you. The error message consists of a pair of DAP error codes. The first error code indicates a specific error type. The second error code identifies the specific reason for that error.

User Action: To determine the meanings of the displayed error codes, refer to the PATHWORKS for DOS, *DECnet Programmer's Reference Manual*. For further assistance, see the person responsible for your network.

The specified file is not a virtual disk file.

Explanation: The NDISK file that you specified does not have the proper attributes. You may have made an error in the file name.

User Action: Specify a file that has been created with the CREATE command.

C.1.3 Execution Error Messages

The following error messages can occur when you attempt to execute an NDU command or fail to install the virtual disk driver beforehand:

ALLOCATION used for CREATE only, value IGNORED.

Explanation: The ALLOCATION parameter was not accepted with the command you specified.

User Action: : You can only supply the ALLOCATION parameter with the CREATE command. You cannot use it with other NDU commands.

All the drives are in use, cannot OPEN another drive.

Explanation: All four disk drives are in use.

User Action: Close one of the open drives before opening another.

All the drives are in use, CREATE function not done.

Explanation: You cannot create a new virtual disk because all drives are in use.

User Action: Close one of the drives.

Cannot CLOSE a drive, the disk driver is not installed.

Explanation: You cannot close a virtual disk without having the virtual disk driver installed.

User Action: Refer to the appropriate installation guide for your system for instructions on installing the virtual disk driver. Install the virtual disk driver, and run the NDU utility to close the drive.

Cannot CREATE a drive, the disk driver is not installed.

Explanation: You cannot create a virtual disk without having the virtual disk driver installed.

User Action: Refer to the appropriate installation guide for your system for instructions on installing the virtual disk driver. Install the virtual disk driver, and run the NDU utility to create the disk drive.

Cannot CREATE a disk file without including NDISK parameter.

Explanation: You cannot issue a CREATE command without the NDISK parameter.

User Action: Reissue the CREATE command, and specify the name of the file that will contain the virtual disk on the remote node.

Cannot CREATE a disk file without including NODE parameter.

Explanation: You cannot issue a CREATE command without the NODE parameter.

User Action: Reissue the CREATE command, and specify the name of the remote node with NODE.

Cannot CREATE a disk greater than 32MB running DOS less than DOS 4.0.

Explanation: You cannot define the maximum size of the virtual disk to be larger than 32 megabytes unless you use DOS Version 4.0 or greater.

User Action: Specify the MAX entity to be 32 megabytes or smaller, or install DOS Version 4.0 or greater.

Cannot DELETE a disk file without including NDISK parameter.

Explanation: You cannot issue a DELETE command without the NDISK parameter.

User Action: Reissue the DELETE command, and specify the name of the file that will contain the virtual disk on the remote node.

Cannot DELETE a disk file without including NODE parameter.

Explanation: You cannot issue a DELETE command without the NODE parameter.

User Action: Reissue the DELETE command, and specify the name of the remote node with NODE.

Cannot DELETE, the disk driver is not installed.

Explanation: You cannot delete a virtual disk without having the virtual disk driver installed.

User Action: Refer to the appropriate installation guide for your system for instructions on installing the virtual disk driver. Install the virtual disk driver, and run the NDU utility to delete the disk drive.

Cannot OPEN a drive, the disk driver is not installed.

Explanation: You have not installed NDU properly.

User Action: Refer to the appropriate installation guide for your system for instructions on installing the virtual disk driver. Repeat the procedure, and run the NDU.

Cannot OPEN a drive without including NDISK parameter.

Explanation: You cannot issue an OPEN command without the NDISK parameter.

User Action: Reissue the OPEN command, and specify the name of the file that will contain the virtual disk on the remote node.

Cannot OPEN a drive without including NODE parameter.

Explanation: You cannot issue an OPEN command without the NODE parameter.

User Action: Reissue the OPEN command, and specify the name of the remote node with NODE.

Drive *name* is already OPEN; CREATE function not done.

Explanation: The drive is already open. You cannot format the new data file.

User Action: Use the SHOW STATUS command to list the open drives.

Drive *name* is already OPEN, cannot OPEN it again.

Explanation: The drive is already open.

User Action: Try another drive.

Drive *name* is not a network disk.

Explanation: The drive name that you specified is invalid. The drive name consists of one alphabetic character.

User Action: Use the SHOW STATUS command to list the valid drive names.

Drive *name* is not a network disk, cannot CLOSE it.

Explanation: You cannot close a drive which is not a virtual disk drive.

User Action: Use the SHOW STATUS command to list the open drives.

Drive *name* is not a network disk, cannot OPEN it.

Explanation: You can use only the drive names listed by the SHOW STATUS command.

User Action: Use a proper drive name.

Drive *name* is not a network disk, CREATE function not done.

Explanation: You can use only the drive names listed by the SHOW STATUS command.

User Action: Use a proper drive name.

Drive *name* is not OPEN, cannot CLOSE it.

Explanation: You cannot close an unopened drive.

User Action: Use the SHOW STATUS command to list the status of the drives.

Drive *name* is not OPEN, cannot DELETE its disk.

Explanation: You cannot delete a disk without having the drive opened.

User Action: Use the SHOW STATUS command to list the status of the drives.

Drive *name* is too long. It can only be a single character.

Explanation: You supplied an invalid name for the disk drive. A valid drive name consists of one alphabetic character.

User Action: Use the SHOW STATUS command to list the valid drive names.

DRIVE, NDISK, or PRINTER must be specified, the CLOSE function was not done.

Explanation: You cannot issue the CLOSE function without specifying DRIVE, NDISK, or PRINTER as a parameter.

User Action: To perform the CLOSE function, you must specify either the name of the virtual disk drive or the file that will be the virtual disk on the remote node.

Length of NDISK parameter is too long.

Explanation: An invalid length was supplied for the file name. The NDISK parameter specifies the name of the file which will be the virtual disk on the remote node. The valid range for the file name is 1 to 126 alphanumeric characters.

User Action: Enter a valid file name.

Length of NODE parameter is too long.

Explanation: You specified an invalid node name. A valid node name consists of one to six alphanumeric characters. It must contain at least one alphabetic character.

User Action: Enter a valid node name.

Length of USER parameter is too long.

Explanation: You supplied an invalid user name. The string must be less than 120 characters. NDU could not recognize the user name. A valid user name consists of access control information: *user-id*, *password*, and *account*. Each field can include 1 to 39 alphanumeric characters.

User Action: Enter a valid USER name.

NDISK not found, the CLOSE function was not done.

Explanation: You must specify either DRIVE or NDISK; the CLOSE function was not done.

User Action: To determine which drive, you must specify a drive name, an NDISK file specification, or both.

The Disk Driver is not installed.

Explanation: You cannot create or use a virtual disk without first installing the Virtual Disk Driver.

User Action: Refer to the appropriate installation guide for your system for instructions on installing the virtual disk driver. Repeat the installation procedure, and then try to run NDU.

C.2 Virtual Printer Error Messages

The following error messages can occur when you use the virtual printer feature:

C.2.1 Command Line Error Messages

The error messages listed in this section are all related to the syntax of a command line.

DELETE function failed: Connect failed, Unrecognized node name.

Explanation: A valid node name consists of 1 to 6 alphanumeric characters. It must contain at least one alphabetic character. The node name can also be a unique numeric address in the format of *area.number*.

User Action: Retry the operation using a valid node name.

OPEN/CREATE function failed: Connect failed, Unrecognized node name.

Explanation: A valid node name consists of 1 to 6 alphanumeric characters. It must contain at least one alphabetic character. The node name can also be a unique numeric address in the format of *area.number*.

User Action: Retry the operation using a valid node name.

Unexpected syntax error.

Explanation: You mistyped a command.

User Action: Reenter the command using correct syntax.

Unrecognized command.

Explanation: You mistyped an NDU command or tried to use a command that NDU could not recognize.

User Action: Reenter the command using correct syntax.

Unrecognized HELP option; try typing just HELP.

Explanation: You typed an invalid HELP option.

User Action: Type HELP for assistance.

Unrecognized NPRINT file name.

Explanation: A valid file name consists of 1 to 126 alphanumeric characters. The file name can include device and directory fields, but it must be in the proper format for the remote node's system type.

User Action: Retry the operation using a valid file name.

Unrecognized Parameter.

Explanation: You supplied an invalid parameter with an NDU command.

User Action: Type HELP for assistance.

Unrecognized SHOW option.

Explanation: NDU could not recognize the option. The only valid option for the SHOW command is STATUS.

User Action: Reenter the SHOW command using the STATUS option.

Unrecognized USER name.

Explanation: NDU could not recognize the USER name. A valid USER name consists of access control information: *user-id*, *password*, and *account*. Each field can include 1 to 39 alphanumeric characters. Enter a valid USER name.

User Action: Reenter the user name using correct syntax.

C.2.2 Network Errors

A network error consists of a pair of messages separated by a colon. The message format is:

OPEN/CREATE function failed: *dap-specific-message*

Only the second message changes. The second half of the message string can be one of the following DAP messages:

A received DAP message was poorly formed.

Explanation: During the dialog with the file access server on the remote node (which was necessary to execute your request), a message was received of the proper type, but it did not contain the proper contents.

User Action: For an explanation of the DAP message, refer to the PATHWORKS for DOS, *DECnet Programmer's Reference Manual*. For additional assistance, see the person responsible for your network.

An unexpected DAP message was received.

Explanation: During the dialog with the file access server on the remote node (which was necessary to execute your request), a message was received of the wrong type.

User Action: For an explanation of the DAP message, refer to the PATHWORKS for DOS, *DECnet Programmer's Reference Manual*. For further assistance, see the person responsible for your network.

Internal error.

Explanation: An internal error message was detected.

User Action: See the person responsible for your network for assistance.

The remote server's buffer is too small.

Explanation: During the dialog with the file access server on the remote node (which was necessary to execute your request), a message was received which included the maximum possible buffer size for the remaining messages. This size is too small to support the communication.

User Action: See the person responsible for your network for assistance.

The remote server could not find that file.

Explanation: The NDISK file that you specified was not found at that remote node.

User Action: You must specify a file which exists.

The remote server reports DAP error *xx*.

Explanation: The remote File Access Listener is relaying a DAP error message to you. The error message consists of a pair of DAP error codes. The first error code indicates a specific error type. The second error code identifies the specific reason for that error.

User Action: To determine the meanings of the displayed error codes, refer to the PATHWORKS for DOS, *DECnet Programmer's Reference Manual*. For further assistance, see the person responsible for your network.

C.2.3 Execution Error Messages

The following error messages can occur when you attempt to execute an NDU command or fail to install the virtual printer driver beforehand.

Cannot CLOSE the printer, the Printer Driver is not installed.

Explanation: You cannot close the printer without having the virtual printer driver installed.

User Action: Refer to the installation guide for instructions on installing the virtual printer driver. Install the virtual printer driver, and run the NDU utility to close the printer.

Cannot OPEN/CREATE the printer, the Printer Driver is not installed.

Explanation: You cannot open/create the virtual printer without having the virtual printer driver installed.

User Action: Refer to the PATHWORKS for DOS, *Installing and Configuring (with Diskettes)* for instructions on installing the virtual printer driver. Install the virtual printer driver, and run the NDU utility to open/create the printer.

Cannot OPEN/CREATE the printer without including NODE parameter.

Explanation: You cannot issue the OPEN or CREATE command without the NODE parameter.

User Action: Reissue the OPEN or CREATE command, and specify the name of the remote node with NODE.

Length of NODE parameter is too long.

Explanation: You supplied an invalid node name. A valid node name consists of one to six alphanumeric characters. It must contain at least one alphabetic character.

User Action: Enter a valid node name.

Length of NPRINT parameter is too long.

Explanation: You supplied an invalid file name for the NPRINT parameter. A valid file name consists of 1 to 126 alphanumeric characters. The file name can include device and directory fields, but it must be in the proper format for the remote node's system type.

User Action: Enter a valid file name.

Length of USER parameter is too long.

Explanation: You supplied an invalid user name. The string must be less than 120 characters. NDU could not recognize the user name. A valid user name consists of access control information: *user-id*, *password*, and *account*. Each field can include 1 to 39 alphanumeric characters.

User Action: Enter a valid USER name.

Printer is not OPEN, cannot CLOSE it.

Explanation: You cannot close an unopened printer.

User Action: Use the SHOW STATUS command to list the status of the printer.

The Printer Driver is not installed.

Explanation: You cannot create or use a virtual printer without first installing the Virtual Printer Driver.

User Action: Refer to the appropriate installation guide for your system for instructions on installing the virtual printer driver. Repeat the installation procedure, and then try to run NDU.

Data Access Protocol Error Messages

The Network Task Error log utility provides extended error support to transparent file access operations. This appendix lists the Network File Transfer Data Access Protocol (DAP) error messages that this utility can return.

D.1 Overview

The DAP messages return status from the remote file system or from the operation of the cooperating process using DAP. The 2 byte status field (16 bits) is divided into two fields:

- Maccode (bits 12-15) - Contains the error type code (see Table D-1).
- Miccode (bits 0-11) - Contains the specified error reason code (see Table D-2, Table D-3, or Table D-4, depending on the error type).

D.1.1 Maccode Field

The value returned in the maccode field describes the functional type of the error that has occurred. The specific reason for the error is given in the miccode field. Miccode values correlating to each maccode value listed in Table D-1 are found in the table referred to in the last column.

Table D-1 Data Access Protocol Maccode Field Values

Field Value (Octal)	Error Type	Description	See
0	Pending	The operation is in progress.	Table D-3
1	Successful	Returns information that indicates success	Table D-3

(continued on next page)

Table D-1 (Cont.) Data Access Protocol Maccode Field Values

Field Value (Octal)	Error Type	Description	See
2	Unsupported	This implementation of DAP does not support the specified request.	Table D-2
3	Reserved		
4	File open	Errors that occur before a file is successfully opened.	Table D-3
5	Transfer error	Errors that occur after a file is opened and before it is closed.	Table D-3
6	Transfer warning	For operations on open files, indicates that the operation completed, but not with complete success.	Table D-3
7	Access termination	Errors associated with terminating access to a file.	Table D-3
10	Format	Error in parsing a message. Format is not correct.	Table D-2
11	Invalid	Field of message is invalid (that is, bits that are meant to be mutually exclusive are set, an undefined bit is set, a field value is out of range, or an illegal string is in a field).	Table D-2
12	Sync	DAP message received out of synchronization.	Table D-4
13-15	Reserved		
16-17	User defined status maccodes		Table D-4

D.1.2 Miccode Field

The value returned in this field identifies the specific reason for the error type defined in the maccode field (see Section D.1.1). Miccode field values are defined in three different tables, each table associated with certain maccode values, as outlined below:

- Table D-2: For use with maccode values 2, 10, and 11
- Table D-3: For use with maccode values 0, 1, 4, 5, 6, and 7
- Table D-4: For use with maccode value 12

Table D-2 follows. The DAP message type number (column 1) is specified in bits 6 to 11, and the DAP message field number (column 2) is specified in bits 0 to 5. The field where the error is located is described in the third column.

Table D-2 Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
Miscellaneous message errors		
00	00	Unspecified DAP message error
	10	DAP message type field (TYPE) error
Configuration message errors		
01	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	BITCNT field (BITCNT)
	20	Buffer size field (BUFSIZ)
	21	Operating system type field (OSTYPE)
	22	File system type field (FILESYS)
	23	DAP version number (VERNUM)
	24	ECO version number field (ECONUM)
	25	USER protocol version number field (USRNUM)
	26	DEC software release number field (DECVER)
	27	User software release number field (USRVER)
	30	System capabilities field (SYSCAP)
Attributes message errors		
02	00	Unknown field

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	10	DAP message flags field (FLAGS)
	11	Data stream identification (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN 256)
	14	Bit count field (BITCNT)
	20	Attributes menu field (ATTMENU)
	21	Data type field (DATATYPE)
	22	Field organization field (ORG)
	23	Record format field (RFM)
	24	Record attributes field (RAT)
	25	Block size field (BLS)
	26	Maximum record size field (MRS)
	27	Allocation quantity field (ALQ)
	30	Bucket size field (BKS)
	31	Fixed control area size field (FSZ)
	32	Maximum record number field (MRN)
	33	Run-time system field (RUNSYS)
	34	Default extension quantity field (DEQ)
	35	File options field (FOP)
	36	Byte size field (BSZ)
	37	Device characteristics field (DEV)
	40	Spooling device characteristics field (SDC); reserved
	41	Longest record length field (LRL)
	42	Highest virtual block allocated field (HBK)
	43	End-of-file block field (EBK)
	44	First free byte field (FFB)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Micode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	45	Starting LBN for contiguous file field (SBN)
Access message errors		
03	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Access function field (ACCFUNC)
	21	Access options field (ACCOPT)
	22	File specification field (FILESPEC)
	23	File access field (FAC)
	24	File-sharing field (SHR)
	25	Display attributes request field (DISPLAY)
	26	File password field (PASSWORD)
Control message errors		
04	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Control function field (CTLFUNC)
	21	Control menu field (CTLMENU)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	22	Record access field (RAC)
	23	Key field (KEY)
	24	Key of reference field (KRF)
	25	Record options field (ROP)
	26	Hash code field (HSH); reserved for future use
	27	Display attributes request field (DISPLAY)
	30	Block count (BLKCNT)
Continue message errors		
05	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Continue transfer function field (CONFUNC)
Acknowledge message errors		
06	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	15	System-specific field (SYSPEC)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Access complete message errors

07	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Access complete function field (CMPFUNC)
	21	File options field (FOP)
	22	Checksum field (CHECK)

Key definition message errors

12	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Key definition menu field (KEYMENU)
	21	Key option flags field (FLG)
	22	Data bucket fill quantity field (DFL)
	23	Index bucket fill quantity field (IFL)
	24	Key segment repeat count field (SEGCNT)
	25	Key segment position field (POS)
	26	Key segment size field (SIZ)
	27	Key of reference field (REF)
	30	Key name field (KNM)
	31	Null key character field (NUL)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	32	Index area number field (IAN)
	33	Lowest level area number field (LAN)
	34	Data level area number field (DAN)
	35	Key data type field (DTP)
	36	Root VBN for this key field (RVB)
	37	Hash algorithm value field (HAL)
	40	First data bucket VBN field (DVB)
	41	Data bucket size field (DBS)
	42	Index bucket size field (IBS)
	43	Level of root bucket field (LVL)
	44	Total key size field (TKS)
	45	Minimum record size field (MRL)
Allocation message errors		
13	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Allocation menu field (ALLMENU)
	21	Relative volume number field (VOL)
	22	Alignment options field (ALN)
	23	Allocation options field (AOP)
	24	Starting location field (LOC)
	25	Related file identification field (RFI)
	26	Allocation quantity field (ALQ)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	27	Area identification field (AID)
	30	Bucket size field (BKZ)
	31	Default extension quantity field (DEQ)
Summary message errors		
14	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Summary menu field (SUMENU)
	21	Number of keys field (NOK)
	22	Number of areas field (NOA)
	23	Number of record descriptors field (NOR)
	24	Prologue version number (PVN)
Date and time message errors		
15	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Date and time menu field (DATMENU)
	21	Creation date and time field (CDT)
	22	Last update date and time field (RDT)
	23	Deletion date and time field (EDT)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	24	Revision number field (RVN)
	25	Backup date and time field (BDT)
	26	Physical creation date and time field (PDT)
	27	Accessed date and time field (ADT)
Protection message errors		
16	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Protection menu field (PROTMENU)
	21	File owner field (OWNER)
	22	System protection field (PROTSYS)
	23	Owner protection field (PROTOWN)
	24	Group protection field (PROTGRP)
	25	World protection field (PROWLDR)
Name message errors		
17	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	20	Name type field (NAMETYPE)

(continued on next page)

Table D-2 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 2, 10, and 11

Type Number (bits 6-11)	Field Number (Bits 0-5)	Field Description
	21	Name field (NAMESPEC)
Access control list message errors		
20	00	Unknown field
	10	DAP message flags field (FLAGS)
	11	Data stream identification field (STREAMID)
	12	Length field (LENGTH)
	13	Length extension field (LEN256)
	14	Bit count field (BITCNT)
	15	System specific field (SYSPEC)
	20	Access control list repeat count field (ACLCNT)
	21	Access control list entry field (ACL)

Table D-3 follows. The error code number (column 1) is contained in bits 0 to 11. For corresponding RMS or FCS status codes, refer to the appropriate DECnet or RMS documentation for each remote system.

Table D-3 Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
0	Unspecified error
1	Operation aborted
2	F11-ACP could not access file
3	File activity precludes operation
4	Bad area ID
5	Alignment options error
6	Allocation quantity too large or 0 value

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
7	Not ANSI D format
10	Allocation options error
11	Invalid (that is, synchronous) operation at AST level
12	Attribute read error
13	Attribute write error
14	Bucket size too large
15	Bucket size too large
16	BLN length error
17	Beginning of file detected
20	Private pool address
21	Private pool size
22	Internal RMS error condition detected
23	Cannot connect RAB
24	\$UPDATE changed a key without having attribute of XB\$CHG set
25	Bucket format check byte failure
26	RSTS/E close function failed
27	Invalid or unsupported COD field
30	F11-ACP could not create file (STV - system error code)
31	No current record (operation not preceded by get/find)
32	F11-ACP deaccess error during close
33	Data area number invalid
34	RFA-accessed record was deleted
35	Bad device, or inappropriate device type
36	Error in directory name
37	Dynamic memory exhausted
40	Directory not found
41	Device not ready

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
42	Device has positioning error
43	DTP field invalid
44	Duplicate key detected; XB\$DUP not set
45	F11-ACP enter function failed
46	Operation not selected in ORG\$ macro
47	End of file
50	Expanded string area too short
51	File expiration date not yet reached
52	File extend failure
53	Not a valid FAB (BID does not = FB\$BID)
54	Illegal FAC for record operation, or FB\$PUT not set for create
55	File already exists
56	Invalid file ID
57	Invalid flag bits combination
60	File is locked by other user
61	F11 ACP find function failed
62	File not found
63	Error in file name
64	Invalid file options
65	Device/file full
66	Index area number invalid
67	Invalid IFI value or unopened file
70	Maximum NUM (254) areas/key XABS exceeded
71	\$INIT macro never issued
72	Operation illegal or invalid for file organization
73	Illegal record encountered (with sequential files only)
74	Invalid ISI value on unconnected RAB

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
75	Bad key buffer address (KBF = 0)
76	Invalid key field (KEY = 0 or negative)
77	Invalid key of reference (\$GET/\$FIND)
100	Key size too large
101	Lowest level index area number invalid
102	Not ANSI-labeled tape
103	Logical channel busy
104	Logical channel number too large
105	Logical extend error; prior extend still valid
106	LOC field invalid
107	Buffer-mapping error
110	F11-ACP could not mark file for deletion
111	MRN value = negative or relative key > MRN
112	MRS value = 0 for fixed length records and/or relative files
113	NAM block address invalid (NAM = 0 or is not accessible)
114	Not positioned to EOF (with sequential files only)
115	Cannot allocate internal index descriptor
116	Indexed file; primary key defined
117	RSTS/E open function failed
120	XABs not in correct order
121	Invalid file organization value
122	Error in file's prolog (reconstruct file)
123	POS field invalid (POS > MRS; STV = XAB indicator)
124	Bad file date field retrieved
125	Privilege violation (OS denies access)
126	Not a valid RAB (BID does not = RB\$BID)
127	Illegal RAC value

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
130	Illegal record attributes
131	Invalid record buffer address (either odd or not word aligned if BLK-IO)
132	File read error
133	Record already exists
134	Bad RFA value (RFA=0)
135	Invalid record format
136	Target bucket locked by another stream
137	F11-ACP remove function failed
140	Record not found
141	Record not locked
142	Invalid record options
143	Error while reading prolog
144	Invalid RRV record encountered
145	RAB stream currently active
146	Bad record size (RSZ > MRS or NOT = MRS if fixed-length records)
147	Record too big for user's buffer
150	Primary key out of sequence (RAC = RB\$SEQ for \$PUT)
151	SHR field invalid for file (cannot share sequential files)
152	SIZ field invalid
153	Stack too big for save area
154	System directive error
155	Index tree error
156	Error in file type extension on FNS is too big
157	Invalid user buffer address (0, odd, or not word aligned if BLK-IO)
160	Invalid user buffer size (USZ=0)
161	Error in version number

(continued on next page)

**Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with
Maccode Values of 0, 1, 4, 5, 6, and 7**

Error Code (Bits 0-11)	Error Description
162	Invalid volume number
163	File write error (STV = system error code)
164	Device is write locked
165	Error while writing prolog
166	Not a valid XAB (XAB = odd; STV = XAB indicator)
167	Default directory invalid
170	Cannot access argument list
171	Cannot close file
172	Cannot deliver AST
173	Channel assignment failure (STV = system error code)
174	Terminal output ignored due to <Ctrl/O>
175	Terminal input aborted due to <Ctrl/Y>
176	Default file name string address error
177	Invalid device ID field
200	Expanded string address error
201	File name string address error
202	FSZ field invalid
203	Invalid argument list
204	Known file found
205	Logical name error
206	Node name error
207	Operation successful
210	Inserted record had duplicate key
211	Index update error occurred; record inserted
212	Record locked, but read anyway
213	Record inserted in primary key is okay; may not be accessible by secondary keys or RFA

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Micode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
214	File was created, but not opened
215	Bad prompt buffer address
216	Asynchronous operation pending completion
217	Quoted string error
220	Record header buffer invalid
221	Invalid related file
222	Invalid resultant string size
223	Invalid resultant string address
224	Operation not sequential
225	Operation successful
226	Created file superseded existing version
227	File name syntax error
230	Timeout period expired
231	FB\$BLK record attribute not supported
232	Bad byte size
233	Cannot disconnect RAB
234	Cannot get JFN for file
235	Cannot open file
236	Bad JFN value
237	Cannot position to end of file
240	Cannot truncate file
241	File currently in an undefined state; access is denied
242	File must be opened for exclusive access
243	Directory full
244	Handler not in system
245	Fatal hardware error
246	Attempt to write beyond EOF

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
247	Hardware option not present
250	Device not attached
251	Device already attached
252	Device not attachable
253	Shareable resource in use
254	Illegal overlay request
255	Block check or CRC error
256	Caller's nodes exhausted
257	Index file full
260	File header full
261	Accessed for write
262	File header checksum failure
263	Attribute control list error
264	File already accessed on LUN
265	Bad tape format
266	Illegal operation on file descriptor block
267	Rename; two different devices
270	Rename; new file name already in use
271	Cannot rename old file system
272	File already open
273	Parity error on device
274	End of volume detected
275	Data overrun
276	Bad block on device
277	End of tape detected
300	No buffer space for file
301	File exceeds allocated space; no blocks left

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
302	Specified task not installed
303	Unlock error
304	No file accessed on LUN
305	Send/receive failure
306	Spool or submit command file failure
307	No more files
310	DAP file transfer checksum error
311	Quota exceeded
312	Internal network error condition detected
313	Terminal input aborted due to <Ctrl/C>
314	Data bucket fill size bucket size in XAB
315	Invalid expanded string length
316	Illegal bucket format
317	Bucket size of LAN does not equals IAN in XAB
320	Index not initialized
321	Illegal file attributes (corrupt file header)
322	Index bucket fill size is greater than bucket size in XAB
323	Key name buffer not readable or writeable in XAB
324	Index bucket will not hold two keys for key of reference
325	Multibuffer count invalid (negative value)
326	Network operation failed at remote node
327	Record is already locked
330	Deleted record successfully accessed
331	Retrieved record exceeds specified key value
332	Key XAB not filled in
333	Nonexistent record successfully accessed
334	Unsupported prolog version

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
335	Illegal key of reference in XAB
336	Invalid resultant string length
337	Error updating RRVs; some paths to data may be lost
340	Data types other than string limited to one segment in XAB
341	Reserved
342	Operation not supported over network
343	Error on write behind
344	Invalid wildcard operation
345	Working set full (cannot lock buffers in working set)
346	Directory listing: error in reading volume set name, directory name, or file name
347	Directory listing: error in reading file attributes
350	Directory listing: protection violation in trying to read the volume set, directory, or file name
351	Directory listing: protection violation in trying to read file attributes
352	Directory listing: file attributes do not exist
353	Directory listing: unable to recover directory list after continue transfer (skip)
354	Sharing not enabled
355	Sharing page count exceeded
356	UPI bit not set when sharing with BRO set
357	Error in access control string
360	Terminator not seen
361	Bad escape sequence
362	Partial escape sequence
363	Invalid wildcard context value
364	Invalid directory rename operation
365	User structure (FAB/RAB) became invalid during operation

(continued on next page)

Table D-3 (Cont.) Data Access Protocol Miccode Values for Use with Maccode Values of 0, 1, 4, 5, 6, and 7

Error Code (Bits 0-11)	Error Description
366	Network file transfer mode precludes operation
6000 to 7777	User defined errors

Table D-4 follows. The message type number is contained in bits 0 to 11.

Table D-4 Data Access Protocol Miccode Values for Use with Maccode Value 12

Type Number (Bits 0-11)	Message Type
0	Unknown message type
1	Configuration message
2	Attributes message
3	Access message
4	Control message
5	Continue transfer message
6	Acknowledge message
7	Access complete message
10	Data message
11	Status message
12	Key definition attributes extension message
13	Allocation attributes extension message
14	Summary attributes extension message
15	Date and time attributes extension message
16	Protection attributes extension message
17	Name message
20	Access control list extended attributes message

Glossary

ASCII

An acronym for American Standard Code for Information Interchange. An 8-bit standard code adopted to facilitate the interchange of data between various types of data processing and data communications equipment.

access control information

Optional security information that you might need in order to access a remote node. See also **default access control information**.

account

A character string that consists of 1 to 39 alphanumeric characters.

adjacent node

A node in which the local node is physically connected by a single line.

alphanumeric

A character string that contains alphabetic or numeric characters, or a combination of both. The term *alphanumeric* is a combination of the words *alphabetic* and *numeric*.

application

A program (other than the operating system) that performs specific functions to meet user requirements. Applications available with DECnet-DOS include the Network File Transfer (NFT) utility, the Network Device Utility (NDU), the Network Control Program (NCP), and SETHOST utility with network virtual terminal services.

asynchronous mode

A data transmission method that sends one character at a time. Also refers to commands (as in a windowing environment), that may be sent without waiting for a response from the previous command. See also **synchronous mode**.

asynchronous device

A device that transmits signals at irregular intervals to the system with which it is communicating.

binary

The number system with a radix of 2, or a property involving a condition of exactly 2 possibilities.

buffer

A temporary storage area in a node's memory. Buffers temporarily hold data being transferred to and from the node. The size and the number of buffers determine the amount of data that a buffer can store.

circuit

The communications path that operates over a physical line connecting two nodes.

command

An instruction typed in at a terminal. A command requests the software monitoring a terminal or reading a command procedure to perform some predefined operation.

command file

A file containing commands and data that the command interpreter can accept in lieu of the user's typing the commands individually on a terminal.

command switch

A word or character string that modifies the way a command operates.

communication path

The route a message takes, through various hardware devices, when it is sent from one node to another.

counter

A register or storage location that displays statistics about the flow of network messages for your node. Counters record error conditions and accumulate their totals.

database

A collection of interrelated data on one or more mass storage devices. The collection is organized to facilitate efficient and accurate inquiry and update.

data type

An interpretation applied to a string of bits such as integers, reals, or characters.

DEC Multiport Repeater (DEMPR)

A piece of Digital networking hardware that can connect up to eight ThinWire Ethernet segments and optionally connect them to a backbone cable, or DELNI.

DECnet

A family of Digital software products that extend the capabilities of Digital computers to perform various operations over a network. DECnet provides many standard applications for accessing files and databases that exist on other nodes.

DECnet-DOS

The version of DECnet for the IBM compatible computer using MS-DOS operating systems.

DECnet Logical Link (DLL)

A cooperative venture between two tasks with an agreement to communicate. A logical link connection is required before data can be exchanged between two tasks.

DECnet Router

A device that routes data within parts of a DECnet.

default

An assumption made by a system or language translator when no specific choice is given by the program.

default access control information

The Network File Transfer (NFT) utility allows you to use security information, including user name, password, and account number. It uses this information you last specified for the node name. This is the default access-control information. See also **access control information**.

default directory

The directory in which the monitor searches when a user has not provided a directory specification.

default value

An assigned quantity for a device or program that is set by the manufacturer. A default value in a program is usually the most common or safest answer.

DELNI

An electronic device that permits workstations to exchange information over a 20 meter (65 foot) square area. You can connect up to eight workstations to one DELNI, or you can connect two DELNIs to each other. (One unit can connect to eight other DELNI LANS.)

device

A specific name for a disk or diskette that is currently storing data files.

device control unit

A hardware device that controls the reading, writing, or display of data at one or more input/output devices or terminals.

device drivers

Instructions the computer follows to reformat data for transfer to and from a particular peripheral device.

directory

A specific name assigned to a collection of files stored on disk or a diskette. A directory can provide a list of the file names that it contains.

emulate

To imitate the behavior of a terminal on another system. When your personal computer emulates a terminal on a host system, it performs many of the same functions as the host terminal.

end node

A DECnet term referring to a member of the network that can do everything except route packets through on behalf of other nodes. ULTRIX, MS-DOS, and OS/2 implementations of DECnet are all end nodes.

Ethernet

A communications concept for local communication networks that employ coaxial cable as a passive communications medium to interconnect computer systems.

File Access Listener (FAL)

A utility that runs as a background process to enable file transfers and accesses for utilities such as Network Device Utility (NDU) and Network File Transfer (NFT).

file specification

A complete file identification including an optional drive name and path name. The file specification must include at least a file name followed by an optional file type and version number.

hard disk

A fast auxiliary storage device that is either mounted in its own case, or permanently mounted inside a computer. It has a storage capacity of several million bytes of information.

host node

A node on the network that your node can access for the purpose of sharing resources and information.

Job Spawner

Runs as a utility while awaiting incoming connection requests from other nodes in the network. The Job Spawner activates DECnet servers as background processes on your node (such as FAL). The Job Spawner also searches for connection request object names or numbers in its database, and then runs the program.

local area network (LAN)

A privately owned data communications system that offers high-speed communications channels optimized for connecting information processing equipment. The geographical area is usually limited to a section of a building, an entire building, or a group of buildings.

local area transport (LAT)

A proprietary Digital architecture for terminal servers on Ethernet networks designed to conserve bandwidth and off-load processing from hosts.

local node

The node that you are currently using to communicate with other nodes.

logging

The process of recording information from an occurrence in the network. Logging is the process that generates a record of the event.

Microsoft Disk Operating System (MS-DOS)

The standard operating system used by the IBM Personal Computer and compatible computers. The manufacturer of MS-DOS is Microsoft Corporation.

MS-DOS

See Microsoft Disk Operating System.

Network Basic Input/Output System (NETBIOS)

Used in DOS as the interface to the network for accessing information. Similar in function to the Digital Network Architecture and Data Access protocols.

network

A group of interconnected computers or systems that communicate with each other to share resources and information.

Network Control Program (NCP)

The block that contains the necessary information to set up a virtual circuit or to accept or reject a request to set up a virtual circuit. For DOS and OS/2, NCP allows you to perform Network Management tasks and to test the network hardware and software.

Network Device Utility (NDU)

A DECnet utility that defines virtual disk drives and virtual printers on remote systems. Capabilities include definition of files on remote systems as virtual disks and using them as if they were directly connected to your computer.

network driver

A set of instructions or software code that performs most of the functions that involve direct interfacing to the operating system. These functions include buffer management, interprocess communication management, and handling the interface between device drivers. The network driver and the network process appear as a single device driver from the user perspective.

Network File Transfer Utility (NFT)

A program that allows users to access or delete files residing on DECnet hosts that provide network file access capabilities. NFT initiates the service requests that will be carried by the FAL program.

network manager

A person who assigns and updates node names and node addresses. The network manager also provides administrative assistance to persons using the network.

node

Any terminal, station, or communications computer in a computer network. An individual computer system in a network that can communicate with other computer systems in the network.

node address

A unique numeric character string that identifies a node to other systems in the network.

node definition

A character string that identifies a particular node. The definition includes both the node address and the node name.

node name

A one to six character alphanumeric string (which must contain one letter) that identifies a node to other users in the network.

octal

A number system with a radix of 8. Octal numerals are frequently used to represent binary numerals, with each octal digit representing a group of 3 binary digits.

operating system

Software that controls the execution of computer programs and that may provide scheduling debugging, input/output control, and related services.

parameter

A variable that is given a specific value that is passed to a program before execution.

password

A character string that uniquely confirms your identity to the system.

prompt

Text displayed that requests information from the user; also the symbol representing specific information that the user should supply, such as the VMS "\$" command prompt.

protocols

Rules or formats that operating systems must follow to conduct effective communications with other computers in a network.

queue

A waiting line for completing a service, such as printing files.

remote

Physically distant from a local computer such as a video display terminal or printer.

remote node

Any node on the network other than the node you are currently using.

user name

A string consisting of 1 to 39 alphabetic characters identifying a user at a remote node.

utility

A program that helps the user run other programs, operating systems, and equipment.

variable-length record

A record in a file in which records are not uniform in length.

virtual disks

Single remote files that appear to be entire DOS volumes.

virtual printers

Printers, accessed remotely, in network resource sharing.

wildcard

A symbol used to specify multiple files with related names, without specifying each file by its full name.

window

A rectangular section of your screen used for running programs or applications.

A

Access

- checking, 3-3
- remote, 1-4

Access control information

- See also* Security
- changing defaults, 4-10
- defining for remote nodes, 4-7
- defining with the NFT utility, 4-8
- displaying, 4-11
- displaying access defaults, 4-41
- password size, 4-7
- saving, 4-9
- setting defaults, 4-40
- using defaults, 4-9

Accessing

- remote files, 4-1, A-1
- remote printers, 5-4
- the FAL utility, 3-4

Adjacent nodes, 1-2

Algorithm

- with the FAL utility, 4-12

APPEND command

- in NFT utility, 4-19, 4-20

Appending files, 4-15, 4-20

Applications

- running under windows, 1-5

ASCII, 4-13

- file attributes, 4-13
- files, 3-5, 3-6, 4-12, 4-13
- switch, 4-6

Asterisk

- using as a wildcard, A-2

B

Batch files, 2-7

- with the NFT utility, 4-18

Binary files, 3-5, 3-6, 4-12

/BRIEF switch

- Network File Transfer switch, 4-16

C

C language programming, 1-5

Carriage controls

- converting, 4-13

Changing files

- See* Appending files

CLOSE command

- in NDU, 5-7

Command files, 4-18

- running, 4-18
- submitting, 4-43
- used with NFT, 4-2
- with the Job Spawner utility, 2-2

Command line

- error messages, B-2, B-3
- invoking FAL from, 3-2
- NDU error messages, C-2
- sample for FAL, 2-6
- starting FAL from, 3-1

Commands

- in the FAL utility, 3-5

Communications

- physical connection, 1-2
- programming task-to-task, 1-5

- Configurations
 - installing DECnet-DOS, 3-3
 - testing and maintaining network, 1-4
- Connections
 - processing different requests, 2-1
- Control keys
 - Ctrl/Z, 4-7
- COPY command
 - in NFT, 4-15, 4-16, 4-19, 4-24
- Copying
 - files, 4-12, 4-15, 4-24
 - files to remote, 4-14
 - local files to remote node, 4-16
 - multiple files with wildcard, 4-29
 - remote files to local node, 4-15
- CREATE command
 - in NDU, 5-7, 5-14
- Ctrl/Z
 - NFT exit, 4-7

D

- DAP
 - See* Data Access Protocol
- Data
 - ASCII, 4-12, 4-13
 - binary, 4-12
 - image, 4-13
- Data Access Protocol (DAP)
 - error messages, D-1
 - Maccode field, D-1
 - Miccode field, D-2
- Databases
 - creating for Job Spawner, 2-2
 - DECACC.DAT file, 3-3
 - DECOBJ.DAT file, 2-1
 - sample contents of, 2-7
- DECACC.DAT database, 3-3
- DECnet
 - capabilities, 1-5
 - defined, 1-1
 - logging messages in the directory, 2-5
 - maximum file specifications, 4-14
 - Test Receiver, 2-1
- DECnet-DOS
 - configuration procedure, 3-3
 - defined, 1-2
 - functions, 1-2
 - predefined object numbers, 2-3
 - programming library, 1-5
 - using Microsoft Windows with, 1-5
 - using NETBIOS, 1-5
 - utilities, 1-5
 - FAL, 3-3
 - Job Spawner, 2-3
 - NDU, 5-1
 - NFT, 4-1
- DECnet-VAX, 5-3
- DECOBJ.DAT database, 3-3
- DECOBJ.DAT file
 - contents of, 2-7
 - in Job Spawner, 2-1
- Defaults
 - changing access control, 4-10
- DEFINE ACCESS command
 - in NCP, 3-3
- DEFINE OBJECT command
 - in NCP, 2-2, 3-3
- DELETE command
 - in NDU, 5-7, 5-17
 - in NFT, 4-16, 4-19, 4-32
- Deleting files, 4-16
 - using the NFT command, 4-32
- DEMPR
 - See* Digital ThinWire Ethernet Multiport Repeater
- Devices
 - accessing remote, 5-1
 - installing remote printer drivers, 5-5
 - limit of disks in operation, 5-5
 - printer errors, C-10
 - remote print queues, 5-14
 - remote printers, 5-3, 5-4
 - saving space with Microsoft Windows, 3-2
 - sharing disks, 1-4
 - sharing printers, 1-4
 - special considerations, 5-11
 - the NDDRV.SYS driver, 5-5

Devices (cont'd)
 the NPDRV.SYS driver, 5-5
 using NDU, 1-4
 virtual disk access limit, 5-4
 virtual printer access limit, 5-4

Digital ThinWire Ethernet Multiport Repeater (DEMPR), 1-2

Directory
 displaying information, 4-8, 4-16, 4-33
 logging messages in DECnet, 2-5

DIRECTORY command
 in NFT, 4-16, 4-19, 4-33

Disks
 accessing remote, 5-1
 assigning volume names, 5-2
 sharing using NDU, 1-4
 special considerations, 5-11

Displaying
 file contents, 4-44

DOS
 file attributes, 4-5

DOS operating system
 restrictions with FAL, 3-4
 single-tasking, 3-1

DTR
See DECnet Test Receiver

E

Emulations
 NETBIOS, 1-5
 VT100, 1-4
 VT200, 1-4
 VT220, 1-4
 VT320, 1-4
 VT52, 1-4

End nodes, 1-2

Entities
 for the NDU, 5-8

Environments
 in Microsoft Windows, 1-5, 4-3

Error messages
See also Data Access Protocol
 command line, B-3
 execution, C-12

Error messages (cont'd)
 file input/output, B-7
 for the NFT DAP Maccode field, D-1
 for the NFT DAP Miccode field, D-2
 for the NFT DAP protocol, D-1
 in NDU, C-1, C-2
 input/output, B-2
 miscellaneous, B-14
 network, B-3
 network related, B-9
 on command lines, B-2
 secondary, B-11
 warnings, B-6, B-25

ERRORLEVEL
 values explained, 4-2

Ethernet address, 1-2

Executor nodes, 1-2

EXIT command
 in NFT, 4-19, 4-35

Exiting
 NFT, 4-7

F

FAL
See File Access Listener

FAL.EXE executable
 running Microsoft Windows, 3-2

FALNOWIN.EXE file
 saving space without windows, 3-2

File Access Listener (FAL) utility, 1-5
 accessing files, 3-3
 algorithm used with, 4-12
 as an object, 3-3
 command line, 3-2
 command syntax, 3-5
 functions of, 3-1
 invoking, 3-1
 Job Spawner support, 2-1
 performance, 3-4
 remote access, 1-4
 restrictions with DOS, 3-4
 sample command line, 2-6
 starting, 3-2
 stopping, 3-2
 switches, 3-6

File Access Listener (FAL) utility (cont'd)

- syntax problem, 3-4
- three uses of, 3-2
- tuning, 3-4
- using with Microsoft Windows, 3-2

Files

- access using the FAL utility, 3-3
- appending, 4-15, 4-20
- ASCII, 3-5, 3-6, 4-12, 4-13
- batch, 4-18
- binary, 3-5, 3-6, 4-12
- copying, 4-13, 4-15, 4-24
- copying local, 4-11
- copying multiple, 4-29
- copying remote to local node, 4-13
- copying to remote nodes, 4-16
- DDIF type, 4-13, 4-21
- deleting, 4-16, 4-32
- displaying contents of, 4-17, 4-44
- DOTS type, 4-13
- image, 4-12
- input/output error messages, B-2, B-7
- local specifications, 4-11
- logging operations, 4-18
- manipulating, 4-14
- MS-DOS attributes, 3-4
- name specification, 2-2
- naming specification, 4-11
- Network File Transfer (NFT) operations, 4-14
- nonstream, 4-12
- printing, 4-17, 4-38
- remote specifications, 4-11, A-1
- remote to local, 4-15
- specifying remote, A-1
- stream, 4-12, 4-22, 4-28
- temporary, B-9
- transfer of, 4-1
- types of, 4-12
- types of systems, 4-12
- using wildcards, A-4
- wildcard specification, A-2

Foreground

- tasks, 2-1

H

HELP command

- in NDU, 5-7, 5-10, 5-18
- in NFT, 4-4, 4-6, 4-19, 4-36
- switches, 4-36

HELP SHOW command, 4-6

I

Image files, 4-12

/IMAGE switch

- in NFT, 4-13

Information

- displaying directory, 4-33
- own use only, 1-2
- point of origin, 1-2
- sharing, 1-2

Installing virtual device programs, 5-5

Interfaces

- factors in choosing, 1-5
- NETBIOS, 1-5
- programming applications for, 1-4
- task-to-task, 1-5

Invoking

- See also* Starting
- command files, 4-18
- FAL, 3-1
- Job Spawner and FAL, 3-2
- the Job Spawner utility, 2-5
- the NFT utility, 4-2

J

Job Spawner utility, 2-1

- database, 2-1
- DECOBJ.DAT database, 2-1
- on Microsoft Windows, 2-5
- overview, 2-1
- remote access, 1-4
- service program limitations, 2-1
- starting, 2-5, 3-2
- using batch files with, 2-7

L

- LAN
 - See* Local area network
- LIST command
 - in NCP, 2-4
- Local area network (LAN), 1-2
- Local files
 - copying to a remote node, 4-11, 4-12, 4-14, 4-16
 - naming, 4-11
 - providing access, 3-1, 4-1
- Local nodes, 1-2
- Logging
 - file operations, 4-18
 - LOGFILE.TXT, 3-6
 - Spawner messages, 2-5

M

- Macro
 - system calls, 1-5
- Messages
 - command line errors, B-2
 - input/output, B-7
 - miscellaneous, B-14
 - network errors, B-3
 - network related, B-9
 - secondary error messages, B-11
 - warnings, B-6, B-25
- Microsoft Windows, 1-5
 - FAL, 3-2
 - using
 - the NFT utility, 4-3
 - using FAL, 3-2
- Modules
 - software components, 1-2
- MS-DOS
 - file attributes, 3-4
 - operating system, 1-2
- Multiple commands, 4-3

N

- Naming
 - virtual devices, 5-4
- NCP
 - See* Network Control Program
- NDDRV.SYS device driver, 5-5
 - See also* Network Device Utility
- NDU
 - error messages
 - command line, C-2
 - execution, C-5
 - virtual printer, C-10
- NETBIOS
 - emulation, 1-5
- Network Control Program (NCP) utility, 1-4
 - DEFINE ACCESS command, 3-3
 - DEFINE OBJECT command, 2-2, 3-3
 - functions defined, 1-4
 - LIST command, 2-4
 - LIST OBJECT command, 2-2
 - PURGE KNOWN ACCESS command, 3-3
 - PURGE OBJECT command, 2-2
 - PURGE OBJECTS command, 2-4
 - security, 4-9
- Network Device Utility (NDU)
 - CLOSE command, 5-7, 5-12
 - command format, 5-6
 - command line error messages, C-10
 - command verbs, 5-7
 - CREATE command, 5-7, 5-14
 - defined, 5-1
 - DELETE command, 5-7, 5-17
 - disk status display, 5-22
 - ending connection with remote disk, 5-12
 - error messages, C-1, C-2
 - execution error messages, C-5, C-12
 - EXIT command, 5-6
 - getting help for, 5-18
 - HELP command, 5-7, 5-10, 5-18
 - network error messages, C-11
 - network errors, C-3
 - OPEN command, 5-7, 5-20

Network Device Utility (NDU) (cont'd)

- printer status display, 5-22
 - prompts, 5-6
 - running, 5-5
 - sharing devices, 1-4
 - SHOW command, 5-8, 5-22
 - virtual printer error messages, C-10
- ## Network File Transfer (NFT) utility, 1-5
- See* DECnet-DOS utilities (NFT)
- abbreviating
 - commands, 4-4
 - switches, 4-4
 - access control information, 4-8
 - APPEND command, 4-15, 4-19, 4-20
 - command
 - format, 4-4
 - command parts explained, 4-3
 - command prompts, 4-5
 - command summary, 4-19, 4-20, 4-44
 - commands, 4-32
 - COPY command, 4-15, 4-19, 4-24
 - COPY switches, 4-25
 - Data Access Protocol (DAP)
 - error messages, D-1
 - defined, 4-1
 - DELETE switch, 4-32
 - directories display, 4-8
 - DIRECTORY command, 4-16, 4-19, 4-33
 - displaying access information, 4-11
 - ending operation in, 4-7
 - entering commands, 4-3
 - exchanging files, 1-4
 - EXIT command, 4-19, 4-35
 - exiting from, 4-3
 - file operations, 4-14
 - from command line, 4-2
 - from Microsoft Windows, 4-2
 - functions, 4-1
 - getting help, 4-6
 - HELP command, 4-4, 4-6, 4-19, 4-36
 - /IMAGE switch, 4-14
 - invoking, 4-2
 - Maccode field error messages, D-1
 - manipulating files, 4-14
 - Miccode field error messages, D-2

Network File Transfer (NFT) utility (cont'd)

- PRINT command, 4-19, 4-38
- /PRINT switch, 4-17
- SET command, 4-8, 4-19, 4-40
- SHOW command, 4-19, 4-41
- starting, 4-3
 - from command line, 4-2
- SUBMIT command, 4-19, 4-43
- switches, 4-5, 4-33, 4-36, 4-43
 - /BRIEF, 4-16
 - /IMAGE, 4-13
 - /NOCONVERT, 4-13
 - /PRINT, 4-38
- TYPE command, 4-19, 4-44
- using
 - a command file with, 4-2
 - switches, 4-5
 - with COPY, 4-12
 - with Microsoft Windows, 4-3
 - with SET, 4-8
 - with SHOW, 4-11

Network Management Listener (NML) utility, 1-4

Networks

- environment, 1-1
- error messages, B-3
- error messages on, C-11
- errors, C-3
- exiting from, 4-35
- local area, 1-2
- local nodes, 1-2
- maintaining, 1-4
- managing functions, 1-2
- monitoring, 1-4
- nodes defined, 1-2
- remote access, 1-4
- remote nodes, 1-2
- restrictions, 3-4
- routing nodes, 1-2
- sample of, 1-2
- sharing devices (NDU), 1-4
- using executor nodes, 1-2
- using FAL and TFA, 1-4
- using the NML utility, 1-4

NFT

error messages

See Appendix A

NFTNOWIN.EXE space saver

when to use, 4-3

NML

See Network Management Listener

/NOCONVERT switch

in NFT, 4-13

Nodes

access control information, 4-7

adjacent, 1-2

automatic remote access, 1-4

copying file to remote, 4-11

copying files to, 4-16

creating applications for, 1-4

defined, 1-2

end, 1-2

executor, 1-2

local, 1-2

remote, 1-2

remote printers on, 5-4

routing, 1-2

types of, 1-2

Non-Microsoft Windows

FAL, 3-2

Nonstream, 4-12

file system, 4-13

record attributes, 4-13

NPDRV.SYS printer driver, 5-5

O

Objects

DECnet range, 2-2

FAL, 3-3

predefined, 2-3

size of database, 2-2

Spawner, 2-1

Octal, B-1

OPEN command

in NDU, 5-7, 5-20

Operating systems

DOS, 1-2

VMS, 1-5

Operating systems (cont'd)

VMS path specifications, 3-4

P

Passwords

eliminating, 4-7

size of, 4-7

Percent sign

using as a wildcard, A-3

Performance

improving, 3-4

Personal computer

as a node, 1-2

as a server, 2-1

file attributes on FAL, 3-4

handling connection requests with, 2-1

monitoring other systems with, 1-4

using NETBIOS on, 1-5

PRINT command

in the NFT utility, 4-19, 4-38

/PRINT switch

in NFT, 4-17

Printers

accessing remote, 5-1

creating remote queues, 5-14

installing drivers, 5-5

remote, 5-3

using NDU, 1-4

Printing

creating remote queues, 5-14

files, 4-17, 4-38

using the /PRINT switch, 4-17

Privileges

in the FAL utility, 3-4

Programming

creating applications, 1-4

DECnet-DOS library of routines, 1-5

MACRO language, 1-5

servers, 2-4

task-to-task communications, 1-5

Prompts

NDU, 5-6

NFT utility, 4-5

starting the Job Spawner utility from
system, 2-5

PURGE KNOWN ACCESS command
in NCP, 3-3
PURGE OBJECT command
in NCP, 2-4

Q

Question mark
using as a wildcard, A-2

R

Recording
logging file operations, 4-18
Records
variable-length, 4-13
Remote files
accessing, 4-1, A-1
copying to a local node, 4-13, 4-15
printing, 4-17
specifying, A-1
Remote nodes, 1-2
Remote printers, 5-4
closing down connection, 5-12
creating a print queue, 5-14
Routing nodes, 1-2

S

Saving
access control information, 4-9
Security
See also Access control information
access checking, 3-4
access control
password length, 4-7
adding extra, 4-7
changing default control information,
4-10
default information, 4-9
defining node access information, 4-7
displaying access information, 4-11
maximum number of nodes, 4-8
password display, 4-41
setting access control, 4-40

Security (cont'd)
using with NCP, 4-9
Server functions
using Job Spawner
with DTR, 2-1
with FAL, 2-1
Servers
user written, 2-1
using personal computers as, 2-1
Services
multiple, 2-1
Sessions
ending FAL, 3-2
ending NDU, 5-6
ending NFT utility, 4-7
SET command
in NFT, 4-8, 4-19, 4-40
SHOW command
in NDU, 5-8, 5-22
in NFT, 4-19, 4-41
Sockets
closing, 2-7
Job Spawner, 2-4
Software
DECnet, 1-1
managing network, 1-4
modules, 1-2
Software product description (SPD), 1-1
Spawner
See Job Spawner utility
SPAWNER command
invoking, 2-5
SPAWNER LOG
messages, 2-5
SPD
See Software product description
Starting
See also Invoking
the FAL utility, 3-1
the Job Spawner utility, 2-5
Stopping sessions
in FAL, 3-2
in NDU, 5-6
in NFT, 4-7

Streams

- file system, 4-13
- record attributes, 4-13

SUBMIT command

- in NFT, 4-43
- NFT command, 4-19

/SUBMIT switch

- NFT utility, 4-18

Subroutines

- library of, 1-5

Switches

- abbreviating Network File Transfer, 4-4
- /ASCII, 3-6, 4-6
- /Binary, 3-6
- /BRIEF, 4-16
- DELETE, 4-32
- IMAGE, 4-13
- in the FAL utility, 3-6
- Network File Transfer (NFT), 4-3, 4-5
- NFT HELP, 4-36
- /NOCONVERT, 4-13
- PRINT, 4-17
- SUBMIT, 4-18
- using PRINT, 4-38
- using SUBMIT, 4-43

Syntax

- copying files, 4-15
- deleting files, 4-16
- in NDU, 5-6
- in the FAL utility, 3-5
- in the NFT utility, 4-3, 4-20
- running command files, 4-18
- specifying remote files, A-1
- using the NFT utility, 4-4
- using the TYPE command, 4-17
- using wildcards, 4-17
- VMS and DOS compatibility, 3-4
- when to use /SUBMIT, 4-18

Systems

- multiple service with remote, 2-1
- types of, 4-12
- VMS, 1-5, 4-5

T

Task-to-Task communications

- programming, 1-5

Tasks

- foreground, 2-1
- single, 3-1
- using DECnet, 1-2

TFA

- See* Transparent File Access utility

Transparent File Access (TFA) utility

- description of, 1-4
- exchanging files, 1-4

TYPE command

- in NFT, 4-19, 4-44

U

Utilities

- FAL, 1-5, 3-1
- Job Spawner, 2-1
- NDU, 5-1
- NFT, 1-5, 4-1, 4-4

V

Variable-length

- records, 4-13

Virtual devices

- connecting to virtual disks, 5-20
- creating disks, 5-14
- deleting disk space, 5-17
- directing text, 5-5
- disks, 5-4
- installing device drivers, 5-5
- printer errors, C-10
- printers, 5-4
- remote disk volume names, 5-2
- remote disks, 5-3
- remote printers, 5-3
- special considerations, 5-11

- VMS system, 1-5

W

Warnings

error messages, B-6, B-25

Wildcards

asterisk, 4-17, A-2, A-4

combining, A-2

limitations in using, 4-14

percent sign, A-3

question mark, A-2

Windows

running the Spawner command from, 2-5

with DECnet-DOS utilities, 1-5

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

Electronic Orders

To place an order at the Electronic Store, dial 800-DEC-DEMO (800-332-3366) using a 1200- or 2400-baud modem. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Your Location	Call	Contact
Continental USA, Alaska, or Hawaii	800-DIGITAL	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	809-754-7575	Local Digital subsidiary
Canada	800-267-6215	Digital Equipment of Canada Attn: DECdirect Operations KAO2/2 P.O. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6
International	_____	Local Digital subsidiary or approved distributor
Internal ¹	_____	USASSB Order Processing - WMO/E15 or U.S. Area Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

¹For internal orders, you must submit an Internal Software Order Form (EN-01740-07).

Reader's Comments

PATHWORKS for DOS
DECnet User's Guide

AA-PAFGC-TK

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

I would like to see more/less _____

What I like best about this manual is _____

What I like least about this manual is _____

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

I am using **Version** _____ of the software this manual describes.

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

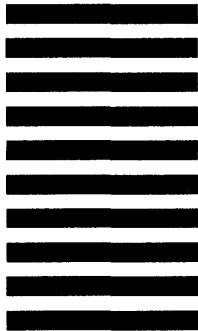
Phone _____

- Do Not Tear - Fold Here and Tape -

digital™



No Postage
Necessary
If Mailed
in the
United States



BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
Telecommunications and Networks
LKG1-3/L12
550 KING STREET
LITTLETON, MA 01460-1289



- Do Not Tear - Fold Here -

digital

Order Number: AA-PAFGC-TK
Printed in the U.S.A.