erver
work

Server Network

VICTOR Publications
380 El Pueblo Road
Scotts Valley, CA 95066
(408) 438-6680

TRADEMARKS

NOTICE

Preliminary VICTOR printing July, 1983.

ISBN 0-88182-089-5 Printed in U.S.A.

IMPORTANT SOFTWARE DISKETTE INFORMATION

For your own protection, do not use this product until you have made a backup copy of your software diskette(s). The backup procedure is described in the user's guide for your computer.

Please read the DISKID file on your new software diskette. DISKID contains important information including:

- The part number of the diskette assembly.
- The software library disk number (for internal use only).
- The product name and version number.
- The date of the DISKID file.
- A list of files on the diskette, with version number, date, and description for each one.
- Configuration information (when applicable).
- Notes giving special instructions for using the product.
- Information not contained in the current manual, including updates, any known bugs, additions, and deletions.

To read the DISKID file onscreen, follow these steps:

1. Load the operating system.
2. Remove your system diskette and insert your new software diskette.
3. Enter--

   TYPE DISKID

4. The contents of the DISKID file is displayed on the screen. If the file is large (more than 24 lines), the screen display will scroll. Type ALT-S to freeze the screen display; type ALT-S again to continue scrolling.

CONTENTS

SECTION I. OPERATOR'S GUIDE

SECTION IV. HARDWARE CONSIDERATIONS

FIGURES

TABLES

PREFACE

The VICTOR Server Network connects a number of desktop computer stations, including diskless stations, to printers and hard disks.

This manual is divided into four sections:

- Section I teaches a Network station operator how to use the Network commands that control his or her Network resources.

- Section II explains how to install the Network software. This includes configuring the servers, installing the stations, and adding users to the Network.

- Section III contains information for the programmer who wants to tailor his or her programs for the Network.

- Section IV contains hardware information.

1. INTRODUCTION TO THE SERVER NETWORK

Welcome to the Server Network. The Server Network enhances the capability of your computer by giving it:

- Considerably increased file storage.
- Private and shared files.
- Printers at various convenient locations.

You will find that you can use the Network efficiently and easily by adding just ten Network commands to your MS-DOS command vocabulary.

The Network consists of a set of computers that are physically connected. The Network computer programs control how and where data is stored, and how data is passed from one computer to another. The Network uses two kinds of computers:

- A server, which is a central computer with a hard disk for mass storage. Usually, it has no keyboard or screen. A server usually is connected to printers, which are located at strategic locations throughout the Network area.
- A station, which is a computer that you use to send data and commands to a server. You -- the Network user -- control where your files are stored and printed by typing MS-DOS and Network commands at your station keyboard.

independent of the Network.

Before using this manual, you should be familiar with the common MS-DOS commands that operate your computer in local mode (not connected or "logged in" to the Network). If you don't know how to use MS-DOS yet, read the MS-DOS section of the Operator's Reference Guide. Then continue with this manual.

The purpose of this section is to show you how to use the ten Network operating commands. The chapters in this section contain the following information:

Chapter 1 -- "Introduction to the Server Network." Describes the purpose of this section. Explains what the Server Network is and how it works.

Chapter 2 -- "Entering and Leaving the Server Network." Describes how to access and leave the Network. Explains the difference between Network and local modes, and tells which MS-DOS commands are applicable to the Network, which ones are not, and which are specific Network commands.

that some Network files need some special handling such as reserving them for updating purposes. Describes how to reserve and release shared files, and how to limit access to them.

Chapter 4 -- "Obtaining Network Information." Shows how to obtain information about the status of the Network and about other Network users.

Chapter 5 -- "Printing Files." Explains how to print files on the Network and local printers.

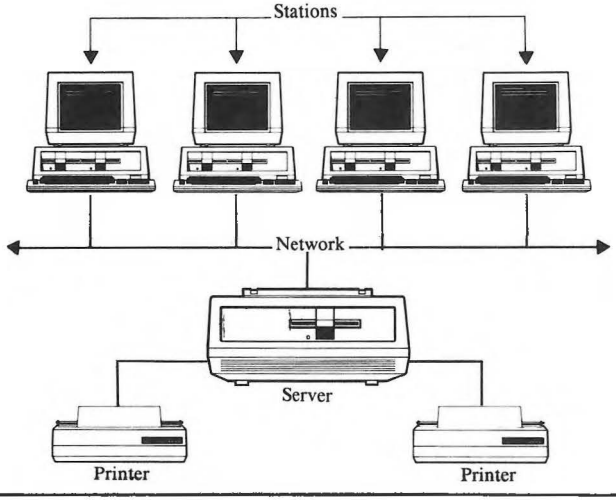
Other manuals you may find useful are the Operator's Reference Guide and the Hard Disk Tool Kit.

1.1 THE NETWORK

The Network consists of:

- As many as 54 Network stations at which users enter local and Network commands and data.
- As many as ten servers which provide large amounts of storage for private and shared files.
- Printers (up to three per server), the use of which can be specified by the station users.

Figure 1-1: A Simple Network Configuration



The resources of the servers are assigned to users, not to the physical stations. As far as the Network is concerned, a user can use any station he or she wants.

1.2 THE SERVERS

A server is a mass storage device -- a combination of software and hardware -- used to store large amounts of data. The software consists of an MS-DOS 2.0 operating system and special programs that provide access to the stored files and allow your files to be printed.

disk drive can be used for archiving files and for transferring files from other networks. It should not be used for general file activity because it slows the Network's performance. The hard disk stores users' files. The software that runs the server is also stored on the hard disk.

Up to ten servers and 54 stations can be linked together with twisted-pair cables. Each server can be connected to as many as three printers, using the two serial and one parallel interfaces.

At installation, each server's hard disk can be divided into one, two, four, or eight equal-sized sections called "volumes." The volumes are given letter designations so that you can access them as if they were actual drives. The person who does the installation can divide the hard disk space unevenly if necessary, but this requires the Hard Disk Tool Kit.

You can have access to one, two, or many Network disk drives on one or more servers. You may also have access to the Network floppy disk drives on the servers.

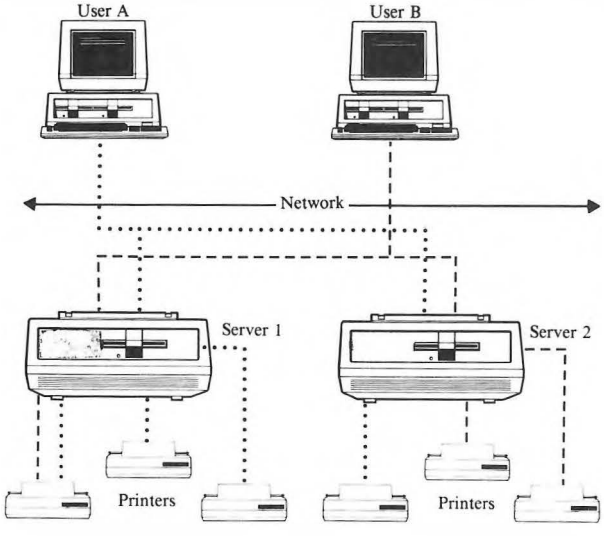
1.3 THE PRINTERS

Each server is connected to as many as three printers. You can have up to four printers assigned to you. All four may be Network printers, or as many as three of them can be local

automatically unless you specify otherwise.

Table 1-2 shows how two users might access two servers and seven printers.

Figure 1-2: Sample Printer/User Configuration



User A can access four of the Network printers. User B can access three of the Network printers and his or her own local printer. Network printers may be shared by any number of users.

They are physically connected to the Network by twisted-pair cables. Some stations have floppy or hard disk drives, some have both, and some have neither. Those without disk drives are called diskless stations.

If your station has disk drives, you engage the Network by turning on your station and inserting a Network MS-DOS System diskette into drive A. This boots up or loads the system programs into your station's memory from the diskette. If your station has a hard disk, both the hard disk boot and the Network software may be on the hard disk. You reference your Network disk drives by specifying their drive letters. For example, if A and B are local drives, then C, D, E, and F might be your Network drives.

1.5 LOCAL AND NETWORK MODES

Your station operates in both local and Network modes. In local mode, you use only the resources of your own station. To use your station in local mode, boot up with a standard MS-DOS System diskette, or boot up with the Network System diskette but reference only your local disk drives.

If you have a diskless station you cannot operate it in the local mode because the only drives you can access are Network drives. When you turn on your diskless station, the Network's software automatically boots your station and puts you onto the Network.

affect the Network. Most commands affect both the Network and your local station. For example, DIR can display only a listing of your local files when you are in the local mode. When you are logged on, DIR can show a listing for your files, either local or on the Network.
The only MS-DOS commands that cannot be applied to the Network are FORMAT, DCOPY, SYSCOPY, and CHKDSK. You can use them, however, for your local drives.

All the Network commands apply only to the Network. These commands are LOGIN, LOGOUT, NETPRINT, NETSTAT, NETUSERS, PASSWORD, PROTECT, RELEASE, RESERVE, and STATION.

1.6 SHARING FILES

The Server Network lets several users share files. Certain drives are designated private while others are designated as shared. You write files you want to remain private to your private Network disk drive. You write files you want to share to a shared disk drive. Shared files, for example, can be very useful when several people are responsible for entering data in an inventory data base.

The system prevents two or more people from writing to a file at one time. If two people attempt to edit a file at the same time, one of them receives an error message when he or she begins updating.

In this chapter, you'll learn how to enter, leave, get help from, and change your password on the Server Network.

2.1 ACCESSING THE NETWORK

You can access the Network only from a computer (station) that is physically wired to the Network. If you have trouble accessing the Network, make sure that you are using a Network station. If you still have trouble, be sure that at least one server is turned on.

If you are using a station with disk drives, you must boot up from a Network MS-DOS System diskette. Follow this procedure:

1. Turn on your station and insert the Network Station System diskette.

2. Wait for a message from the Network. You will be logged in automatically or you will be prompted for your user name.

NOTE FOR DISKLESS STATION USERS: Accessing the Network from a diskless station is simply a matter of turning on the station or pressing the Reset button. If this doesn't work, check that a server is turned on, and the Network cable is connected to your station.

your Network station and access the Network.

- You are automatically logged in to the Network. Start working as usual.

- A log-in prompt appears on the screen asking for your user name. If you are new to the Network, you will have to ask what your user name is.

- A log-in prompt appears on the screen asking for your password. If you are new to the Network, you will have to find out what your password is.

- Log-in prompts appear on your screen asking for your user name and your password.

NOTE: The MS-DOS AUTOEXEC capability is available to Network users. Your Network station diskette contains the file NETWORK.BAT. You can put AUTOEXEC commands into this file. Remember to include a LOGIN command. For more information about AUTOEXEC, see the MS-DOS portion of your Operator's Reference Guide.

2.2.1 AUTOMATIC LOG-IN

If the automatic log-in feature has been activated for your station, you are automatically logged in when you turn on your station. This feature works on diskless as well as disk-equipped stations.

another valid drive.

Automatic log-in can be set for more than one station on the Network. If you want to use another station, it might be set up for automatic log-in or you might have to use one of the manual procedures that follows to log in.

2.2.2 MANUAL LOG-IN

After you first access the Network, this prompt appears on the screen:

Enter your user name:

Type your user name and press Return. Your user name must contain no spaces and it may not exceed 11 characters. The Network displays the Network banner if the log-in procedure is complete.

The Network might ask for your password. It displays:

Enter your password:

Type your password and end it with a Return. For additional security, the password is not displayed but an x on the screen shows your position. You can use the BACKSPACE or DEL key to correct an error in your entry. You can also type ALT-X to erase your entire entry and start all over.

On rare occasions, you may be asked to enter the date and time. This happens when you are the

NOTE: You must enter both the date and time; the program does not accept a Return. Use ALT-C to start the log-in procedure over again if necessary.

The date and time you enter will be used for the entire Network, so you must be careful.

After the Network accepts your log-in, it displays the Server Network banner.

If you need to know your drive assignments, type the command NETSTAT after the MS-DOS prompt. If you need to know your printer assignments and their status, type the command NETPRINT after the MS-DOS prompt.

2.3 LEAVING AND REENTERING THE NETWORK

You may want to leave the Network for several reasons: you may want to turn off your station, reboot with a CP/M-86 or MS-DOS system diskette, or log in another person.

2.3.1 USING THE LOGOUT OR LOGIN COMMAND

Use the Network command LOGOUT or LOGIN when you expect to leave your station unattended for a while, or if you want to lend your station to another person. These commands do exactly the same thing, so you can use whichever one you prefer.

```
LOGOUT
```

or

```
LOGIN
```

LOGOUT or LOGIN releases your reserved files (if any), removes you from the Network, and asks for another user name. If no one wants to log in, type ALT-C, reboot, or turn off your station.

The new person (or perhaps it is you again) uses the new log-in display to log in. When the other person is finished, he or she types LOGOUT or LOGIN to return the station to you or to yet another person. Even if you have automatic log-in on your station, you now have to type your user name to have the Network return the station to you.

A faster method of using LOGOUT or LOGIN is to type the command and new user name on the same line. For example,

```
LOGOUT JONES
```

or

```
LOGIN JONES
```

logs in a user named JONES. If a password is required, you still have to enter it separately.

2.3.2 LEAVING THE NETWORK WITHOUT LOGGING OUT

You can leave the Network simply by turning off your station or rebooting it with another system diskette. This can cause problems, however, so it is not the recommended way to leave the Network.

If you have reserved files and you disconnect from the Network without using LOGOUT or LOGIN, your reserved files are not released. Therefore, if you plan log off by turning off your station you have to give the RELEASE command first to release your reserved files. (Reserving and releasing files is discussed in Chapter 3.) It's a lot safer to get into the habit of always using LOGOUT or LOGIN.

2.4 CHANGING YOUR PASSWORD

You may want to (or be required to) change your password periodically. This is the Network's method of maintaining security for your files. Use the PASSWORD command to change your password.

Type:

```
PASSWORD
```

at the MS-DOS prompt. The Network responds by displaying your user name. It then asks for your old password if you have one. (To create a new password, you must know your old one. If you have forgotten your password, see your Network administrator.)

The Network asks you to type it again for verification.

Press the ACCEPT function key to install your new password. From now on, you will type the new password whenever you log in.

Remember this: No one knows your password. If you forget it, you can't log on to assign yourself a new password. If you do forget your password, go to the Network administrator. Although that person doesn't know your password, he or she can use a special program to assign you a new one. Now you can log on to the Network with your assigned password. Then you may want to use PASSWORD to change your password again so that you have complete password privacy.

2.5 GETTING HELP FROM THE NETWORK

Some times you might forget the exact format or use of one of the Network commands. The Network provides you with helpful information about the commands if you type the command followed by /HELP. For example, if you want to find out about the RESERVE command, type:

```
RESERVE /HELP
```

The Network displays the RESERVE command's format and tells what it does.

keeps you from accessing one of your servers. If this is the case you receive this message:

LOGIN error: Enter R:retry, A:abort, I:ignore

where:

R causes the Network to try to log you in again.

A aborts the log-in process. (You can also use ALT-C for this purpose.)

I continues the log-in procedure, but does not include the offending server as one of your servers.

Type R, A or ALT-C, I in response to this message.

A message you might receive when you enter any of the ten Network commands is:

SERVER {missing / no return} nnnn hh...h R:retry, D:detach, I:ignore

where:

nnnn is the server's number.

hh...h is a diagnostic code to be used by Field Engineering personnel if necessary.

command again.

D removes the offending server, its printers, and drives from your user name until you LOGOUT or LOGIN. This means that you can continue working on the Network, but you cannot access anything on that server.

I causes the Network to ignore the error condition. This is not recommend, but it may be useful in some applications.

Type R, D, or I in response to this command.

user names. You will learn that by using your private drive(s), reserving shared files for your exclusive use, and controlling the read/write status of files, you can avoid most multiple access problems.

At times you want to keep certain files private so that only you have access to them. For example, you don't want employee salary information to be generally available. You will want to share other files. Certain video input forms should be available to all (or several) users. And you will want to allow limited access to some files; you might want to create a file of instructions on a new office procedure that other users could read but not change. When you need to modify a shared file, you will want to reserve it in advance so that no one else can access it at the same time.

3.1 USING YOUR PRIVATE AND SHARED DISK DRIVES

You decide whether a file is private or sharable by where you store it. Remember, the log-in display shows which disk drives are private and which are shared. (You may not have a private disk drive, but you may find it useful to understand the concept.) If the file is to be a private file, place it on your private disk drive. Only you can access it for reading and writing and its name appears only in your directory.

If you want to create a file that can be shared by other users of the Network, place the file on a shared disk drive. Any person who has access to

a time can update it.

3.2 SHARING USER NAMES

Several people can be logged in to the Network with the same user name at the same time. For example, everyone in the accounting department might use the user name, ACCT, to access the Network from the several stations in the department. The accounting department may have both shared and private drives. All the accounting employees have access to the files on the private drive(s). Accounting might share one drive with the payroll department. In addition to the ACCT user name, each person in the accounting department may also have a specific user name for his or her own files.

What if no one should be allowed to change a certain file? What if you need to modify a file and no one should access it while you are modifying it? The PROTECT and RESERVE commands help you handle these situations.

3.3 PROTECTING FILES

You can specify the READ or WRITE status of a file with the PROTECT command. READ means that anyone who has access to the file can read it (display the contents) but cannot make changes to it or delete it. WRITE means that you can both read a file and make changes to it. WRITE is the default status for all files when they are created.

on a private disk, there might be good reason to write-protect certain files on that disk.

The format of the PROTECT command is:

```
PROTECT filespec /status
```

where:

filespec is the optional drive name, the filename, and the extension. If you don't specify the extension, all extensions are matched. If filespec is omitted, PROTECT functions as if *.* were used.

/status is either READ or WRITE. If you leave out /status, the command displays the read/write status of the file.

"Wild cards" are acceptable in Network commands; they work the same way as they do for MS-DOS commands. To set the READ status for all .DAT-type files on drive E, type:

```
PROTECT E:*.DAT /READ
```

You can now display, print, and copy all the .DAT files on drive E, but you cannot write to, add to, or delete them.

To return a file to its normal read/write status, specify WRITE in the command. For example:

To display the status of all the files on drive C, you can use PROTECT with just the drive name:

```
PROTECT C:
```

Anyone with access to a drive can change the read/write status of the files on it.

3.4 RESERVING FILES FOR UPDATING

Whenever you want to modify a file, you should reserve it for your exclusive access in advance. The RESERVE command gives you temporary private access to one or more files. This command prevents two people from attempting to modify a file at the same time.

You can have only one reservation for one or more files in effect at a time. A reservation is valid only for the length of time you are logged on or until you release the files with the RELEASE command.

The RESERVE command's format is:

```
RESERVE filespec [,filespec ...]
```

If you want to reserve JONES.DAT on drive E and SMITH.PAS on drive G for update, type:

```
RESERVE E:JONES.DAT,G:SMITH.PAS
```

reservations are denied and the names of the files causing the problem are displayed.

You are asked if you want to retry. Type R if you do; type ALT-C if you do not.

If some of the files you want to reserve remain unavailable, enter a new RESERVE command that does not include the unavailable file(s), or try again later.

Any new RESERVE command rescinds the previous RESERVE command if you have one in effect. If you decide to add more files to your reservation list, you must include all the previously reserved files as well as the new ones in your new command. You can completely change your reservation list by listing only new filenames in your new RESERVE command.

3.5 RELEASING RESERVED FILES

The RELEASE command releases your reserved files. Remember that files are not released automatically when you reboot or turn off your station so you must use the RELEASE command in advance. To release your files, type:

```
RELEASE
```

It is important to release your reserved files as soon as possible after working with them because other people may want to use them.

information about your drive assignments and about other Network users.

When you are in local mode, you should know how much disk space you have left and what the names of your files are. When on the Network, you need similar kinds of information to make the best use of the Network resources. The Network commands STATION, NETSTAT, and NETUSERS let you obtain Network information.

4.1 DISPLAYING STATION AND SERVER NUMBER INFORMATION

You can use the STATION command to determine your own station's station number, what other stations are present, and whether a certain server is turned on.

The format for the STATION command is:

STATION [number]

where:

number — is the number of a station or server. If you omit number, the the number of your own station is displayed.

Servers are numbered 0 through 9. Stations are numbered 10 through 63. Most organizations place stickers on the Network computers to designate their numbers.

```
STATION
```

The Network displays:

```
This station's number is __.
```

To find out if station 21 is present, type:

```
STATION 21
```

The Network responds with one of three messages:

```
Station 21 is present.

Station 21 is not responding.

Station 21 is this station.
```

The first message means that station 21 is turned on and connected to the Network. The second message means that station 21 is not connected to the Network right now or is turned off. It could also mean that station 21 does not exist. The third message means that you are sitting at station 21.

To obtain a list of all the stations currently connected to the Network, type:

```
STATION  /ALL
```

The Network responds with a list of stations and servers now connected to the Network.

drives, type:

NETSTAT

Your screen displays information such as:

NETWORK DRIVE STATUS

DRIVE ASSIGNMENT	SPACE UNUSED	SPACE USED	ACCESS STATUS	SERVER
A	------	------	-------	LOCAL
B	------	------	-------	LOCAL
C	8240 K	1760 K	shared	1
D	4234 K	766 K	shared	0
E	4850 K	150 K	private	1
F	70 K	450 K	floppy	1
G	2260 K	240 K	shared	2
H	488 K	2012 K	private	2
I	488 K	2012 K	shared	2
J	25 K	2475 K	shared	2
K	------	------	------	not assigned

where:

DRIVE ASSIGNMENT is a list of all the drives assigned to you.

space is available to all Network users who have access to the drive.

USED SPACE — is the total amount of space that has been used by all users of the drive to store files. For private drives, the space used is for your files only. Floppy disk drive information is dependent on the disk in the drive at the moment. Question marks are displayed if no diskette is in the drive.

ACCESS STATUS — tells whether the drive is shared or private. SHARED means that several designated users can access files on this drive. PRIVATE means that only you can access the files and that their names appear only in your directory.

SERVER — is the number of the server the drive is on. "LOCAL" means that the drive is a local drive (at your station). "not assigned" means that the drive is not used.

In the example no information is provided for drives A and B because they are local drives. If you want to find out about them, use the MS-DOS CHKDSK command.

NETSTAT C:

Information for drive C only is displayed.

4.3 DISPLAYING NETWORK SERVER STATUS

You can display status information for any one of the Network servers.

The format of the NETSTAT command is:

```
NETSTAT /SERVER=n
```

where:

n — is the number of the server for which you want information displayed.

To display the status of server 2, type:

```
NETSTAT /SERVER = 2
```

If you want information for all active servers, you can type:

```
NETSTAT /SERVER
```

or:

```
NETSTAT /SERVER=ALL
```

Here is a sample display:

```
Server Number .........0
Server Version.......2.00
Open Files.............12
Reserved Files.........3
Your Open Files.........0
Your Reserved Files....1
No of Print Jobs
 Parallel..............1
 Serial A..............7
 Serial B..............0
Printer Status
 Parallel............ON
 Serial A............OFF
 Serial B............OFF
```

where:

Server Number	is the number of a server whose information is being displayed.
Server Version	is the version of Network software that your Network uses.
Open Files	is the total number of files on the drive now in use.
Reserved Files	is the total number of files on the drive that have been reserved.
Your Open Files	is the number of files you are using.

Number of Print Jobs	is the number of files waiting to be printed.
Printer Status	is either ON or OFF. ON means that the printer is ready to print. OFF means that the printer is turned off, off-line, unplugged, or not ready to print.

You can also use another version of the NETSTAT command to display the version number of the Network interface in your station. Type:

```
NETSTAT /VERSION
```

4.4 DISPLAYING USER STATUS

To display the names of the users currently using the Network, type:

```
NETUSERS
```

This display appears:

```
Current Network Users

  User          Station

ACCOUNTING        20
BRUCE             15
DOUG              12
```

```
NETUSERS /ALL
```

The display you see might look like this:

```
        Configured Network Users

-----------------------------------------------
ACCOUNTING  BROWN   BRUCE   DOUG  JOHN  KATHYP
SCOTT       TONY    WALT
```

To display the names of all the Network users assigned to a certain server, specify the server's number. For example, to display all Network users assigned to server number 3, type:

```
NETUSERS /SERVER=3
```

on the Network printers.

The Network NETPRINT command lets you:

- Print files on the Network printer(s).
- Print files on your local printer (the printer attached to your station).
- Print multiple copies.
- Display information about printers and print queues.
- Cancel a print request.
- Temporarily hold up the output of a printout.

To use NETPRINT successfully, you may have to make some changes to your logical and physical device assignments with the MS-DOS SETIO command. These changes, however, are usually done for you at installation time.

5.1 UNDERSTANDING SPOOLING AND PRINT QUEUES

The term spooling is a hold-over from the early days of computing. We use it in this manual so that you can communicate with people in the computer industry.

Spooling is the act of copying a file to a temporary storage area. The file copy remains in the temporary area until it can be transferred to another slow-receiving device.

in the queue. As each file arrives in the spool storage area, it is given a number. Files are printed in the order of assigned numbers.

If the printer is attached to the server on which your file is stored, the original file is placed in the queue. Its status is changed to READ/ONLY while it is spooled or being printed. You can read the file, but you can't make changes to it.

If the file is on a different server from the printer, a copy of the file is spooled to the printer's server and the status of the original file does not have to be changed. You can update the original file while its copy is spooled or being printed. That way your original file is not tied up while waiting to be printed or while it is being printed. After the file is printed, the spooled copy is erased automatically.

Spooling and queuing are applicable to Network printers only. If you use the NETPRINT command to print a file on your local printer, it is printed directly from your stored file.

5.2 YOUR PRINTER ASSIGNMENTS

One or more printers is probably assigned to your user name. One of these printers is your "default printer." This means that when you tell the Network to print one of your files, it is automatically printed on your default printer unless you specify otherwise.

You receive a display that looks something like this:

```
Logical device      Physical device

     CON            TTY CRT BAT UL1
     AUXIN          TTY PTR UR1 UR2
     AUXOUT         TTY PTP UP1 UP2
     LST            TTY CRT LPT UL1
```

LST is the only logical device that applies to you because it is the one that refers to printers. UL1 is highlighted because it is your default printer.

Now you know the name of your default printer. You may not know, however, which physical printer is associated with the names in the display. Ask the installer which printer belongs to which name. And don't let the names -- CRT, for example -- fool you. They are common built-in designations, but they really don't mean anything to the computer. You know they are all printers, and the computer is content if it has a set of letters it recognizes.

To change the default printer assignment, use another form of the SETIO command. To change the printer default to the printer designated by CRT, type:

```
SETIO LST = CRT
```

5.3 PRINTING A FILE

Use the NETPRINT command to print files on the Network printers. This command also works for printing on your local printer.

You can print a file on the default (or previously assigned) printer.

The format for this command is:

```
NETPRINT filespec
```

where:

filespec — is the optional drive name, the filename, and the extension.

The specified file is spooled to the default printer's print queue. It is printed as soon as the files in front of it have been printed. You can use any standard MS-DOS wildcard specification to print groups of files.

You can use the NETPRINT command to specify the printer on which the file is to be printed. Use this method as an alternative to SETIO if you want only one file to be printed at a different location from the rest of your files. This method is useful if you want your files printed near you, but you want one file printed near another person so that he or she has easy access to the hardcopy.

Use this command format if you want to print files at a location other than your default location:

where:

filespec — is the optional drive name, the filename, and the extension.

prtname — is TTY, CRT, LPT, or UL1. **prtname** can also be 1 for TTY, 2 for CRT, 3 for LPT, or 4 for UL1.

Your file is spooled and queued for the specified printer and printed as soon as the files in front of it have been printed.

5.4 PRINTING MULTIPLE COPIES

To print more than one copy of the file, add to the NETPRINT command:

```
/COPIES = n
```

where:

n — is the number of copies to be printed. The maximum number of copies allowed is 127.

If you want to print three copies of the file WILSON.TXT on drive C on the LPT Network printer, you type:

```
NETPRINT C:WILSON.TXT /PRINTER=LPT /COPIES=3
```

Remember that printing multiple copies takes time, possibly preventing other users from using the printer.

5.5 DISPLAYING THE SPOOLER STATUS

The NETPRINT command without parameters or options displays the status of all the spoolers assigned to you. To display the spooler status, type:

```
NETPRINT
```

The resulting display shows spooler information listed by printer. It shows how many files are in the queue, who they belong to, their size, and whether each is currently printing or waiting to be printed.

The display you see might look like this:

```
Printer: 1 (TTY)  Date: 06/24/83   Time: 13.58
Server : 1        Port: Serial A   Status:  ON

Filename   FileSize  Username    Copies  Time   Queue Entry
-------------------------------------------------------------

Network.DOC  2368    BOB           1    12:30   Printing
MSDOS.DOC   10368    BOB           1    12:40   2
Letter.PRN    368    DOUG          1    12:42   3
Report.PRN   1024    ACCOUNTING    2    12:45   HOLD
Report1.PRN  5024    ACCOUNTING    2    12:46   5
```

5.6 DISPLAYING THE STATUS OF YOUR OWN PRINT FILES

This form of the NETPRINT command displays spooler status information also, but it applies only to your own print requests. Type:

```
NETPRINT /MY
```

want to display the status for the printer named TTY, type:

```
NETPRINT /PRINTER = TTY
```

or:

```
NETPRINT /PRINTER = 1
```

5.7 HOLDING A PRINT REQUEST

Use the /HOLD option with the NETPRINT command to suspend the printing of a file that is currently being printed, or to keep a queued file from being printed:

```
NETPRINT filespec /HOLD
```

where:

filespec	is the optional drive name, the filename, and the extension.

To resume printing or to put the file back in line to be printed, use this form of the command:

```
NETPRINT filespec /RELEASE
```

where:

filespec	is the optional drive name, the filename, and the extension.

command, printing stops, then restarts at the beginning of the file. You would use this command in case of a problem such as a paper jam, or if a new print element or ribbon is needed.

If you have more than one file in the queue with the same filename, any command parameter affects only the first file with that name.

NOTE: You cannot use the RESTART option on a file that is being "held" with /HOLD.

The format of this command is:

NETPRINT filespec [other parameters] /RESTART

where:

filespec — is the optional drive name, the filename, and the extension.

other parameters — is:

[/PRINTER=prtname [/COPIES = n]]

where:

prtname is TTY or 1, CRT or 2, LPT or 3, or UL1 or 4.

n is the number of copies to be printed (maximum is 127).

5.9 CHANGING PRINTERS

The RESTART option is also a convenient means of switching printers if you realize after spooling a file to one printer that another printer is available. To change the selected printer, use this format:

```
NETPRINT filespec /RESTART
     /PRINTER=oldprtname TO /PRINTER = newprtname
```

where:

filespec	is the optional drive name, the file name, and the extension.
oldprtname	is the name of the originally specified printer.
newprtname	is the name of the printer on which the file will now be printed.

5.10 CANCELING A PRINT REQUEST

Use this form of the NETPRINT command to cancel an active print request:

```
NETPRINT filespec /CANCEL
```

If the file is still in the queue, it is deleted. If the file is being printed, the printing is terminated and the file is erased from the queue.

This section explains how to install and use the Network-associated software that:

- Configures the server(s).
- Creates the station Network System disks used by each station.
- Adds user names to the Network.

NOTE: This section assumes that the entire Network is physically in place and properly connected. If you wish to add another station or server to your existing Network, read Section IV, the Hardware Considerations section of this guide. Physically install the station or server, then come back to this section and install the appropriate software.

You should read the first five chapters of this guide before using this section of the manual. If you understand how the operator uses the Network, you will have greater insight into some of the installation decisions you will be making. You should be acquainted with MS-DOS 1.25. The MS-DOS commands that you will use during installation are COPY, FORMAT, SYSCOPY, DCOPY, DIR, and DEL; they are not described in detail in this section. If you have problems, refer to the MS-DOS section of the Operator's Reference Guide.

You need a map of the positions of all the components of the Network -- the stations, the servers, and the printers connected to the servers. You should know the number of each server (0 through 9) and station (10 through 63). To avoid confusion, you might affix decals

STATION

and:

STATION /ALL

at each Network station to make sure that it is properly connected and has its proper address.

Finally, you need to know who is going to use the Network, what the relationship is among the users, what the user names and passwords will be, and how each user will use the Network.

The chapters in this section contain the following information:

Chapter 6 -- Introduction to the Installation Procedure. Explains the purpose of the section, and discusses what the installation procedure entails. Gives guidelines and discusses factors to be considered when configuring the system and assigning user resources.

Chapter 7 -- Configuring the Server(s). Describes how to make copies of the master Server Software disk, then explains how to use these disks to create the servers' hard disk volumes.

Chapter 9 -- Installing the Users. Explains how to install user names on the Network, and how to assign Network drives and printers to each user. Also explains how to modify the configurations of current users, and how to delete users.

6.1 CONFIGURATION CONSIDERATIONS

These considerations affect some of the decisions you make during the installation procedure:

- Initial number of users.
- Expected number of users.
- Types of usage.
- Sizes of files.
- Private and shared file needs.
- Private and shared user names.
- Physical locations of stations and printers.
- Network access security requirements.

A typical Network configuration might contain these components:

- Two servers.

- 49 users, one per station.
- One private disk drive per user.
- Three shared disk drives per user, two on one server, and the third on the other server for backup purposes.
- Two printers, one attached to each server.
- Both Network printers available to each user.

This is just an example of one configuration. All of the quantities are flexible and can be changed to meet individual Network needs. Quantities can be varied up to these maximums:

- Ten servers with:
  - -- Three printers per server.
  - -- 14 drives per server.
- 54 stations.
- 200 user names with:
  - -- Four printers per user name.
  - -- 15 drives per user name.

server, the Network response can be slow. You can ease this problem by dividing the hard disk resources carefully among users. Another way to handle the problem is to divide the Network into two separate Networks.

Generally, up to ten users can access a server at one time. For file-intensive work (where programs are being compiled and files are generated), perhaps only five users should be assigned disk drives on one server. Word processing, on the other hand, is not usually file-intensive, so as many as 20 users can access drives on a single server.

6.3 DETERMINING DRIVE SIZES AND USES

Part of the installation procedure includes dividing each server's hard disk into one, two, four, or eight separate sections called volumes. (Users perceive volumes as disk drives.) The size of the volumes you assign to each user should be determined by the size of the files to be stored. If more than eight volumes are required you must use the Hard Disk Tool Kit.

Large files should be stored on large volumes. Allow for growth in file size where appropriate. Allow, also, for backing up large files. (If files are too large to be backed up on a floppy disk, then they should be backed up on another

Another factor to consider is that users usually share volumes. As one user's directory grows, the other users' available space on that volume shrinks. If possible, consider in advance which users' directories might grow quickly and assign drive space accordingly. Otherwise, be ready to reassign volumes when users start being "squeezed out."

diskettes to configure each server's hard disk into volumes. These are your first tasks after the Network hardware has been set up and connected.

You have been provided with one Server Distribution diskette for each server. You should make a copy of one of these diskettes to act as a backup.

NOTE FOR DISKLESS STATIONS: If you have only diskless Network stations, you must use the server as a station until your Network is operational. To do this, attach a screen and keyboard from one of your diskless stations to the server and use a hard disk operating system with the Operator's Reference Guide to create a hard disk station. You will use this server as a station until you initialize it as a server. Make sure that you have made backups of all your disketttes before initializing the server. If you have more than one server, initialize the one being used for copying and building last.

7.1 COPYING A SERVER DISTRIBUTION DISKETTE

Before you do anything else, follow this procedure to make a backup copy of one of the Server Distribution diskettes.

1. Insert a standard MS-DOS 1.25 hard disk operating system diskette in drive A of a station and boot the station.

3. Insert one of the Server Distribution diskettes in drive B. Remove the system diskette from drive A and insert a blank diskette in its place.

4. Copy the Server Distribution diskette onto the diskette in drive A. If you do not know how to use the DCOPY command, see the Operator's Reference Guide.

5. Remove the copy from drive A and label it SERVER DISTRIBUTION DISKETTE -- BACKUP COPY. Put it in a safe place for permanent storage.

7.2 CREATING THE SERVER SYSTEM DISKETTES

Each server must have its own system diskette. Follow this procedure to create one system diskette for each server in your Network:

1. Label a blank diskette SERVER n SYSTEM DISKETTE, where n is the number of the server for which you are creating the diskette.

2. Use the MS-DOS FORMAT command to format the diskette.

3. Put a hard disk operating system (system tracks) onto the Server n System Diskette. Do this by using the MS-DOS SYSCOPY command to place an MS-DOS 1.25 Hard Disk Operating System on the Server System Diskette.

4. Copy all files from the Server Distribution

backup purposes.

6. If you have more servers, repeat steps 1 through 5 until you have a Server System Diskette for every server in your Network.

7.3 CONFIGURING THE SERVERS

You have already decided how many volumes you want on each hard disk. You made this decision by:

- Knowing the physical configuration of the Network.
- Knowing the capacity of the servers' hard disks (usually ten megabytes).
- Reading the Operator's section of this manual.
- Understanding how each user will use the Network.

Now that system diskettes have been created for each server, you must customize each of the servers. Follow this procedure for each server to be set up. Boot from a server's Server System Diskette. Invoke the NETSETUP program. This program creates a Server Initialization Diskette from the Server System Diskette.

Enter the server number (0 - 9) that appears on the Server System Diskette's label.

- Whether or not you require a large system volume on that server.

  Large system volumes are only required if you have, or anticipate having, over 50 users accessing this server, and/or you have over 10 diskless Network station operating systems to store on this server (for example, different keyboards). Answering YES reduces the size of all other volumes by about 2%.

- The number of volumes into which the server's hard disk is to be divided.

  NETSETUP allows you to configure the server's hard disk into 1, 2, 4, or 8 volumes of equal size. Choose 4 if you have no files larger than 2500Kb. Otherwise, choose the number of volumes that provides sufficient volume size for anticipated large files.

- The size of the server's print spooling queue.

  The number of spool queue entries specifies the maximum number of outstanding (that is, not printed) print requests for all printers attached to the server. Specifying a large queue increases the size of the spool queue file on volume 2 of the server.

SERVER n INITIALIZATION DISKETTE

To initialize and configure your server, boot that server from its SERVER n INITIALIZATION DISKETTE.

CAUTION: initializing and configuring a server destroys all files on that server's hard disk.

The success or failure of the initialization process is reported to you by a voice-like message. If the initialization fails, or no message occurs, it may be informative to connect a screen and keyboard to the server to diagnose the problem. If the initialization succeeds, the initialization diskette is automatically neutralized, and may be recycled.

7.4 CREATING CUSTOM-SIZED VOLUMES

If you determine that you will require "non-standard" volume configurations (such as unevenly-sized volumes, a different number of volumes, or just different volume sizes), the Hard Disk Tool Kit must be used.

NOTE FOR DISKLESS STATIONS: If you have diskless stations, copy all files from the Diskless Station Operating System diskette onto the Server's drive A. Now you can boot from hard disk. Diskless station can now be booted from the Network. When you boot the diskless station, it will have it's

procedure that follows.

Use HDSETUP to set the sizes of the hard disk volumes. When using HDSETUP, the following guidelines must be strictly followed:

- Drive (or volume) A must be the SYSTEM volume. This volume will be used by the Network, and is not addressable by users of the Network. This volume should be set to at least 327Kb for a "small system volume" or at least 527Kb for a "large system volume."

- Drive B must be assigned to the floppy disk drive.

- Drives C to O (for as many volumes as you desire) are the usable volumes for network access. Remember that drive C is also used to store spooled printer output, so you may want to increase its size, or allow fewer users to access it.

- Minimize the AU (allocation unit) size. Because the server is dedicated to Network services and will not run programs, there is no real penalty for having smaller AU's.

Follow these directions to create your custom-sized volumes:

1. Perform NETSETUP as usual, but don't worry about what parameters you specify for System Volume Size and Number of Volumes. NETSETUP produces a file called AUTOEXEC.BAT,

2. After using HDSETUP and NETSETUP, print out AUTOEXEC.BAT, INTERRIM.BAT, and STEP3.BAT.

3. Attach a screen and keyboard to the server to be set up.

4. Enter the commands from your printed output of AUTEXEC.BAT that follow the AUTOSET (you won't require an AUTOSET since a HDSETUP was already performed). When this is completed, the server is initialized and ready to be booted from its hard disk.

System diskettes for disk-based stations. One Network Station Distribution diskette is provided with each station. You may want to use DCOPY to make backup copies of these diskettes.

If a Network station user wants to be logged on to the Network, he or she boots up with the Network Station System diskette. If a user does not want to be logged on to the Network, he or she can boot up with a standard MS-DOS or CP/M-86 system disk.

8.1 PREPARING THE NETWORK STATION SYSTEM DISKETTES

You have been provided with one Station Distribution Diskette for each station. You should make a copy of one of these diskettes to act as a backup. The backup diskette should be stored in a safe place.

NOTE FOR DISKLESS STATIONS: You needn't create Network Station System Diskettes for your diskless stations because they have no floppy disk drives from which to boot. Instead, after installing yourself by using a copy of the server Distribution Diskette in a server's floppy disk drive, copy what is needed from each user's Network Station System Diskette (except the operating system) to a Network volume, which that user can access. Continue to read this section to find out about items to consider.

Follow this procedure for each Network station, to create Network Station System Diskettes:

diskette.

2. Use the MS-DOS FORMAT command to format the new Network System Diskette.

3. Put an operating system onto the Network System Diskette using SYSCOPY from an MS-DOS 1.25 operating system.

4. Use COPY to place the file SYSCOPY.EXE onto the Network System Diskette (SYSCOPY.EXE is available on your operating system disk).

5. Copy all files from the Station Distribution Diskette onto the Network System Diskette (using COPY *.*).

6. The Station Distribution Diskette may now be put away as a backup diskette.

7. Optionally, you may want this station's user to have access to certain of the Network Maintenance Functions. If desired, copy:

```
INSTALL.*
NETSETUP.*
NETTIME.EXE   and/or
NETFIXUP.EXE
```

   onto the Network System Diskette. Other MS-DOS utilities or application programs can be copied onto the Station's Network System Diskette.

9. INSTALLING NEW USERS

This chapter explains how to use the INSTALL program to install new users, and how to change the status of current users. Installing a user includes assigning a user name, password, Network disk drives (volumes), and Network printers.

During the initial installation procedure, all servers should be turned on.

For each user you install, you must know in advance:

- Whether the user should have automatic log-in for a particular station(s), or if he or she should be prompted for a user name and password.
- Whether the user will be assigned to a disk-based or diskless station.
- The physical location of the user's most commonly used station so that appropriate Network printers can be assigned.
- Whether the user's station has a local printer.
- How many, what size, and what kinds of drives (private, shared, or floppy) should be assigned to the user.

as a user of the Network. Regardless of where you intend to store the INSTALL program permanently, the first user name installation (your own user name) must take place from the INSTALL.EXE program on the Server Distribution diskette.

If you are installing yourself as the first user of the system, place your Network Station System diskette in drive A of a Network station and press the Reset button. Place the Server Distribution Diskette in drive B. When the A> prompt appears, type:

```
B:
```

When the B> prompt appears, type:

```
INSTALL
```

If you are installing another user from the INSTALL program that you have placed on a Network drive, you first boot up with your station's Network Station System diskette. Then specify the drive's name. If the program is on Network drive F, for example, type:

```
F:
```

Now type on the command line:

```
INSTALL
```

or if, for example, you are installing a user named PAT, type:

```
INSTALL PAT
```

name. Type the name and press Return.

The Network now queries its user name file for the user name. If the user name is found, the program asks you for the user's password. After entering the password, you can modify the user's Network configuration.

If the user name is not found, certain information is requested for the new user. The bottom of the display that requests the information contains function key assignments. For this display, the assignments are:

QUIT	leaves the INSTALL program without adding the user to the Network.
ACCEPT	tells the Network to store all the information you have entered on the screen and to show the next display. Press this function key when you have finished entering all the requested information.
DELETE	deletes the user from the Network if the user's name is already in the Network user name file.
ANOTHER	ignores the information you have entered and returns you to this display to install another user.
HELP	gives you information about how to enter data on this display. Press this key when you need assistance.

you to give information to the program.

Enter the information requested in the display in the highlighted areas on the screen. To move from one area to the next, press the cursor up- and down-arrow keys. To select between choices on a line, press the space bar or use the cursor keys.

Enter the following information:

Name: Enter a user name. The name you enter can be no longer than 11 characters. Only letters and numbers are acceptable entries. If you enter an illegal character such as a control character or punctuation, a tone sounds.

Password: Enter the user's password. As many as 20 characters are allowed. Only letters and numbers are acceptable entries. If the user is to have no password, press Return. When you enter the password, it is not displayed on the screen. The program asks you to enter the password twice to ensure that it was typed correctly.

Login Type: Choose either AUTOMATIC or MANUAL by pressing the space bar or the left or right cursor arrow keys. If you choose AUTOMATIC, you will enter the number of the user's automatic log-in station on a later display. If you choose AUTOMATIC and if you have

Password if Auto-Login

Choose YES or NO by pressing the space bar, or left- or right-arrow keys. If you choose YES, the user will be prompted for a password at his or her own station even though the rest of the log-in procedure is automatic. If you choose NO, the user will be prompted for a password only if he or she logs in at a station other than his or her own.

Remote Printer: Choose YES or NO. If you choose YES, the Printer Assignment display will appear later.

When you finish entering information on this display, press function key 2 for ACCEPT. If you change your mind about installing the user, press function key 1, QUIT, to leave the INSTALL program. Or you can press function key 5, ANOTHER, to install another user name.

If you chose automatic log-in, a display now appears from which you can choose the station(s) at which the user will be logged in automatically. Move the cursor to a desired station number with the arrow keys and press function key 5, ADD, to select a station. Select as many stations as you want. If you want to remove a station from those selected, move the cursor to the station number and press function key 4, DELETE.

If you select no stations (none are highlighted), then the user will not be logged in automatically.

9.2 ASSIGNING PRINTERS

If you answered YES to the remote printer question, this display appears:

```
Remote Printer Assignment  INSTALL, Version 1.0


Device                          Printer
Assignment    Server            Port
-----------------------------------

TTY           Local             Serial A
CRT             0               Parallel
LPT             1               Serial A
UL1           Local             Serial B


( message line )

---------  -----------  ------------  ---------
¦1 QUIT ¦  ¦2 ACCEPT ¦  ¦5 DEFAULT ¦  ¦7 HELP ¦
---------  -----------  ------------  ---------
```

Use the cursor keys to move to the desired areas on the screen. You can change information only in the SERVER and PRINTER PORT columns.

The function key assignments are shown at the bottom of the screen. The new one is described here.

to the desired line and press the DEFAULT function key.

Enter the following information on the display:

SERVER — Change the entries by typing L for Local Printer, by entering 0 through 9 for the server number, or by pressing the space bar to increment the entry already on the screen.

PRINTER PORT — Each user can be assigned four remote printers or up to three local printers and one remote printer. Port specifications for local printers cannot be changed.

When assigning printers to a user name, consider the user's physical location and the type of printer needed.

To change a server designation, move the cursor to the desired server location and type 0 through 9, or L for Local; or press the space bar to increment the value. To change a remote printer port designation, type A, B, or P for Serial A, Serial B, or Parallel; or press the space bar to increment the value.

When you have entered all your information press function key 2, ACCEPT, to advance to the Drive Mapping display.

9.3 ASSIGNING DISK DRIVES

Two displays let you assign drives (hard disk volumes) to the user. The first display, the Drive Mapping display, shows you the original or a default configuration for the user. You choose drives for the user by making changes to this configuration. The second display, File Server Volumes, shows you all the drives available for a selected server. You can substitute drives from this display for the user's displayed assignments.

DRIVE ASSIGNMENT	SERVER	VOLUME NUMBER	ACCESS STATUS	SPACE AVAILABLE
A	LOCAL	-	---	---
B	LOCAL	-	---	---
C	0	1	shared	340 K
D	0	2	shared	1 K
E	0	3	private	53 K
F	1	1	floppy	---
G	1	2	shared	16 K
H	2	1	private	8 K
I	2	2	shared	225 K
J	2	3	shared	75 K
K	3	1	shared	25 K
L	3	2	private	15 K
M	3	3	shared	300 K
O	3	4	shared	88 K

(message line)

1 QUIT	2 ACCEPT	3 VOL LIST	4 CLEAR	5 DEFAULT	6 UNDO	7 HELP

You can make changes to the SERVER, VOLUME NUMBER, and ACCESS STATUS columns only. Move from line to line and from column to column by using the cursor keys. To change an entry on the cursor line, position the cursor at the desired column and press the space bar to increment the entry, or type a new entry. The new entry can be the server or volume number or the first letter of the desired access status.

VOL LIST — displays additional drive information for the server listed on the cursor line. If the line is unassigned, information is displayed for a specified server.

CLEAR — removes the cursor line and all the lines below it from the drive assignments.

If you want to remove an individual line from the assignments, type a dash on the cursor line instead of using CLEAR.

DEFAULT — selects the optional default drive assignment. To specify a new default drive assignment, move the cursor to the desired line. Press the DEFAULT function key to remove the current default setting. Press the DEFAULT key again to assign default status to the drive on the cursor line. If you do not select a default, no drive change will take place when the user logs in.

UNDO — returns the drives to the assignments they had when the display first appeared.

Each column contains the following information for each drive assigned to the user:

drives. Drive assignments need not be contiguous; there can be missing letters in the sequence of drive assignments.

If the user has a disk-based station, drives A and B are usually local drives. If the user is assigned to a diskless station, drives A and B can be Network drives. For stations with hard disks, there may be more local drives.

This column is informational. You cannot change the entries.

SERVER — This is the number of the server associated with the drive name. In the sample, server 2 is assigned drive F. You can change the server assignment by pressing the space bar, or by typing another server number. Type L to set the drive to LOCAL, or type - to "unassign" it.

SERVER VOLUMES — The Server Volume is the assigned volume on the server's hard disk. Change the volume assignment by pressing the space bar or by typing a new volume number. Volume 1 is always the server's floppy disk. It should be used for file transfer if other floppy disks are unavailable. Volume 2 is used for printer spool queuing. This drive can be

A third way to change the volume number is to press function key 3, VOL LIST, to see the File Server Volume display.

```
File Server Volumes      INSTALL, Version 1.0

                  SERVER        SPACE
                  VOLUMES     AVAILABLE
                  -------     ---------
Server #5           1           49 K
                    2          250 K
                    3           20 K
                    4           75 K
                    5          110 K
                    6          200 K
                    7            7 K
                    8           10 K

------------------   --------------------------
¦1 QUIT ¦2 ACCEPT ¦   ¦5 ANOTHER ¦6 UNDO ¦7 HELP¦
------------------   --------------------------
```

determine the volumes' original sizes by the number of volumes on the server. Move from line to line on this display by using the cursor keys. The highlighted line indicates the currently selected volume.

If you see nothing on the File Server Volumes display that you want to add to the user's drive configuration, you can choose function key 5, ANOTHER. This message appears on the screen:

Which file server would you like to see? (0 - 9)

Type the number of the server for which you want to see drive information. Function keys 6, UNDO, and 2, ACCEPT, return you to the Drive Mapping menu. UNDO leaves the drive assignment unchanged. ACCEPT changes the drive assignment to the assignment indicated by the current cursor position.

private or shared. If the directory is private, only the user can display it and only the user can access the files. Press the space bar to alternate between SHARED and PRIVATE. For the floppy disk, the only designation allowed is FLOPPY.

SPACE AVAILABLE This column tells how much space is currently available on the drive. The entries in this column change automatically as you specify different servers and volumes.

Press the ACCEPT function key to create the drive configuration for the user. The Initial Data Assignment display appears on the screen again. You can either add another user to the Network or press function key 1, QUIT, to leave the INSTALL program.

You can also take this opportunity to place the INSTALL program files in another location and then continue the user installation process.

INSTALL.DAT to a permanent location. Here are places to store the program:

- Your own or a designated person's private Network drive, so that only the designated person has access to it.

- A Server Network Station System diskette, to be used only when a privileged person has to change a user's status, or add a user. If someone else uses your system diskette, he or she can install users and change users' configurations.

- A shared drive, thus making user installation and password changing available to all users of that drive (not recommended).

To copy the program from your own system diskette to a Network drive to which you have access, follow this procedure:

1. Place your Server Distribution Diskette in drive A of a station and boot up.

2. Type this MS-DOS command to copy the program to a Network drive:

```
COPY A:INSTALL.* d:
```

where:

d is the Network drive designation to which the program is to be copied.

DEL A:INSTALL.*

9.5 DELETING A USER FROM THE NETWORK

Before deleting a user from the Network, all of the user's private files must be deleted. If possible, the user should do this. If this is not possible, you can (if you have maintained a list of user names and passwords) delete the files yourself.

To delete a user, invoke the INSTALL program and enter the user's user name and password. The INSTALL program adds a function key assignment, 3 DELETE. Press function key 3 to delete the user from the Network.

This message appears on the screen:

> **Are you sure you want this user to be deleted from the Network?**

Function keys 1 and 2 are now assigned to NO and YES. Press function key 2 to verify that you want the user deleted.

User Name display appears. Enter the user's user name. If the user's user name file is found, this message appears:

Configuration file has been found for this user.

If the user has a password, you are asked to enter the password before you are allowed to change the configuration.

Now you can change any information on any of the displays using the method you used when you first entered it. If you do not want to change information on a display, press the ACCEPT function key to go on to the next display. If you do want to change information, change it and then press ACCEPT.

maintenance utilities. NETTIME sets the time every time the system is rebooted. NETFIXUP performs various routine (and a few not-so-routine) maintenance tasks.

10.1 MAINTAINING THE NETWORK TIME

Any time the Network is rebooted, the Network time and date must be reset. Just as with the standard MS-DOS, the Network uses its time utility to note the times files have been accessed.

NETTIME sets the time and date from the time and date at your local station. It executes automatically when the first user logs in after the system has been booted.

If you want to change the Network time at some other time, first use the MS-DOS TIME and DATE utilities to set the local time and date at your station, then use NETTIME.

To set the Network time and date, type:

```
NETTIME
```

The Network display looks like this example:

```
Network date and time is Tue 1-Jan-1983,
   01:00:00.0

Local date and time is Tue 1-May-1983,
   11:42:30.08
```

If you type 1, the NETTIME program replaces the current Network time and date with your local station's time and date. If you do not want to change the Network time and date, press Return.

10.2 NETWORK MAINTENANCE FUNCTIONS

The NETFIXUP utility rectifies various situations that might occur in the life of your Network. It also performs routine functions such as changing passwords and deleting users. A user might forget his or her password, hardware errors could destroy certain files on the server, or performance tuning may be desired. NETFIXUP can take care of all these problems.

CAUTION: NETFIXUP is very powerful and should not be made available to the general Network user; it has the potential to totally destroy Network and user data.

NETFIXUP displays a menu with choices of 9 functions:

1 - regenerate the AUTOLOGIN file (USERS.ID)
2 - regenerate the NETUSERS file (CURRENT.ID)
3 - display a user's configuration
4 - delete a user
5 - change a user's password
6 - log out a station and release its reservations
7 - access server system files
8 - edit Network parameters
9 - quit this program

The following describes the available choices in more detail:

Functions 1 and 2. Regenerating the AUTOLOGIN and NETUSERS files is useful if those files become trashed (either through a hardware failure or through some system software error).

The AUTOLOGIN file's name is USERS.ID and is on each of the servers' system volumes. AUTOLOGIN contains 64 records, one for each possible Network station or server. The contents of the record is the user name, which is to automatically log into that station, and a boolean value indicating if auto-login is used for that station.

The NETUSERS file's name is CURRENT.ID and is also on the servers' system volumes. NETUSERS contains the user names of all users who are currently logged into the Network. The contents of the file is 64 records, each possibly containing a user name.

To regenerate each of these files, NETFIXUP reads the old file, deletes any nonsensical records, and writes the data out to a new file.

Function 3. To display a user's configuration, enter the user's user name. A full-screen display appears with that user's disk drive and printer assignments.

the Network.

Function 5. NETFIXUP also can be used to change a password. You cannot see the user's old password, but you can change the password. In this way, the user will know if anyone has been "playing" with his or her files. Specify the user name and the new password after choosing choice 5.

Function 6. Logging out a station is valuable when files have been reserved but, through some software or hardware error, cannot be released. After requesting choice 6, enter the station's number. The station's current user is logged out, freeing all resources. Note that logging out a user is very powerful, and you should not log out a user if he or she is currently doing work on the Network; it's a good way to make an enemy.

Function 7. This choice allows you to map one of your local drive designators to a server's system volume. The information you need to specify is: (1) the drive designator to be used in the mapping, and (2) the server to be mapped. This mapping stays in effect until you issue a LOGOUT or LOGIN command.

By mapping your drive N, for example, to a server's system volume, you can upgrade its FSNET.EXE software (which will be put into effect when that server is next booted), or update the LOGIN message file. To do either of these, copy the new file to N:FSNET.EXE or N:NETWORK.MSG after exiting from NETFIXUP.

to access the file N:NETWORK.MSG (assuming your drive N was mapped using choice 7). This file will be displayed upon each user's login.

You can have more than one server system volume mapping in effect at any time (that is, you can repeat function 7 for various drive designators and servers).

Function 8. This function displays a second menu of particular "Network software" parameters that you can modify for your particular station. These parameters remain in effect until you reload the Network software at your station, or you modify them again using NETFIXUP. Note that these parameters only affect your station (that is, the station at which you are running NETFIXUP). The Network parameters are:

1 - ENABLE NETWORK FILE I/O ERROR DIAGNOSTICS
This parameter is initially OFF. If set to ON when file I/O errors occur, a message is displayed on your screen providing you with information about the error. For many Network file I/O errors, MS-DOS substitutes standard error messages. For example, FILE RESERVED/ACCESS DENIED may manifest itself as a FILE READ/ONLY error. By setting this switch to ON, specific Network error messages may be obtained.

errors occur, additional diagnostic information is displayed on your station's screen.

3 - ENABLE WILDCARD DELETES This parameter is initially OFF. If set to ON, wildcard deletions on shared volumes may occur. In the normal case, these deletions are denied with a "file not found" error message and DELQ must be used.

4 - NUMBER OF RETRIES BEFORE SERVER 'MISSING' In error recovery,the system retries communication with the server. When the retry limit is reached, the server 'missing' error occurs and the user is asked to Retry, Abort, or Detach. A larger retry count stops the error message from appearing, but may hide real Network problems and make the user think that Network performance is lacking.

5 - WAIT BEFORE 'NO REPLY' ERROR When a request is accepted by the server, the station waits a specified length of time for the request to be filled. This kind of error occurs when the server has acknowledged receipt of the request, but takes too long to respond with the information (such as reading the disk record).

6 - WAIT BEFORE TRANSMIT BOARD FAILURE This timeout request is the amount of time the software waits for the Omninet board to accept a transmission before assuming that the board is not operational.

with another station's request), this is the amount of time the software waits before retrying the request. By waiting a longer time, it is more likely that the old request will be completed (then again, a new one may have been picked up in the meantime), but performance suffers.

8 - WAIT FOR RECEIVE PREPARATION This timeout is the length of time the software waits for the Omninet board to accept a receive requests before assuming that the board is not functioning.

9 - RETURN TO THE MAIN MENU This choice terminates Network parameter setting.

Select your choice, followed by Return.

Main menu choice 9 causes you to exit from the NETFIXUP utility.

THE INTERFACE LIBRARY

This section describes the routines in the Server Network interface library, and tells you how to use these routines in high-level programming languages.

Server Network has a high-level interface that enables you to develop and run programs using files or other resources from several user areas. The interface supports these programming languages:

- MS-BASIC Interpreter
- GW BASIC Interpreter
- MS-BASIC Compiler
- MS-Pascal
- Intel PL/M Compiler
- Microsoft Macro Assembler

The Network functions can:

- Reserve files for private or for shared access.
- Lock and unlock data records in files open for shared access.
- Release files from reserved status.
- Determine and set the write-protection status of files.

- Determine the types of Network errors, when they occur.

Chapter 12 describes the Network functions that the high-level interface routines can perform. Chapter 13 gives the codes you use in BASIC to request the Network routines, and the error codes returned by the routines in both BASIC and MS-Pascal. Chapter 14 describes how to access and use the Server Network interface library from high-level programming languages.

The appendixes give the calling sequences for programming languages. Appendix A gives examples of routines for BASIC, and a sample program using Network functions. Appendix B gives the EXTERNAL declarations that you use to include Network functions in MS-Pascal. Appendix C deals with Assembly language. Appendix D deals with PL/M.

These are the fifteen Network functions:

1. RESERVE FILE
2. RESERVE FOR SHARED ACCESS
3. LOCK RECORD
4. UNLOCK RECORD
5. RELEASE FILE
6. RELEASE ALL
7. SET PROTECT
8. GET PROTECT
9. GET SERVERSTAT
10. GET SERVERID
11. TEST STATION
12. GET FREE SPACE
13. GET USERNAME

11.1 SHARED-FILE MANAGEMENT CONSIDERATIONS

Shared Network files can be accessed by several users or user programs at one time. This situation is much like a multi-user environment in a time-sharing system.

Although shared files can be accessed by many users, the Server Network normally prevents two users or programs from writing to the same file at the same time, or from modifying a file that is being read by someone else. This is called automatic write-locking. When you open a file, however, MS-DOS cannot tell whether you will modify or just read the file. For this reason, automatic locking never occurs until you write to the file.

Because the Network keeps track of all opened and closed files, you should close files as soon as you finish with them. If you leave a file open until you finish your session, the MS-DOS COMMAND program releases it from the open file list at that time, but no other user can reserve or update the file while your program is executing. This restriction applies to an output file too; close it immediately to make it available to the Network as soon as possible.

The previous discussion implies that:

- You should not create files with names that might be duplicated by another user. Avoid names like TEMP, STUFF, and JUNK on shared drives.

the RESERVE FILE function described in the next chapter.) You can also reserve the file by writing to it immediately (write-locking).

- If two users run a program that creates the same file on a shared drive, the second user receives a file creation error message, and his or her file is not created.

- If you access a file that someone else has opened, your attempt to write out processed data or to make the program accept file input may fail.

If you want to create a temporary file or any other output file, check for an error to see if someone else has the file open. If your attempt is not successful, change the filename in some unique way and try again. A simple technique is to append the station number to the filename with the GET MYID function. This technique lets all Network users run the program on a shared disk volume without colliding because the station ID is unique on the Network.

When you cannot change the output filename, request another name from the user or try to create the file at the beginning of the operation. If you are successful, you will have exclusive use of the file until you are finished. This practice prevents others from using the program on the same Network drive, but you do not lose the user's data by discovering that you cannot create or write to the file at the end of the session.

the input file. In this case, your program is denied access, just as if the file were not found. You can use the Search First function (MS-DOS function 17H) to see if the file is actually there but has been updated or reserved by someone else. You can also use the GET NETWORK ERROR function to see if there was a Network operation error (such as "file reserved").

11.2 PROTECTING SHARED FILES

You can protect shared files to prevent some of the problems inherent in multi-user systems. The Server Network has several commands and functions that reserve files. Reasons for reserving files are described here and in Chapter 12. The use of the reserve functions and commands is described in Chapter 14, the appendixes, and in the Operator's portion of this guide.

Here are four ways to protect shared files:

1. You can set a Network file to READ (READ-Only) status with the Network PROTECT command or the Network SET PROTECT function. No one, including you, can update a READ file; attempts to do so result in an "access denied" error.

   Generally, it's a good idea to give READ status to all program files -- especially standard programs that can produce a

by a customized version.

2. You can lock an existing shared file that you have opened by writing to it immediately. Until you write to the file, other users can also open it. Once you write to a file, that file is locked so that no one else can open or modify it until you close the file. Attempts to open a write-locked file generate an MS-DOS "file not found" error. MS-DOS does not allow concurrent access to a WRITE file that is being created or updated.

   This mechanism can be bypassed by application programs capable of managing their own concurrent access to a file or files.

3. You can reserve one or more Network files with the Network RESERVE FILE function (or RESERVE command). This function ensures that no other Network station can open or modify the files until they are released by the user with the RELEASE ALL function or RELEASE command. Attempts to open a file reserved by another user result in a "file not found" error.

4. You can reserve files with the RESERVE FOR SHARED ACCESS Network function. All programs that use this function can open and update a file at the same time, but no one else can open the file. These programs should also use the LOCK RECORD Network function to reserve private access to the specific record(s) to be updated. Release the records with UNLOCK RECORD. Release the file with RELEASE ALL.

can occur. A deadlock develops like this. Suppose program 1 tries to reserve files A, B, and C, in that order. At the same time program 2 tries to reserve files C, B, and A, in that order. Program 1 reserves file A, and program 2 reserves file C. Next, program 1 reserves file B. Now program 2 tries to reserve file B, but it is already reserved. Program 2 continues to attempt to reserve file B. Meanwhile, program 1 attempts to reserve file C, which has already been reserved by program 2. Neither file will ever become free to be reserved by the other program because each program is "hung up" waiting for the other's reserved file to become free. (You can free yourself from a deadlock like this with the RELEASE ALL Network function, which releases all your reservations.)

12. SERVER INTERFACE FUNCTIONS

This chapter lists and describes the Network functions performed by the interface routines. Sections 12.1 through 12.3 list the functions that you use to reserve and release files, or to lock and unlock records. Sections 12.4 through 12.6 list the functions that get or set status information about the station or the server.

To use the Network functions, you must follow the format for the language you are using and include parameters as shown in Appendixes A and B.

NOTE: In future versions of the Network, some of these functions may not be supported because of changes in the Network architecture.

12.1 RESERVING FILES FOR PRIVATE ACCESS

The RESERVE FILE function gives you exclusive access to a file. Make a RESERVE FILE request at the beginning of program operation if your program will write to that file. If you do not reserve the file, your write can fail if another user is accessing the file.

RESERVE FILE: Reserves a file for access only by the station making the request.

Filenames with wild-card characters are not allowed. You can reserve only one file with each RESERVE FILE call; to reserve more than one file, make multiple calls. If a reservation request fails, perform a RELEASE ALL and start over.

Any number of users can open and modify a file if they all reserve it for shared access first and then lock and unlock the data records they want. Your reserve request succeeds only if the file exists and has not been opened for private access by any other user.

When you access a file you have reserved for shared access, first lock the data record you want to access, then perform the read or write, and finally unlock the record when the operation is completed.

You lock records by locking a unique key to the record (see LOCK RECORD in Section 12.2.2).

12.2.1 RESERVING FILES FOR SHARED ACCESS

RESERVE FOR SHARED ACCESS: Reserves a file for concurrent access using record locking. Use this function before opening the file.

Filenames with wild-card characters are not allowed. You can reserve only one file with each RESERVE call; to reserve more than one file, make multiple reserve requests.

As with RESERVE FILE, if any reserve request fails, you should release each reserved file before you continue or before you attempt to reserve the file again (to avoid deadlocks). When access is completed, release all reserved files with RELEASE ALL or RELEASE FILE.

You must identify the record to be locked by a unique key of up to 64 characters or numbers. This key must be unique to any file being accessed on the drive. One way to make sure the key is unique is to append the file's name to it.

All programs accessing the file must first attempt to lock the record you want with the same key. You can use the physical address or a logical key for locking a record. The Network provides no protection at the time of access without this locking system (the same set of unique keys to the file). If a LOCK attempt fails, you must unlock all locked records (to avoid deadlocks).

Do not extend a file opened for shared access. Restrict your access to records that already exist. If you must extend the file, reserve it first with the RESERVE FILE function.

LOCK RECORD: Locks a record for access only by the station making the request.

This function allows concurrent access of random access files by reserving unique keys (strings) to those records before accessing them.

Only one string should be reserved at a time; all reserved strings should be released if access to a string is denied (to prevent deadlocking). When the need for a reservation is completed, you must explicitly release the reservation with UNLOCK RECORD or RELEASE ALL.

Each UNLOCK call can unlock only one record; to release all reservations and locked records, use RELEASE ALL.

12.3 RELEASING RESERVED FILES

Use either RELEASE ALL or RELEASE FILE to release all your reserved files at the end of program operation. Failure to release reservations means that other users can be denied access to those files.

RELEASE FILE: Releases a file that you have reserved.

Use this function (or RELEASE ALL) to give access to a reserved file to other users.

RELEASE ALL: Releases all the files and records you have reserved on any Network server. RELEASE ALL also releases any files and records that were reserved with the RESERVE utility.

Always RELEASE ALL when operations on reserved files are completed.

bit for each attribute, as shown in Table 12-1. A file can have several attributes set -- such as READ-Only, System status, Archived, etc.

Use GET PROTECT to return the status of the attribute byte (see Appendix A). With SET PROTECT you can change certain attributes of a file. For example, you can give READ status to a file and prevent changes to the file. You can also give a file ARCHIVE status by turning on bit 5; do this after you have backed up the file.

Table 12-1: Bits in the Attribute Byte

BIT	ATTRIBUTE
0	OFF - Read/WRITE
	ON - READ-Only
1	Hidden File
2	System File
3	Volume ID
4	Directory
5	Archive

SET PROTECT: Changes the write-protection status of the specified file (on a non-local drive).

You can set files to READ-Only, Read/WRITE (default), Hidden, System, or Archive status by turning on the corresponding bits (see Table 12-1 and Appendixes A and B). You cannot set the Volume ID or Directory attributes with this function.

Archive status is used by a backup program to indicate that a file has been backed up.

GET PROTECT: Returns the write-protection status attributes of the specified file. (See Appendixes A and B.)

12.5 GETTING SERVER STATUS

This section describes the functions that give you information about the servers on the Network. These functions can:

- Give you server status data such as server version and name, number of drives available on the server, and total number of files open on the Network.
- Give you the server number for any remote drive.
- Test to see if a station is on.
- Give you the amount of free space on the non-local drive you name.

GET SERVERSTAT: Copies the server's status information record into your program variables. (See Appendixes A and B for specific data returned.)

server).

You can use the server_id as a parameter for other station interface functions.

TEST STATION: Performs a test to see if the specified station is powered on.

This function does not indicate whether the station is using the Network or is in local mode.

GET FREE SPACE: Returns the amount of free space on the specified logical drive (non-local drives only).

This function copies the drive information into your program variables.

12.6 GETTING STATION STATUS

GET USERNAME and GET MYID give you information about your station.

You should use GET NETWORK ERROR when an error occurs; this function returns the code that identifies the error.

GET USERNAME: Returns your user name as an 11-byte string left-justified and padded with trailing spaces.

GET NETWORK ERROR: Returns an error code for an MS-DOS file function that did not succeed, such as an OPEN, write (PUT in BASIC), or read (GET in BASIC).

When you make this request after getting the error, you get back the code for the system function just performed. For example, if a file is reserved by another user and you attempt to open it, you receive a "File not found" error message. If you then call GET NETWORK ERROR, you get the correct error code (224 File Busy).

13. FUNCTION-REQUEST AND ERROR CODES

This short chapter gives you:

- The codes to use in BASIC when you request Network functions.
- The error codes returned by Network functions.

Table 13-1 gives the codes that you use in any BASIC program to specify Network functions. Substitute this code for E.RROR.CODE% (the last variable in the CALL statement), as shown in Appendix A. This variable tells the interface library what function you want. To include Server Network functions in MS-Pascal programs, make them EXTERNAL declarations (see Section 14.3 and Appendix B).

Table 13-1: Network Function-Request Codes for BASIC

CODE	FUNCTION REQUESTED
19	RESERVE FILE
30	RESERVE FOR SHARED ACCESS
31	LOCK RECORD
32	UNLOCK RECORD
20	RELEASE FILE
23	RELEASE ALL
28	SET PROTECT
29	GET PROTECT
17	GET SERVERSTAT
27	GET SERVERID
16	TEST STATION

33 GET NETWORK ERROR

Table 13-2 lists and defines the error codes returned by the Server Network functions. These codes are returned for all language interfaces.

Table 13-2: Error Codes Returned

CODE	MEANING
0	Successful call to the Network server. All variables that were to receive information have received information. (If there is an error, however, the variables will not contain Network server information.)
53	File not found (MS-BASIC). The file does not exist on the Network.
64	Bad filename (MS-BASIC). The filename does not conform to MS-DOS conventions.
128	Local drive. The drive number is not on the server; it is a local drive (on the station).
129	Network error. You have loaded a version of the operating system that does not support the Server Network. Boot your system with a Network station diskette (MS-DOS 1.25 or higher, with Network software).

130 **Station not active.** The requested station did not respond to the test.

131 **Access denied.** The Network server could not access the requested information.

132 **Invalid attribute.** The passed file attribute did not correspond to attribute specifications, or tried to set an attribute that cannot be set by the user.

133 **File not reserved.** The requested file or files are not reserved.

134 **Key not reserved.** The key string does not match any keys currently locked.

135 **Invalid name string length.** The string passed must be at least 11 bytes long. (Not used in Assembly language or PL/M.)

136 **Invalid server name string length.** The string passed must be at least 11 bytes long. (Not used in Assembly language or PL/M.)

137 **Invalid filename string length.** The string passed must be at least 12 characters long. (Not used in Assembly language or PL/M.)

138 **Invalid drive specification.** The drive number is not in the range 0 (default) to 15 (drive O:).

(Not used in Assembly language or PL/M.)

140 Server not active. The requested server was not present on the Network.

141 Drive not mapped. The passed drive specification is not on the Server Network or local (unassigned).

142 Invalid station ID number. The station number must be in the range 10 - 63.

143 No reservations. RELEASE ALL returns this error if the server (0 - 9) is active and returned an error.

144 File reserved. The file or record you are trying to access, reserve, or lock is reserved.

145 Physical transmission error. The Server communication has failed due to physical or software connection problems.

146 Key Reserved. This key has already been reserved by another user.

224 File busy. The file is already reserved, or open.

225 Bad directory.

226 Bad file closure.

227 File not open.

229	Bad request. The function-request code is out of range.
230	Active full. The RESERVE file/key list is full.
231	File not found. The file does not exist.
232	Bad spool. The printer spool queue is full.
253	Transporter received message time out.
254	Transporter retried too many times.
255	Transporter transmitted message time out.

NOTES: Codes 224-255 are returned only by the GET NETWORK ERROR function.

Codes 224-232 are logical errors.

Codes 253-255 are physical errors.

You can access the Server Network high-level drivers from MS-BASIC and GW-BASIC Interpreters, MS-BASIC Compiler, and MS-Pascal.

- To use GW-BASIC or MS-BASIC Interpreter, you must load the module SERVIBAS.COM before you access the server (see Section 14.1).

- After development of an MS-BASIC Compiler, Assembly language, PL/M, or MS-Pascal program, you must link one of these modules with your program:

  -- SERVCBAS.LIB for MS-BASIC Compiler

  -- SERVASM.LIB for Assembly language

  -- SERVPAS.LIB for MS-Pascal

  -- SERVPLM.LIB for Intel PL/M

SERVIBAS

This command loads the module SERVIBAS.COM into memory. Your screen displays the message:

File Server/BASIC interface ready - Version x.x

You must load this interface before you try to access the Server Network from a BASIC program. You can load the interface only once, however, during an operating session. If you attempt to load SERVIBAS.COM again, the bell sounds and this error message is displayed:

Error. File Server/BASIC Interface Previously Installed.

After the Server/BASIC interface is loaded, BASIC programs can access the Server Network with the CALL statement. Before you can use CALL, however, you must define the segment and offset of the server interface entry point with the DEF SEG statement. The entry address of the Server/BASIC interface is found in interrupt vector table (IVT) entry 219 (the 4 bytes of memory starting at address 0:876). By PEEKing these 4 bytes, you can

```
 5   DEF FNPKW!(ADDR) = PEEK(ADDR) + 256 * PEEK(ADDR + 1)    'peek word
10   SERV.ADDR = 219 * 4                'interrupt 219 * 4 = offset into ivt
15   DEF SEG = 0                        'segment of interrupt vector table
20   SERVER = FNPKW!(SERVR.ADDR)        'get offset of server call
25   SERV.SEG = FNPKW!(SERV.ADDR + 2) 'get segment of server call
30   IF SERV.SEG THEN GOTO 50           'check for server present
35   PRINT "Error: Server not present !"
40   STOP
50   REM  -- MAIN PROGRAM
```

After you execute these statements, call the server interface from BASIC with a statement in this form:

```
DEF SEG = SERV.SEG
CALL SERVER (parameter1, ..., parametern,
             E.RROR.CODE%, FUNCTION.CODE%)
```

where: parameter1, ..., parametern are the parameters for this function (see Appendix A).

E.RROR.CODE% is a variable to receive any error code returned (see Table 13-2).

FUNCTION.CODE% is the number of the Network function you want (see Table 13-1).

- You must define numeric variables as integers, either by appending a percent sign to their names, or by using the DEFINT statement.

- You must pass parameters as initialized variables.

14.2 MS-BASIC COMPILER

The SERVCBAS.LIB library contains:

- A dispatcher for the high-level routines in BASIC to the high-level Server Network interface.

- An interface from the high-level routines to the low-level BIOS routines.

Use the following statement format to CALL a Server Network function in compiled BASIC:

```
CALL SERVER (parameter1, ..., parametern,
             E.RROR.CODE%, FUNCTION.CODE%)
```

where: SERVER is the dispatcher in the SERVCBAS library that interfaces to Server high-level routines.

parameter1, ..., parametern are the parameters for the high-level routine being accessed (see Appendix A).

E.RROR.CODE% is a variable to receive any error code that is returned (see Table 13-2).

Table 13-1).

After writing your program, use the following procedure to link your routine (represented as <filename>) with the library of Network routines:

```
LINK <filename>
Run File [A:filename]: (user discretion)
List File [nul.MAP]: (user discretion)
Libraries [.LIB]: SERVCBAS.LIB
```

14.3 MS-PASCAL

You must declare all Server Network functions as EXTERNAL (see Appendix B). Build your program and compile it as usual, using Network functions. When you link your program, specify an additional library to be searched -- SERVPAS. SERVPAS.LIB contains the high-level Network routines and an interface from these routines to the BIOS routines. The filename used in MS-Pascal has the normal MS-DOS filename format; it is not an FCB format.

14.4 INTEL PL/M

You must declare all Server Network functions as EXTERNAL (see Appendix C). Build your program and compile it as usual, including the Network PL/M procedures. At link time specify, in addition to the libraries normally needed, SERVPLM. SERVPLM.LIB contains the high-level Network procedures and an interface from these libraries to the BIOS functions. The FCB pointer described is not parsed nor checked for

PL/M. Therefore, those error messages in Table 13-2 pertaining to string length errors will not be reported by the library procedures.

14.5 ASSEMBLY LANGUAGE

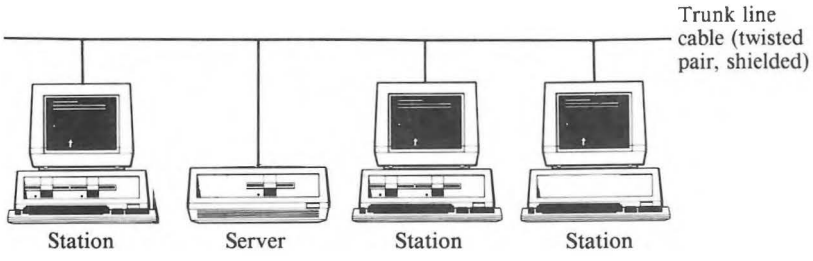
The Assembly language library is identical to the PL/M library in the type of data structures and calling conventions used. The cautions regarding the FCB and strings from PL/M also apply to these library error checking mechanisms. You must declare all Server Network procedures as EXTERNAL:FAR (see Appendix D). Build your program and assemble it as usual, including the Network Assembler procedures. At link time, in addition to the libraries normally used, specify SERVASM. SERVASM.LIB contains the high-level Network procedures and an interface from these libraries to the BIOS function calls.

15. SERVER NETWORK HARDWARE CONSIDERATIONS

This section discusses the hardware considerations that affect the software installation of the Network, positioning of hardware components, and the addition of new servers and stations to the Network.

15.1 OVERVIEW

The Server Network is a local network on the user's premises, consisting of two to sixty-four computers. These are connected together along a simple cable called a trunk line, or "network trunk." (See Figure 15-1.) From one to ten of these computers may be 256K computers, equipped with hard disk drives, which act as Network servers. From one to 54 computers, known as Network stations, can be entirely diskless, or equipped with their own floppy or floppy and hard disk drives. Any station has access to all the peripherals controlled by all the servers, and to all information in the Network (except for files designated private, reserved files, and so on). Thus the stations do not require disks or printers of their own.



Every Server Network must have at least one server. Every server in the Network must have an internal hard disk.

Each server can be equipped with up to three printers. The server provides automatic spooling for its printers, handling station requests on a first-in, first-out basis.

Keyboards and screens are not required for the server computers, unless the user desires them. The servers are resources for the station computers. The number of servers needed depends on the user's mass storage (hard disk) and printing requirements.

Any networked computer may join the Network or leave it at any time. (To be "networked," the computer's Network interface software must be loaded with the operating system.)

between user data packets.

15.2 COMPONENTS

Very few components are needed to convert a computer for Server Network use. The Omninet board, accessory interface plate, and connecting cable are packaged as a conversion kit. The junction box must be ordered separately. Each networked computer requires:

1. A Server Network junction box to be physically installed on the trunk cable, within 5 feet of the computer's location. Each networked computer must have a junction box.

2. A Server Network electronics board (Omninet board), which plugs into any expansion slot inside the computer.

3. A small accessory plate, which holds a jack for accepting the plug from the junction box cable. Inside the computer, a pair of wires from this jack runs to the Omninet board. This accessory plate replaces a blank plate that covers an access hole on the rear of the computer.

Also available is a station software diskette for disk-equipped computers being upgraded for service as stations, and an optional auto-boot ROM for such computers. (These normally boot up and log

A computer with its Omninet board installed, whether it is a server or a station, is called a Network node. Each node is connected to the Network trunk via its own junction box.

15.3 THE TRUNK LINE CABLE

The trunk line can be up to 1500 feet (about 450 meters) long. Use Victor Part No. 104386-01 for the trunk cable, available from Field Engineering. This is a braid-shielded twisted pair. This cable is used because of its low loss and characteristic impedance. It will not radiate the high-frequency signal that causes interference.

Do not attempt to exceed the 1500-foot length or substitute a different cable type without consulting your distributor. This total length should accommodate up to 64 computers without degradation of the signal due to loading losses.

Field Engineering will install or support the installation of the Network trunk cable. Either they or the dealer's hardware technicians can install the Omninet boards in the computers.

The trunk line cable must be terminated at its extreme ends. Each Server Network junction box contains a terminating resistor installed at the factory. (Termination is described in the Server Network hardware installation instructions.) There are two ways to install a trunk cable.

each end. The terminating resistors should not be removed from these junction boxes. The end junction boxes may or may not be connected to a computer. Junction boxes in the middle of the trunk line must have their terminating resistors removed.

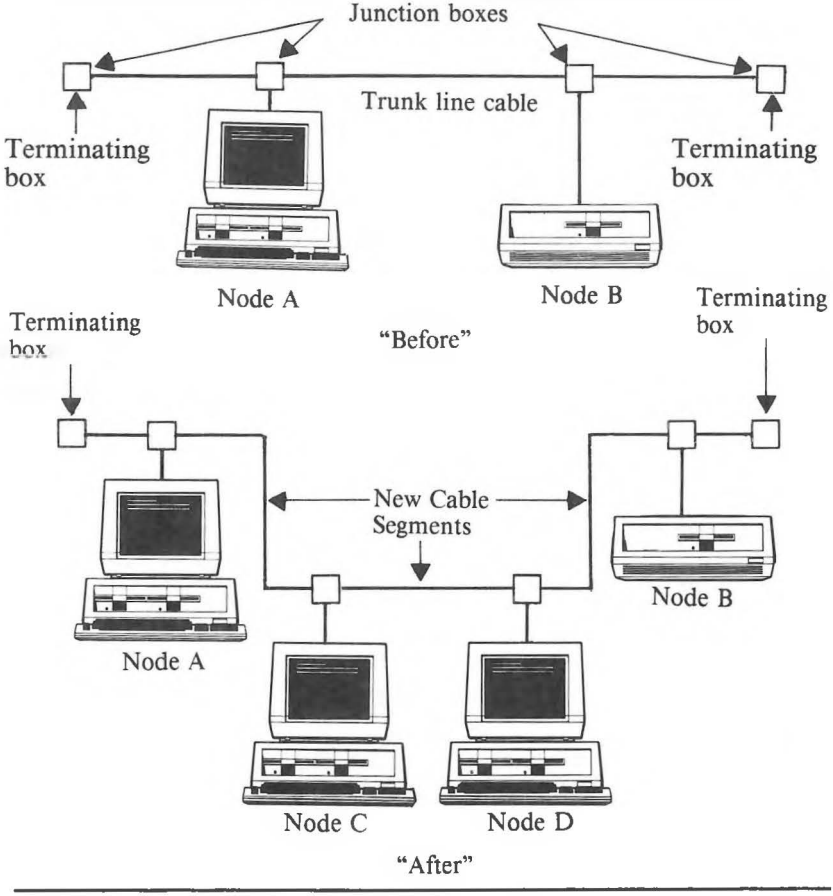
2. If the Network starts small and is enlarged "computer by computer," with new trunk cable segments added to one end or the other, the "first" and "last" junction boxes (corresponding to the computers at the ends of the trunk cable) have their terminating resistors connected, as detailed in the Server Network hardware installation instructions. When new computers and cable segments are added at one end of the cable, the terminating resistor in the original "end" junction box must be disconnected. (The original terminations are not changed when new computers, or additional cable segments, are added in the middle of the trunk rather than at the ends.)

The trunk cable runs from computer to computer in a direct straight-line series (even though the cable does turn corners, go up and down walls, and so on). You cannot configure the Network in a "T" or "Star" configuration.

You can extend the trunk line anywhere along its length, not only from its ends. For example, suppose you have two nodes, A and B, at adjacent locations, and you wish to add a third computer, C, to be physically located between them. The

computers, causing the trunk line to run A-C-D-B. You might add any number of nodes, as long as the total number of nodes on the Network does not exceed 64. Again, the total trunk cable length must not exceed 1500 feet. See Figure 15-2.

Figure 15-2: Adding Nodes (Computers) to the Trunk Line Along the Original Length ength

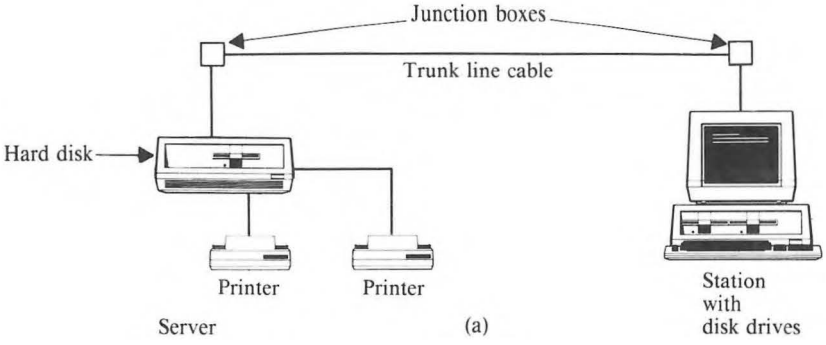


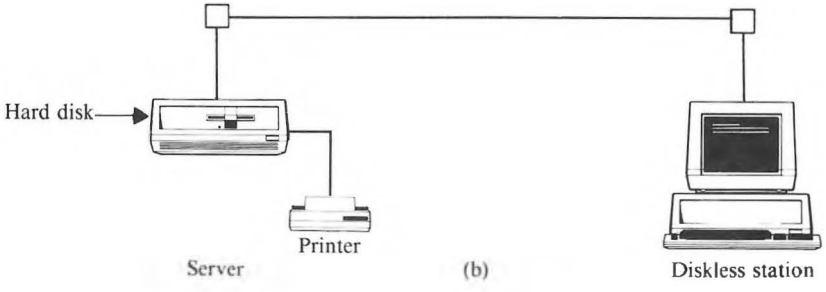
boxes can be preinstalled at those locations or spliced into the loops as computers are delivered. ("Splicing" is the cutting of the cable and attaching the cut ends of the cable's wires and shields to terminal screws inside the junction box.) The lengths within the loops must be reckoned in the total length of the cable, not to exceed 1500 feet.

As your Network grows, tracking the wiring paths to an increasing number of stations and servers becomes an increasingly complex problem. This can cause irritation and lost time if you ever need to isolate a fault in the Network. To prepare for this, label all junction boxes with a number and all cables with their destination junction boxes' numbers.

15.4 CONFIGURATIONS

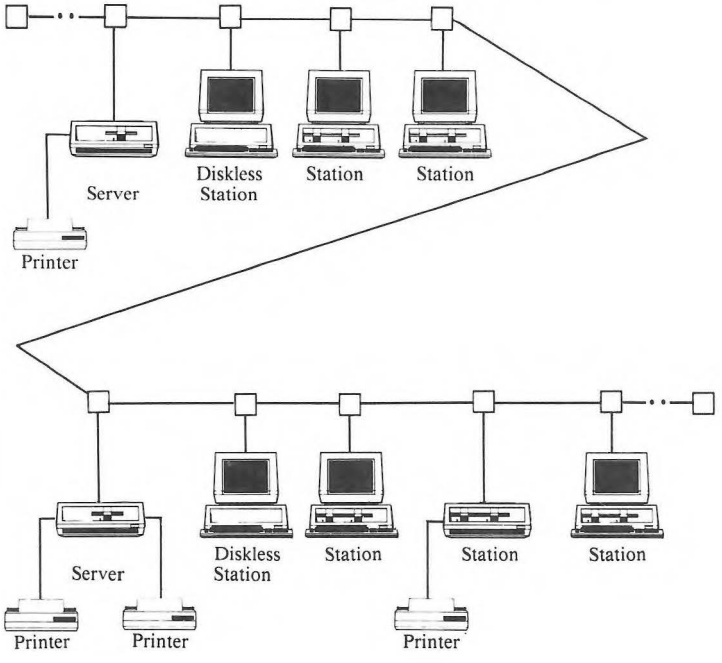
A simple Network could involve just two regular (disk-equipped) computers connected with a segment of trunk line. One must act as a server and the other as a station. The server must have at least one hard disk. This is the minimum Network configuration. See Figure 15-3a.





A simple Network could also involve one server and one diskless station. The server provides all disk facilities. See Figure 15-3b.

Figure 15-4. Typical Network with Many Servers and Stations



All printers and disks for Network use are connected to the servers only, but all stations have access to them. (Printers and disks can also be attached to stations, but these are not accessible to the Network: they serve their station computers only.)

Any or all of the servers in Figure 15-4 can be equipped with keyboards and displays, as suits the user's convenience, though these fill no function

It is not necessary to regard any node's physical location as permanent. Any computer on the Network can be plugged into any junction box along the trunk cable. A new junction box can be spliced into the existing cable at a new location. Or the trunk may be extended between any two nodes (as illustrated in Figure 15-2) to the new location and a new junction box installed. Thus, any Network node is portable. Suppose you install an initial trunk line containing 40 junction boxes, but are initiating your Network with only eight computers. You can plug those eight computers into any eight of the junction boxes, as convenient. If you have to move a computer, you unplug it from its original junction box and just plug it into some other.

The trunk line is essentially a shared two-wire bus for all the computers in the Network. As far as any computer on this bus is concerned, all the other computers are in the same place. To "find" another node it needs to communicate with, the computer invokes that node's Network address, a two-digit number that does not reflect physical location.

15.5 IMPLEMENTATION

A Server Network has no central host computer. Nor do any of the server nodes serve as controllers for any particular set of station nodes. All computers in the Network are simply connected to the trunk line, in the same way, and share the line.

information packets addressed to it. The addresses run from 00 to 63, with 00 through 09 reserved for servers. The address for each node is set by means of DIP switch positions on its own Omninet board. The DIP switch settings are explained in the hardware installation instructions. It is not necessary to assign address numbers sequentially, either in terms of the computers' physical locations along the trunk line, or of the dates of their installation in the Network. Just be sure that no two nodes are assigned the same number.

All user messages, from any computer, are formatted by the Network board circuitry into information packets that also contain addressing and other overhead information. Each packet includes the address (00 to 63) of the intended receiving computer. Any computer can send a broadcast message to all others simultaneously by using the special address 255. A common broadcast message is the computer's beginning-of-operations signal that it is in the Network and ready for communications.

Each Network node can support up to four receiving "sockets." A socket is a separate or secondary address within the computer's memory to which incoming messages can be directed. The transmitting node knows the addresses of the receiving sockets, just as it knows the primary addresses (00 to 63) of all its sister nodes. (The socket addresses are uniform: in any computer

negative acknowledgement signal is sent back to the transmitting computer.

Interference among computers is controlled by means that are conceptually simple. When any computer in the Network has a message packet ready for transmission, it samples the line. If the line is quiet (no other node transmitting), the computer sends its packet. The packets are all brief (about 250 to 2000 bytes) and the sending node relinquishes the line at the end of each packet, affording other nodes an opportunity to transmit. The trunk line is considered quiet when receiving computers detect more than 15 consecutive one bits. (The Network board hardware automatically inserts zero bits as needed in a packet, so that no long run of one bits occurs during a data transmission.)

A server computer's hard disk can be divided, by means of the operating software, into up to 14 data areas called volumes. If a volume is assigned to a particular user, it becomes a virtual disk drive for that user (in addition to any real disk drives on his or her station computer). That is, if the user has floppy disk drives A and B on his or her computer, and is assigned two volumes, these become his or her disk drives C and D. If the user is using some other computer, he or she still has access to "drives" C and D. A volume may also be used as a shared data space among several users. The Network software has provision for passwords, reserved files, and so on, so that data can be protected from inadvertent change or unauthorized access.

APPENDIX A: NETWORK ROUTINES IN BASIC PROGRAMS

In BASIC you use the CALL statement syntax for passing parameters to the Server Network high-level interface. Section A.1 defines the BASIC calling sequences for the Server Network functions. Section A.2 is a short sample program that demonstrates reserving a file for shared access, locking a record, modifying the record, unlocking the record, and finally releasing all reservations.

A.1 CALLING SEQUENCES

- For a Network routine in any BASIC program, the last two parameters passed must be:

  -- E.RROR.CODE%, the integer variable to receive the error code (see Table 13-2).

  -- FUNCTION.CODE%, the integer variable containing the function request code (see Table 13-1).

- In Interpretive BASIC, the DEF SEG must be set to the value SERV.SEG for Network functions (see Section 14.1). For compiled BASIC, the DEF SEG is not necessary.

- You must initialize all variables before calling the interface (see examples).

```
IF E.RROR.CODE% <> 0 THEN PRINT "error:";
E.RROR.CODE%
```

You can also set up a string array of all the error codes (see Table 13-2) and then print the one(s) returned, as in the sample program in Section A.2.

FUNCTION: RESERVE FILE

Parameter: FLNME$ = filename

Error Codes Returned: 53, 64, 128, 129, 130, 131, 138, 141, 144, 145

Example:

```
 5  DEF SEG                                  'initialize to BASIC's DS
10  FLNME$ = "TEST.DOC" + ""                 'set filename into string
15  FUNCTION.CODE% = RESERVE.FILE%           'initialize function request
20  CALL SERVER (FLNME$, E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: RESERVE FOR SHARED ACCESS

Parameter: FLNME$ = filename

Error Codes Returned: 53, 64, 128, 129, 130, 131, 138, 141, 144, 145

Example:

```
 5  DEF SEG                                          'initialize to BASIC's DS
10  FLNME$ = "TEST          " + ""                   'setup filename
15  FUNCTION.CODE% = RESERVE.SHARED.ACCESS%  'initialize function request
20  CALL SERVER (FLNME$, E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: LOCK RECORD

Parameters: KEY$ = character code (up to 64 characters)
DRIVE% = 0-15 (0 = default, 1 = A:, ...)

Error Codes Returned: 128, 129, 130, 131, 138, 139, 141, 145, 146

Example:

```
 5  DEF SEG                                          'initialize to BASIC's DS
10  KEY$ = "rec:10" + ""                             'setup key for record
15  DRIVE% = 0                                       'setup default drive
20  FUNCTION.CODE% = LOCK.RECORD%                    'initialize function request
25  CALL SERVER (KEY$, DRIVE%, E.RROR.CODE%, FUNCTION.CODE%)
```

characters)
DRIVE% = 0-15 (0 = default, 1 = A:,...)

Error Codes Returned: 128, 129, 130, 134, 138, 139, 141, 145

Example:

```
 5  DEF SEG                                    'initialize to BASIC's DS
10  KEY$ = "rec:10" + ""                       'setup key for record
15  DRIVE% = 0                                 'setup default drive
20  FUNCTION.CODE% = UNLOCK.RECORD%            'initialize function request
25  CALL SERVER (KEY$, DRIVE%, E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: RELEASE FILE

Parameter: FLNME$ = filename

Error Codes Returned: 53, 64, 128, 129, 130, 131, 132, 133, 138, 141, 144, 145

Example:

```
 5  DEF SEG                                    'initialize to BASIC's DS
10  FLNME$ = "TEST" + ""                       'set filename into string
15  FUNCTION.CODE% = RELEASE.FILE%             'initialize function request
20  CALL SERVER (FLNME$, E.RROR.CODE%, FUNCTION.CODE%)
```

Example:

```
 5  DEF SEG                                    'initialize to BASIC's DS
10  FUNCTION.CODE% = RELEASE.ALL%              'initialize function request
15  CALL SERVER (E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: SET PROTECT

Parameters: FLNME$ = filename
ATTR% = protection status:
0 = Read/WRITE (bit 0)
1 = READ-only (bit 0)
2 = Hidden file (bit 1)
4 = System file (bit 2)
20H = Archive (bit 5)

Error Codes Returned: 53, 64, 128, 129, 130, 131, 132, 138, 141, 144, 145

Example:

```
 5  DEF SEG                                    'initialize to BASIC's DS
10  FLNME$ = "B:TEST.DOC" + ""                 'set filename into string
15  ATTR% = 1                                  'make it read-only
20  FUNCTION.CODE% = SET.PROTECT%              'initialize function request
25  CALL SERVER (FLNME$, ATTR%, E.RROR.CODE%, FUNCTION.CODE%)
```

ATTR% = Variable to receive attributes

Returns:
ATTR% = 0 Read/WRITE (bit 0)
ATTR% = 1 READ-only (bit 0)
ATTR% = 2 Hidden file (bit 1)
ATTR% = 4 System file (bit 2)
ATTR% = 8 Volume ID (bit 3)
ATTR% = 10H Directory (bit 4)
ATTR% = 20H Archive (bit 5)

Error Codes Returned: 53, 64, 128, 129, 130, 131, 132, 138, 141,145

Example:

```
 5  DEF SEG                                        'initialize to BASIC's DS
10  FLNME$ = "B:TEST.DOC" + ""                     'set filename into string
15  FUNCTION.CODE% = GET.PROTECT%                  'initialize function request
20  CALL SERVER (FLNME$, ATTR%, E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: GET SERVERSTAT

PARAMETERS: SERV.NAM$ = server version
STATUS.ARRAY%(0) = server name (16 char volume name of system drive on the server)
STATUS.ARRAY%(1) = number of drives available on server
STATUS/ARRAY%(2-17) = drive type (0 = hard, 0FFH = floppy)

```
                    STATUS.ARRAY%(21-23) = number of
                                           entries per
                                           printer
                    STATUS.ARRAY%(24) = total Network
                                        open files
                    STATUS.ARRAY%(25) = total Network
                                        reserved files
                    STATUS.ARRAY%(26) = number of user
                                        open files
                    STATUS.ARRAY%(27) = number of user
                                        reserved files
```

Error Codes Returned: 129, 130, 136, 140, 145

Example:

```
 5  DIM STATUS.ARRAY%(27)
10  SERV.NAM$ = STRING$(16,+ " ")          'initialize string
15  FUNCTION.CODE% = GET.SERVERSTAT%       'initialize function request
20  E.RROR.CODE%=0
25  STATUS.PTR% = VARPTR(STATUS.ARRAY%(0)
30  DEF SEG                                'initialize to BASIC's DS
35  CALL SERVER (SERV.NAM$,STATUS.PTR%,E.RROR.CODE%,FUNCTION.CODE%)
```

NOTE: SERV.NAM$, E.RROR.CODE%, and FUNCTION.CODE% must be initialized before STATUS.PTR%, and no other variable assignments can be between the STATUS.PTR% initialization and the CALL statement.

SERVER.ID% = server number (0 - 9)

Error Codes Returned: 128, 129, 130, 138, 141, 145

Example:

```
 5  DEF SEG                                        'initialize to BASIC's DS
10  DRIVE% = 0                                     'set to default drive
15  FUNCTION.CODE% = GET.SERVER.ID%                'initialize function request
20  CALL SERVER (DRIVE%, SERVER.ID%, E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: TEST STATION

Parameter: STATION.ID% = station for testing (0 - 63)

Error Codes Returned: 129, 130, 140, 142, 145

Example:

```
 5  DEF SEG                                        'initialize to BASIC's DS
10  STATION.ID% = 5                                'setup to test station 5
15  FUNCTION.CODE% = TEST.STATION%                 'initialize function request
20  CALL SERVER (STATION.ID%, E.RROR.CODE%, FUNCTION.CODE%)
```

DRV.INFO% = variable to receive drive information

Returns:
- DRV INFO%(0) = free allocation units
- DRV INFO%(1) = sectors per allocation unit
- DRV INFO%(2) = bytes per sector
- DRV INFO%(3) = total allocations on specified drive

Error Codes Returned: 128, 129, 130, 131, 138, 141, 145

Example:

```
 5  DIM DRV.INFO%(4)                        'reserve array space
10  DEF SEG                                 'initialize to BASIC's DS
15  DRIVE% = 2                              'select drive B
20  FUNCTION.CODE% = GET.FREE.SPACE%  'initialize function request
25  E.RROR.CODE%=0
30  DRV.INFO.PTR%=VARPTR(DRV.INFO%(0))
35  CALL SERVER (DRIVE%, E.RROR.CODE%, FUNCTION.CODE%)
```

NOTE: The assignment of DRV.INFO.PTR% must be the last variable assignment before the CALL statement.

FUNCTION: GET USERNAME

Parameter: USER.NAME$ = user name

Error Codes Returned: 129, 130, 135, 145

Example:

```
 5  USER.NAME$ = STRING$(11," ") + ""      'initialize string to 11 chars
10  FUNCTION.CODE% = GET.USERNAME%         'initialize flag
15  DEF SEG                                'initialize to BASIC's DS
20  CALL SERVER (USER.NAME$, E.RROR.CODE%, FUNCTION.CODE%)
```

FUNCTION: GET MYID

Parameter: MYID% = variable to receive
identification (10 - 63)

Error Codes Returned: 129, 130, 145

Example:

```
 5  DEF SEG                                'initialize to BASIC's DS
10  FUNCTION.CODE% = GET.MYID%             'initialize function request
15  CALL SERVER (MYID%, E.RROR.CODE%, FUNCTION.CODE%)
```

function

Error Codes Returned: 129, 130, 145, 224-232, 253-255

Example:

```
 5  DEF SEG                                  ' initialize to BASIC's DS
10  FUNCTION.CODE% = GET.NETWORK.E.RROR% ' initialize function request
20  CALL SERVER (E.RROR.CODE%, FUNCTION.CODE%)
```

```
1000  '*****************************************************************************
****
      '#####        BASIC File Server Interface Demonstration Program          #####
      '#############################################################################
1005  '-----------------------------------------------------------------------------
       -----        Routine to Initialize function #'s & Error Strings          -----
       -----------------------------------------------------------------------------
1010   DIM E.RROR$(20)                          'set up error string space
1015   RESTORE 3025
      :FOR E.RROR% = 0 TO 17
         :READ E.RROR$(E.RROR%)
      :NEXT E.RROR%                            'load error string array
1020   RELEASE.ALL% = 23
      :RESERVE.SHARED.ACCESS% = 30
      :LOCK.RECORD% = 31
      :UNLOCK.RECORD% = 32                     'load function request numbers
1025   DATA "Error. File not found", "Error. Bad file name", "Error. Local Drive",
          "Network error.", "Error. Station not active.", "Error. Access denied."
1030   DATA "Error. Invalid attribute.", "Error. File not reserved.",
          "Error. Key not reserved.", "Error. Invalid name string length",
          "Error. Invalid server name string length."
1035   DATA "Error. Invalid filename string length.",
          "Error. Invalid drive specification", "Error. Invalid key string lenth"
1040   DATA "Error. Server not active.", "Error. Drive not mapped",
          "Error. Invalid station id number.", "Error. No reservations"
1045   '-----------------------------------------------------------------------------
       -----                        Main Program                               -----
       -----------------------------------------------------------------------------
1050   FLNME$ = "Data.bas"                      'get filename for next call
1055   GOSUB 2000                               'do reserve for shared access
1060   IF E.RROR.CODE% THEN GOTO 6000           'display error & terminate
1065   OPEN "Random", #1, FLNME$, 128           'set up to modify a record
1070   FIELD #1, 40 AS NAME$, 5 AS ADDR$ 30 AS STREET$, 30 AS CITY$
                12 AS STATE$, 5 AS ZIP$
1075   KEY$ = FLNME$ +":0"                      'set up to lock 1st record
1080   GOSUB 3000                               'do a lock record request
1085   IF E.RROR.CODE% THEN GOTO 6000           'display error & terminate
1090   GET #1                                   'get 1st record
1095   LSET NAME$ = "New Name"                  'modify record
1100   PUT #1                                   'put modified record back
1105   GOSUB 4000                               'do an unlock record request
1110   IF E.RROR.CODE% THEN GOTO 6000           'display error & terminate
1112   GOSUB 5000                               'release all reservation
1114   IF E.RROR.CODE% THEN GOTO 6000           'display error & terminate
1115   CLOSE #1                                 'set up to terminate modification
1120   PRINT "success!"                         'end of modification
1125   END
1995   '-----------------------------------------------------------------------------
       -----          Subroutine Reserve for Shared Access                    -----
       -----------------------------------------------------------------------------
2000   FUNCTION.CODE% = RESERVE.SHARED.ACCESS%  'get function request number
2005   FLNME$ = FLNME$ + ""                     'make sure string is in data space
2010   CALL SERVER (FLNME$, E.RROR.CODE%, FUNCTION.CODE%)  'call the interface
2015   RETURN
2995   -----------------------------------------------------------------------------
       -----          Subroutine Lock Record                                  -----
       -----------------------------------------------------------------------------
3000   FUNCTION.CODE% = LOCK.RECORD%            'get function request number
3005   KEY$ = KEY$ + ""                         'make sure string is in data space
3010   CALL SERVER (KEY$, E.RROR.CODE%, FUNCTION.CODE%)    'call the interface
3015   RETURN
3995   '-----------------------------------------------------------------------------
       -----          Subroutine Unlock record                                -----
       -----------------------------------------------------------------------------
4000   FUNCTION.CODE% = UNLOCK.RECORD%          'get function request number
4005   KEY$ = KEY$ + ""                         'make sure string is in data space
4010   CALL SERVER (KEY$, E.RROR.CODE%, FUNCTION.CODE%)     'call the interface
4015   RETURN
4995   '-----------------------------------------------------------------------------
       -----          Subroutine Release All                                  -----
       -----------------------------------------------------------------------------
5000   FUNCTION.CODE% = RELEASE.ALL%            'get function request number
5005   CALL SERVER E.RROR.CODE%, FUNCTION.CODE%  'call the interface
5010   RETURN
5995   '-----------------------------------------------------------------------------
       -----          Routine to Display Error & Terminate                    -----
       -----------------------------------------------------------------------------
6000   IF E.RROR.CODE% < 53 THEN OFFSET% = 0    'get offset into string
6005   IF E.RROR.CODE% = 64 THEN OFFSET% = 1    'get offset into string
6010   IF E.RROR.CODE% > 64 THEN OFFSET% = E.RROR.CODE - 126   'get offset
6015   PRINT E.RROR$(OFFSET%)
6020   END
```

APPENDIX B: DECLARATIONS FOR MS-PASCAL

High-level Network functions must be declared as EXTERNAL procedures in your MS-Pascal program. This appendix lists the EXTERNAL declarations for all the high-level routines described in Chapter 12.

B.1 TYPE DECLARATIONS

For the EXTERNAL declarations listed in this appendix, you must declare the following types:

```
TYPE
  SERVER_RECORD = RECORD
    server_version          [00]:    WORD;
    server_name             [02]:    STRING(16);
    number_drives           [18]:    WORD;
    drive_type              [20]:    ARRAY[0..15] OF
                                     BYTE;
    printer_status          [36]:    ARRAY[0..2] OF
                                     BYTE;
    queue_size              [39]:    ARRAY[0..2] OF
                                     WORD;
    num_open_files          [45]:    WORD;
    num_res_files           [47]:    WORD;
    user_open_files         [49]:    WORD;
    user_res_files          [51]:    WORD;
    reserved                [53]:    ARRAY[0..15] OF
                                     WORD;

                    END;

  DRIVE_INFO_ARRAY = ARRAY[0..3] OF WORD;
```

two common parameters:

```
file_name = filename
err_code = error code variable
```

```
PROCEDURE Reserve_file(file_name : LSTRING;
          VARS err_code : WORD ); EXTERNAL;
```

Error Codes Returned: 53, 64, 128, 129, 130, 131, 138, 141, 144, 145

```
PROCEDURE Reserve_shared_access(file_name :
          LSTRING; VARS err_code : WORD );
          EXTERNAL;
```

Error Codes Returned: 53, 64, 128, 129, 130, 131, 138, 141, 144, 145

```
PROCEDURE Lock_record( key : LSTRING; drive :
          WORD; VARS err_code : WORD ); EXTERNAL;
```

where: key = character code (up to 64 characters)

Error Codes Returned: 128, 129, 130, 131, 138, 139, 141, 144, 145, 146

```
PROCEDURE Unlock_record( key : LSTRING;
          drive : WORD; VARS err_code : WORD );
          EXTERNAL;
```

```
   Error Codes Returned:  128, 129, 130, 134,
                          138, 139, 141, 145

PROCEDURE Release_file(file_name : LSTRING;
          VARS err_code : WORD ); EXTERNAL;

   Error Codes Returned:  53, 64, 128, 129, 130,
                          131, 132, 133, 138,
                          141, 145

PROCEDURE Release_all( VARS err_code : WORD );
          EXTERNAL;

   Error Codes Returned:  129, 130, 131, 143, 145

PROCEDURE Set_protect( file_name : LSTRING;
          attribute : WORD; VARS err_code :
          WORD ); EXTERNAL;

   where: attribute = sets protection status as:

                  0   = Read/write (bit 0)
                  1   = Read-only (bit 0)
                  2   = Hidden file (bit 1)
                  4   = System file (bit 2)
                 20H  = Archive (bit 5)

   Error Codes Returned:  53, 64, 128, 129, 130,
                          131, 132, 138, 141,
                          144, 145
```

where: attribute = variable that receives
protection status, as:

```
0   = Read/write (bit 0)
1   = Read-only (bit 0)
2   = Hidden file (bit 1)
4   = System file (bit 2)
8   = Volume ID (bit 3)
10H = Directory (bit 4)
20H = Archive (bit 5)
```

Error Codes Returned: 53, 64, 128, 129, 130, 131, 132, 138, 141, 145

```
PROCEDURE Get_server_status( VARS server_info :
          SERVER_RECORD; VARS err_code : WORD );
          EXTERNAL;
```

The Get server_status procedure copies the server's status information record (see Appendix A) into SERVER_RECORD.

Error Codes Returned: 129, 130, 136, 140, 145

```
PROCEDURE Get_server_id( drive : WORD;
          VARS server_id, err_code : WORD );
          EXTERNAL;
```

where: drive = drive number (0 = default, 1 = A:, ... 15 = O:)
server_id = 0 - 9

Error Codes Returned: 128, 129, 130, 138, 141, 145

```
   where:  station_id = station to test (0 - 63)

   Error Codes Returned:  129, 130, 140, 142, 145


PROCEDURE Get_free_space( drive : WORD;
          VARS drive_info : DRIVE_INFO_ARRAY;
          VARS err_code : WORD ); EXTERNAL;

   where:  drive = logical drive number (0 - 15)

   Error Codes Returned:  128, 129, 130, 131,
                          138, 141, 145


PROCEDURE Get_user_name( VARS user_name : LSTRING;
          VARS err_code : WORD ); EXTERNAL;

   Error Codes Returned:  129, 130, 135, 145


PROCEDURE Get_my_id( VARS myid, err_code :
          WORD); EXTERNAL;

   where: myid = variable to receive
                 identification (10-63)

   Error Codes Returned:  129, 130, 145


PROCEDURE Get_network_error( VARS err_code :
          WORD ); EXTERNAL;

   Error Codes Returned:  129, 130, 145, 224-232,
                          253-255
```

High-level Network functions must be declared as EXTERNAL using the MEDIUM case. This appendix lists the procedure declarations for the high-level routines described in Chapter 12.

```
    $medium code pw(109) pl(43) ot(1)
    $include(legal.plm)

    $include(plm.inc)


/*-------------------------------------------------
          Declare structures & Variables
-------------------------------------------------*/
dcl fcb struc(
    drive_number          byte,
    file_name       (8)   byte,
    extension       (3)   byte,
    current_block         word,
    record_size           word,
    file_size             dword,
    date                  word,
    time                  word,
    reserved        (8)   byte,
    current_record        byte,
    random_record   (2)   word    );
```

```
    server_name      (16) byte,
    number_drives         word,
    drive_type       (16) byte,
    print_status      (3) byte,
    queue_size        (3) word,
    no_open_files         word,
    no_reserve_files      word,
    user_open_files       word,
    user_resrv_files      word,
    server_reserved(16)   word     );


dcl drive_information     struc(
    sectors_alloc_unit    word,
    free_alloc_units      word,
    total_alloc_units     word,
    byte_per_sector       word     );

dcl drive_number          byte,
    error_code            byte,
    user_name        (16) byte,
    station_number        byte,
    server_id             byte,
    attribute             word,
    key              (63) byte;

/*-------------------------------------------------
                Procedure Declaration
-------------------------------------------------*/
Get_Free_Space: Proc      (drive_number, drive_info,
                           error_code) Ext;
    dcl drive_number      byte,
        drive_info        pointer,
        error_code        pointer;
End     Get_Free_Space;
```

```
End     Get_User_Name;

Test_Station:   Proc    (station_number,
                        error_code)     Ext;
    dcl station_number  byte,
        error_code      pointer;
End     Test_Station;

Get_Server_Status:      Proc    (server_info,
                        error_code)     Ext;
    dcl server_info     pointer,
        error_code      pointer;
End     Get_Server_Status;

Get_My_Id:      Proc    (station_number,
                        error_code)     Ext;
    dcl station_number  pointer,
        error_code      pointer;
End     Get_My_Id;

Reserve_File:   Proc    (fcb_ptr, error_code)
                        Ext;
    dcl fcb_ptr         pointer,
        error_code      pointer;
End     Reserve_File;

Release_File:   Proc    (fcb_ptr, error_code)
                        Ext;
    dcl fcb_ptr         pointer,
        error_code      pointer;
End     Release_File;

Release_All:    Proc    (error_code)    Ext;
    dcl error_code      pointer;
End     Release_All;
```

```
    dcl drive_number    byte,
        server_id       pointer,
        error_code      pointer;
End     Get_Server_Id;

Set_Protect:    Proc    (fcb_ptr, attribute,
                        error_code)     Ext;
    dcl fcb_ptr         pointer,
        attribute       byte,
        error_code      pointer;
End     Set_Protect;

Get_Protect:    Proc    (fcb_ptr, attribute,
                        error_code)     Ext;
    dcl fcb_ptr         pointer,
        attribute       pointer,
        error_code      pointer;
End     Get_Protect;

Reserve_Shared_Access:  Proc    (fcb_ptr,
                        error_code)     Ext;
    dcl fcb_ptr         pointer,
        error_code      pointer;
End     Reserve_Shared_Access;

Lock_Record:    Proc    (key, error_code)
                        Ext;
    dcl key             pointer,
        drive_number    byte,
        error_code      pointer;
End     Lock_Record;

Unlock_Record:  Proc    (key, error_code)
                        Ext;
    dcl key             pointer,
        drive_number    byte,
```

```
Get_Network_Error:      Proc    (error_code)
                        Ext;
    dcl error_code      pointer;
End     Get_Network_Error;
```

High-level Network procedures must be declared as EXTERNAL:FAR in your program. This appendix lists all declarations needed for the procedures described in Chapter 12. In addition, examples of how each procedure is called clarify the parameter passing.

```
;-------------------------------------------------
;     Msdos File Server Information Structure
;-------------------------------------------------
servr_info      struc
    serv_ver    dw      ?           ;server version #
    serv_name   db      16 dup(?)   ;server name string
    num_drvs    dw      ?           ;number of server drives
    drv_typ     db      16 dup(?)   ;drive types
    pri_stat    db      3  dup(?)   ;printer status
    que_siz     dw      3  dup(?)   ;printer queues awaiting
    no_ofil     dw      ?           ;total # of network open files
    no_rfil     dw      ?           ;total # of network reserved files
    us_ofil     dw      ?           ;total # of user open files
    us_rfil     dw      ?           ;total # of user reserved files.
    serv_resrv  dw      16 dup(?)   ;reserved for future use
servr_info      ends
```

```
fcb_    struc
        flg             db      ?               ;flags extended fcb
        zer_fld         db      5 dup (0)       ;zero field
        atr_byt         db      ?               ;bit 1 = 1, search hidden files
                                                ;bit 2 = 1, search system files
        drv_cod         db      ?               ;drive code
        fil_nme         db      8 dup (?)       ;file name
        fil_ext         db      3 dup (?)       ;file extension
        cur_blk         db      2 dup (?)       ;current block
        rec_siz         db      2 dup (?)       ;record size
        fil_siz         db      4 dup (?)       ;file size
        date_           db      2 dup (?)       ;date with record as above
        time_           db      2 dup (?)       ;time with record as above
        reserved        db      8 dup (?)       ;reserved for Msdos
        cur_rec         db      ?               ;cur rec within blk for seq
        ran_rec         db      4 dup (?)       ;random record
fcb_    ends
```

```
    byte_per_sector     dw      ?
drv_info        ends

dgroup  group  data
data    segment public  'data'
     assume  ds:dgroup

fcb     fcb_    <,,,,,,,,,,>            ;extended file control block

server_info     servr_info      <,,,,,,,,,,>    ;define the structure

drive_info      drv_info        <,,,>    ;define drive information
                                             ;array

error_code      db   false    ;variable to receive errors
drive_number    dw   ?        ;   "      to pass logical drive number
station_number  dw   ?        ;   "      to pass station number to test
server_id       db   true     ;   "      for server identification
attribute       dw   ?        ;   "      for file attributes
key             db   64 dup (?)       ;key string array
user_name       db   11 dup (?)       ;user name string array

data    ends

cgroup  group  code
code    segment public  'code'
        assume  cs:cgroup

 extrn   GET_MY_ID:far, GET_SERVER_ID:far, TEST_STATION:far
 extrn   GET_FREE_SPACE:far, SET_PROTECT:far, GET_PROTECT:far
 extrn   RESERVE_FILE:far, RESERVE_SHARED_ACCESS:far, LOCK_RECORD
 extrn   UNLOCK_RECORD:far, RELEASE_FILE:far, RELEASE_ALL:far
 extrn   GET_USER_NAME:far, GET_SERVER_STATUS:far
 extrn   GET_NETWORK_ERROR:far
```

```
     call    get_network_error
     ret

get_network_error$      endp

GET_MY_ID$      proc    near
     mov     ax, offset dgroup:station_number
     push    DS
     push    ax

     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    get_my_id
     ret

get_my_id$      endp

GET_SERVER_ID$  proc    near
     mov     ax, offset dgroup:server_id
     push    DS
     push    ax

     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    get_server_id
     ret

get_server_id$  endp
```

```
        mov     ax, offset dgroup:error_code
        push    DS
        push    ax

        call    test_station
        ret

test_station$   endp

GET_FREE_SPACE$ proc    near
        push    drive_number

        mov     ax, offset dgroup:drive_info
        push    DS
        push    ax

        mov     ax, offset dgroup:error_code
        push    DS
        push    ax

        call    get_free_space
        ret

        get_free_space$ endp

SET_PROTECT$    proc    near
        mov     ax, offset dgroup:fcb.drv_cod
        push    DS
        push    ax

        push    attribute

        mov     ax, offset dgroup:error_code
        push    DS
        push    ax
```

```
set_protect$     endp

GET_PROTECT$     proc    near
     mov     ax, offset dgroup:fcb.drv_cod
     push    DS
     push    ax

     mov     ax, offset dgroup:attribute
     push    DS
     push    ax

     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    set_protect
     ret

get_protect$     endp

RESERVE_FILE$    proc    near
    mov     ax, offset dgroup:fcb.drv_cod
    push    DS
    push    ax

    mov     ax, offset dgroup:error_code
    push    DS
    push    ax

    call    reserve_file
    ret

reserve_file$    endp

RESERVE_SHARED_ACCESS$  proc    near
     mov     ax, offset dgroup:fcb.drv_cod
```

```
     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    reserve_shared_access
     ret

reserve_shared_access$  endp

LOCK_RECORD$    proc    near
     mov     ax, offset dgroup:key
     push    DS
     push    ax

     push    drive_number

     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    lock_record
     ret

lock_record$    endp

UNLOCK_RECORD$  proc    near
     mov     ax, offset dgroup:key
     push    DS
     push    ax

     push    drive_number

     mov     ax, offset dgroup:error_code
     push    DS
     push    ax
```

```
unlock_record$  endp

RELEASE_FILE$   proc    near
     mov     ax, offset dgroup:fcb.drv_cod
     push    DS
     push    ax

     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    release_file
     ret

release_file$   endp

RELEASE_ALL$    proc    near
     mov     ax, offset dgroup:error_code
     push    DS
     push    ax

     call    reserve_file
     ret

release_all$    endp
```

```
        push    ax

        mov     ax, offset dgroup:error_code
        push    DS
        push    ax

        call    reserve_file
        ret

get_user_name$  endp

GET_SERVER_STATUS$      proc    near
        mov     ax, offset dgroup:server_info
        push    DS
        push    ax

        mov     ax, offset dgroup:error_code
        push    DS
        push    ax

        call    get_server_status
        ret

get_server_status$      endp

CODE    ENDS
```

shows you how to configure your 30 MB hard disk as a file server. Use these instructions to configure your 30 MB file server after you have made backup copies of all your software diskettes. (Chapter 7 of your Server Network User's Guide explains how to make backups.)

Use the diskette labeled **Server Distribution, 30 MB** to configure your 30 MB file server. This diskette is referred to as the **Server System Build diskette** throughout the remainder of this supplement. You can configure all 30 MB servers with this diskette.

Determining the Number of Volumes

Before you configure your 30 MB file server, determine the number of volumes you will use. Base this decision on the physical configuration of your network, the capacity of the servers' hard disks (thirty megabytes in this case), and an understanding of how each user will use the network. In this installation, the size of the spool queue is automatically set to 200, the largest queue size available.

The Configuration Process

The configuration process consists of initializing your hard disk and then installing the Server Network software. If you are using a standard server configuration, use your Server System Build diskette for the entire process. If you are using a custom configuration, you must use both the Hard Disk Tool

Standard Configuration

The standard configuration process consists of booting from your Server System Build diskette and running a batch file that initializes the hard disk volumes and installs the Server Network software.

You must know the server number when you configure your server because the installation process checks whether the correct server number is being configured.

CAUTION: Configuring the server destroys all files on that server's hard disk.

Follow these steps to configure the server.

1. Put the Server System Build diskette into the floppy disk drive and boot the system.

2. Begin server configuration by running the batch file SERVER.BAT. Type

   server n 3v(cr)

   where:

   - o n is a number 0 - 9 that is the number of the server you are configuring. (All server networks must have a Server 0.)

   - o v is a number 1 - 4, 6, or 8 that is the number of user-volumes on the hard disk. Choose a number of volumes that provides sufficient

If the number n entered does not match the address dip-switch number on the network card, an audio message informs you continuously until you reboot the system. After you reboot, run the SERVER.BAT batch file again, entering the number n that matches the address dip switch on the network card. See the DISKID file of the Server Distribution (1 of 2) diskette for more information on the dip switch settings for the network card.

The SERVER.BAT file invokes the AUTOSET program, which initializes your hard disk. If your hard disk has been initialized previously, AUTOSET asks you whether you want to back up the data on your hard disk and whether you really want to initialize the hard disk. Answer these questions appropriately to continue.

If you indicate with a YES answer that you want to initialize the hard disk, AUTOSET initializes the hard disk. Then, AUTOSET instructs you to put a Hard Disk Operating System diskette in the floppy disk drive and press any key.

Leave the Server System Build diskette in the floppy disk drive and press any key. After you press a key, the system reboots from the Server System Build diskette with your hard disk initialized.

The rest of the configuration process continues without your input, although you might gain a greater understanding of the configuration process by observing it. The rest of the process includes:

- Copying the Server Network Software and all supporting files.

When the process is complete, an audio message reports the success or failure of the configuration process.

Here is an example of a typical server configuration command line:

server 0 34(cr)

This command configures Server 0's 30 MB hard disk as:

- A 700K Volume A
- A floppy disk drive, Volume B
- 4 7200K (approx.) volumes, Volumes C through F

Another example of a typical server configuration command line is as follows:

server 2 36(cr)

This command configures Server 2's 30 MB hard disk as:

- A 700K Volume A
- A floppy disk drive, Volume B
- 6 5000K (approx.) volumes, Volumes C through H

Creating Custom-Size Volumes

If you want to use a server configuration other than 1 - 4, 6, or 8 user-volumes, you must use the Hard

configuring.

2. Read the HDSETUP section of the Hard Disk Tool Kit manual as well as the Hard Disk Tool Kit II (version 2.2) DISKID file.

3. Insert the Hard Disk Tool Kit II (version 2.2) diskette (1 of 2) in the server's B drive and type:

   hdsetup(cr)

   This command loads the HDSETUP utility into memory. To establish custom-size hard disk volumes, follow these guidelines as you make selections in HDSETUP:

   - Volume A must be the System Boot volume. This volume is used by the network. Network users cannot address Volume A. You should allocate 700K for the System Volume.

   - Volume B must be the floppy disk drive.

   - Volumes C - O can be the volumes for network users. Volume C, however, is also used to store spooled printer output, so you might want to either increase the size of Volume C or limit the number of users who can access Volume C.

   - Minimize the allocation unit (AU) size to conserve disk space.

5. Run the SERVER.BAT file by typing

server n no(cr)

where n is the number of the server you are configuring and **no** indicates that the hard disk is already initialized.

If the number n entered does not match the address dip switch number on the network card, an audio message informs you continuously until you reboot the system. After you reboot, run the SERVER.BAT batch file again, entering the number n that matches the address dip switch on the network card. See the DISKID file of the Server Distribution (1 of 2) diskette for more information on the address dip switch settings for the network card.

The rest of the configuration process continues without your input, although you might gain a greater understanding of the configuration process by observing it. The rest of the process includes:

- Copying the MS-DOS 2.11 system tracks to the server A Volume.
- Copying the server software and all supporting files.

When the process is complete, an audio message reports the success or failure of the initialization process.