

Module 9

File Systems

9.1 OBJECTIVES

Students will be able to:

- Identify acceptable file names for use with FAT and NTFS partitions
- Understand New Technology File System (NTFS)
- Understand File Allocation Table 16 (FAT16)
- Understand File Allocation Table 32 (FAT32)
- Understand basic DOS commands

9.2 OVERVIEW

In this module we will discuss the various file systems that are used with different Microsoft Operating Systems. We will cover the advantages between FAT and NTFS, discuss the differences between FAT16, FAT32, and NTFS and how to decide which file system to use. We will also discuss basic DOS commands and utilities.

Operating System	File System Format
Windows 2000	NTFS
	FAT16
	FAT32
Microsoft Windows NT version 4.0	NTFS
	FAT16
Windows 95 OSR2 and Windows 98	FAT16
	FAT32
MS-DOS	FAT16

Table 9-1. Operating Systems Supported File Systems

9.3 FAT 16

File Allocation Table (FAT) refers to a disk format, which is a way of organizing the storage space on a hard disk. The table organizes information about the files on the hard disk, representing each one as a chain of numbers that identifies where each part of a file is located. The FAT itself is similar to a table of contents in a book – the operating system uses it to look up a file and find which cluster that file is written to on the hard disk.

FAT is probably the most widely recognized disk format, being read by most operating systems. Microsoft originally devised FAT to manage files on floppy disks, and adapted it as a standard for file and disk management in MS-DOS. A 12-bit fat was first used for managing floppy disk and logical drives smaller than 16 MB. MS-DOS version 3.0 introduced the 16-bit FAT for larger drivers.

9.3.1 ADVANTAGES OF FAT 16

- MS-DOS, Windows 95, Windows 98, Windows NT, Windows 2000, and some UNIX operating systems can use FAT16.

- There are many software tools that can address problems and recover data on FAT16 volumes.
- If you have a startup failure, you can start the computer by using an MS-DOS bootable floppy disk to troubleshoot the problem.
- FAT16 is efficient, in speed and storage, on volumes smaller than 256MB.

9.3.2 DISADVANTAGES OF FAT 16

- The root folder can manage a maximum of 512 entries. The use of long file names (LFNs) can significantly reduce the number of available entries.
- FAT16 is limited to 65,536 (64 KB) clusters, but because certain clusters are reserved, it has a practical limit of 65,624. The largest FAT16 volume on Windows 2000 is limited to 4 GB and a cluster size of 64 KB, however, on Windows NT 4.0, Windows 95 and Windows 98 the volume size is limited to 2 GB.
- FAT16 is inefficient on larger volume sizes, as the size of the cluster increases. The space allocated for storing a file is based on the size of the cluster allocation granularity, not the file size. For example, a 10 KB file stored on a 1.2 GB volume, which uses a 32 KB cluster, wastes 22 KB of disk space.
- The boot sector is not backed up.
- There is no built-in file system security or compression scheme with FAT16.

9.4 FAT32

FAT32 goes beyond the capabilities of FAT16. The most prominent feature is that it supports drives of up to 2 terabytes (TB) in size. In addition, FAT32 decreases the cluster size on large drives, thus reducing the amount of unused space. For example, with FAT16, a 2 GB drive has a 32 KB cluster size. The same driver under FAT32 has 4 KB clusters.

To maintain the greatest possible compatibility with existing programs, networks, and device drivers, FAT32 was implemented with as little change as possible existing architecture, internal data structures APIs, and on-disk format for Windows 98.

All of Microsoft's bundled disk tools (Format, FDISK, Defrag, and MS-DOS-based and Windows-based ScanDisk) have been revised to work with FAT32. In addition, Microsoft is working with leading device driver and disk tool vendors to support them in revising their products to support FAT32.

9.4.1 ADVANTAGES OF FAT32

FAT32 has the following enhancements:

- The root folder on a FAT32 drive is an ordinary cluster chain and can be located anywhere on the volume. For this reason, FAT32 does not restrict the number of entries in the root folder.

- FAT32 uses smaller clusters (4 KB for volumes up to 8GB), so it allocated disk space more efficiently than FAT16. Depending on the size of your files, FAT32 creates the potential for tens and even hundreds of megabytes of additional free disk space on larger volumes compared to FAT16.
- FAT32 can automatically use the backup copy of the file allocation table instead of the default copy (with FAT16, only a disk repair tool such as Chkdsk can implement the backup.)
- The boot sector is automatically backed up at a specified location on the volume, so FAT32 volumes are less susceptible to single points of failure than FAT16 volumes.

9.4.2 DISADVANTAGES OF FAT 32

- The largest FAT32 volume that Windows 2000 can format is 32 GB. On Windows 95 OSR2, and Windows 98, FAT32 supports drives up to 2 terabytes in size.
- FAT32 volumes are not directly accessible from operating systems other than Windows 95 OSR2 and Windows 98.
- If you have a startup failure, you cannot start the computer by using an MS-DOS or Windows 95 (excluding version OSR2 and later) bootable floppy disk.
- There is no built-in file system security or compression scheme with FAT32.

9.5 VIRTUAL FILE ALLOCATION TABLE (VFAT)

The 32-bit VFAT driver provides a 32-bit protected-mode code path for manipulating the file system stored on a disk. It is also re-entrant and multithreaded, providing smoother multitasking performance. The 32-bit file access driver improved over the driver originally provided with Windows for Workgroups 3.11, and is compatible with more MS-DOS-device drivers and hard disk controllers.

Under MS-DOS and Windows 3.1, manipulation of the FAT and writing to or reading from the disk is handled by the Int21h MS-DOS function and is 16-bit real-mode code. Being able to manipulate the disk file system from protected mode removes or reduces the need to transition to real mode in order to write information to the disk through MS-DOS. This results in a performance gain for file I/O access. The combination of the 32-bit file access and 32-bit disk access drivers results in significantly improved disk and file I/O performance.

9.6 NTFS

The Windows NT file system (NTFS) provides a combination of performance, reliability, and compatibility not found in the FAT file system. It is designed to quickly perform standard file operations such as read, write, and search – and even advanced operations such as file-system recovery – on very large hard disks.

The NTFS file system includes security features required for file servers and high-end personal computers in a corporate environment. The NTFS file system also supports data access control and

ownership privileges that are important for the integrity of critical data. While folders shared on a Windows NT computer are assigned particular permissions, NTFS files and folders can have permissions assigned whether they are shared or not. NTFS is the only file system on Windows NT that allows you to assign permissions to individual files. However, when you move or copy a file from an NTFS to a FAT volume, permissions and other attributes unique to the NTFS file system are lost.

The NTFS file system has a simple, yet very powerful design. Basically, everything on the volume in a file and everything in a file is an attribute, from the data attribute, to the security attribute, to the file name attribute. Every sector on an NTFS volume that is allocated belongs to some file. Even the file system metadata (information that describes the file system itself) is part of a file.

Like the FAT file system, the NTFS file system uses clusters as the fundamental unit of disk allocation. In the NTFS file system, the default cluster size depends on the volume size. In Disk Administrator, you can specify a cluster size to 4K. If you use the command prompt program **format** to format your NTFS volume, you can specify any cluster size supported by the NTFS file system.

9.6.1 ADVANTAGES OF NTFS

- NTFS is a recoverable file system. A user seldom needs to run a disk repair program on an NTFS volume. NTFS guarantees the consistency of the volume by using standard transaction logging and recovery techniques. In the event of a system failure, NTFS uses its log file and checkpoint information to automatically restore the consistency of the file system.
- NTFS supports compression on volumes, folders, and files. Files that are compressed on an NTFS volume can be read and written by any Windows-based application without first being decompressed by another program; decompression happens automatically during the file read. The file is compressed again when it is closed or saved.
- NTFS support all Windows 2000 file system features (the version that goes with Windows 2000) and the version that comes with NT 4.0 supports all file system features on NT4.0.
- NTFS does not restrict the number of entries in the root folder.
- Windows NT and Windows 2000 can format volumes up to 2 terabytes with NTFS.
- NTFS manages disk space more efficiently than FAT, using smaller clusters (4 KB for volumes up to 2 terabytes).
- The boot sector is backed up to a sector at the end of the volume.
- NTFS minimizes the number of disk accesses required to find a file.
- On NTFS volumes, you can set permissions on shares, folders, and files that specify which groups and users have access, and what level of access is permitted. NTFS file and folder permissions apply to users working on the local computer and to users accessing the file over the network from a shared folder. You can also set share permissions that operate on network shares in combination and file and folder permissions.

9.6.2 DISADVANTAGES OF NTFS

- NTFS volumes are not accessible for MS-DOS, Windows 95, or Windows 98.
- For very small volumes that contain mostly small files, the overhead of managing NTFS can cause a slight performance drop in comparison to FAT.
- Accessing the NTFS-formatted system volume when corrupted or deleted system files and prevents the computer from starting.

9.7 FILENAMES

9.7.1 NTFS AND FAT32 FILENAMES

9.7.1.1 NTFS AND FAT32 NAMING CONVENTIONS

Under NTFS and FAT32 the following is file naming conventions used when creating filenames:

- File and directory names can be up to 255 characters long, including any extensions.
- Names preserve case but are not case-sensitive. NTFS makes no distinction of filenames based on case.
- Names can contain any characters except for the following: ? " / \ < > * | :

9.7.1.2 LONG FILENAMES

NTFS and FAT32 is compatible with FAT, in that it will accept and use standard FAT 8.3 filenames, but NTFS and FAT32 also supports long filenames. NTFS and FAT32 keep an 8.3 filename along with the long filename. The NTFS and FAT32 8.3 filenames can be viewed in File Manager and at the command line using "DIR /X". By creating 8.3 filenames for files, NTFS and FAT32 allows MS-DOS and Windows 3.x applications to recognize and load these files even though they have NTFS and FAT32 long filenames. In addition, when an MS-DOS or Windows 3.x application saves a file on an NTFS and FAT32 volume, both the 8.3 filename and NTFS and FAT32 long filename are retained.

9.7.2 FAT16 NAMING CONVENTIONS

FAT uses the traditional 8.3 file naming convention and all filenames must be created with the ASCII character set.

- The name of a file or directory can be eight characters long, a period (.) separator, and a three-character extension.
- The name must start with either a letter or number and can contain any characters except for the following: . " / \ [] : ; | = ,

If any of these characters are used, unexpected results may occur.

- The name cannot contain any spaces.
- The following names are reserved: CON, AUX, COM1, COM2, COM3, COM4, LPT1, LPT2, LPT3, PRN, NUL
- All characters will be converted to upper case.

9.7.3 GENERATING AND VIEWING SHORT FILENAMES

Both NTFS and FAT32 filenames can be up to 256 characters and can contain spaces, multiple periods, and special characters that are illegal in MS-DOS (FAT16) filenames. NT and Windows 98 maintain these FAT 8.3 filenames for each long filename. This ensures that a program designed for FAT filenames can access and work with files created under Windows 98 and Windows NT. Long filenames also provide additional information about a file, such as the date of the last file modification.

9.7.3.1 GENERATING AND VIEWING SHORT FILENAMES

Windows 98 converts long filenames into DOS 8.3 format by using the following rules:

- Remove special characters (such as spaces).
- If unique, use the first eight characters of the long filename.
- If not unique, use the first six characters, a tilde (~), and a number (for example EASTCO~2.exe).
- For the extension, use the first three characters following the last period. (See Table 9-2 for examples.)

Long Filename	Short Filename
Monthly Report January	MONTHL~1
Monthly Report February	MONTHL~2
Monthly Report March	MONTHL~3
Monthly Report April	MONTHL~4

Table 9-2. Converting Long Filenames to Short Filenames in Windows 98

Use the *dir* command from the MS-DOS prompt to view both long and short filenames.

9.7.3.2 GENERATING SHORT FILENAMES UNDER WINDOWS NT 4.0

Windows NT 4.0 uses the same rules and Windows 98 to convert long filenames into short filenames except when there are 5 or more subsequent files that have the same first six letters. In these cases Windows NT:

- Use only the first two letters of the long filename.

- Generates the next four letters of the short filename by mathematically manipulating the remaining letters of the long filename.
- Appends ~1 (or another number, if necessary, to avoid a duplicate filename) to the result. (See Table 9-3)

Long Filename	Short Filename
Monthly Report January	MONTHL~1
Monthly Report February	MONTHL~2
Monthly Report March	MONTHL~3
Monthly Report April	MONTHL~4
Monthly Report May	MOOFF9~1.TXT
Monthly Report June	MOFEF5~1.TXT

Table 9-3. Converting Long Filenames to short Filenames under Windows NT

To view both long and short filenames from NT Command Prompt, type `dir /X`.

As you see, these filenames can be confusing, therefore it is suggested that the first 8 characters of your long filenames be unique. In the above tables, change files names to names like, January Monthly Report, February Monthly Report, March Monthly Report, etc.

9.8 DOS COMMANDS

The dreaded MS-DOS screen. It has been a long time since many of us had to remember how to manipulate systems using these DOS commands and utilities. However, there comes times when needing to maneuver in DOS is necessary, like during system installs, when having to boot with a floppy because the operating system is corrupt, etc. We will cover some of the basic commands and utilities with possible switches necessary. The files marked with an “*” are considered “internal commands” which means they are part of the boot file, “command.com”. The other files are considered “external commands” and are separate files and must be present to use that utility.

9.8.1 DIR*

Displays a list of files and subdirectories in a directory.

```
DIR [drive:][path][filename] [/P] [/W] [/A[:]attributes]
  [/O[:]sortorder] [/S] [/B] [/L] [/V] [/4]
```

```
[drive:][path][filename]
```

Specifies drive, directory, and/or files to list.

(Could be enhanced file specification or multiple filespecs.)

```
/P      Pauses after each screenful of information.
```

```
/W      Uses wide list format.
```

```
/A      Displays files with specified attributes.
```

```
attributes  D Directories          R Read-only files
```

```
            H Hidden files         A Files ready for archiving
```

```
            S System files         - Prefix meaning not
```

```
/O      List by files in sorted order.
```

sortorder N By name (alphabetic) S By size (smallest first)
E By extension (alphabetic) D By date & time (earliest first)
G Group directories first - Prefix to reverse order
A By Last Access Date (earliest first)
/S Displays files in specified directory and all subdirectories.
/B Uses bare format (no heading information or summary).
/L Uses lowercase.
/V Verbose mode.
/4 Displays year with 4 digits (ignored if /V also given).

Switches may be preset in the DIRCMD environment variable. Override preset switches by prefixing any switch with - (hyphen)--for example, /-W.

9.8.2 MAKE DIRECTORY (MD)*

Creates a directory.

```
MKDIR [drive:]path  
MD [drive:]path
```

9.8.3 CHANGE DIRECTORY (CD)*

Displays the name of or changes the current directory.

```
CHDIR [drive:][path]  
CHDIR[.]  
CD [drive:][path]  
CD[.]
```

.. Specifies that you want to change to the parent directory.

Type CD drive: to display the current directory in the specified drive.
Type CD without parameters to display the current drive and directory.

9.8.4 DELETE (DEL)*

Deletes one or more files.

```
DEL [drive:][path]filename [/P]  
ERASE [drive:][path]filename [/P]
```

[drive:][path]filename Specifies the file(s) to delete. Specify multiple files by using wildcards.

/P Prompts for confirmation before deleting each file.

9.8.5 REMOVE DIRECTORY (RD)*

Removes (deletes) a directory.

```
RMDIR [drive:]path
RD [drive:]path
```

9.8.6 CHANGING DRIVES*

9.8.7 DELETE TREE (DELTREE)

Deletes a directory and all the subdirectories and files in it.

To delete one or more files and directories:

```
DELTREE [/Y] [drive:]path [[drive:]path[...]]

/Y           Suppresses prompting to confirm you want to delete
             the subdirectory.
[drive:]path Specifies the name of the directory you want to delete.
```

Use DELTREE cautiously. Every file and subdirectory within the specified directory will be deleted.

9.8.8 COPY*

Copies one or more files to another location.

```
COPY [/A | /B] source [/A | /B] [+ source [/A | /B] [+ ...]] [destination
[/A | /B]] [/V] [/Y | /-Y]

source      Specifies the file or files to be copied.
/A         Indicates an ASCII text file.
/B         Indicates a binary file.
destination Specifies the directory and/or filename for the new file(s).
/V         Verifies that new files are written correctly.
/Y         Suppresses prompting to confirm you want to overwrite an
           existing destination file.
/-Y        Causes prompting to confirm you want to overwrite an
           existing destination file.
```

The switch /Y may be preset in the COPYCMD environment variable. This may be overridden with /-Y on the command line

To append files, specify a single file for destination, but multiple files for source (using wildcards or file1+file2+file3 format).

9.8.9 XCOPY

Copies files and directory trees.

XCOPY source [destination] [/A | /M] [/D[:date]] [/P] [/S [/E]] [/W]
[/C] [/I] [/Q] [/F] [/L] [/H] [/R] [/T] [/U]
[/K] [/N]

source Specifies the file(s) to copy.
destination Specifies the location and/or name of new files.
/A Copies files with the archive attribute set,
doesn't change the attribute.
/M Copies files with the archive attribute set,
turns off the archive attribute.
/D:date Copies files changed on or after the specified date.
If no date is given, copies only those files whose
source time is newer than the destination time.
/P Prompts you before creating each destination file.
/S Copies directories and subdirectories except empty ones.
/E Copies directories and subdirectories, including empty ones.
Same as /S /E. May be used to modify /T.
/W Prompts you to press a key before copying.
/C Continues copying even if errors occur.
/I If destination does not exist and copying more than one file,
assumes that destination must be a directory.
/Q Does not display file names while copying.
/F Displays full source and destination file names while copying.
/L Displays files that would be copied.
/H Copies hidden and system files also.
/R Overwrites read-only files.
/T Creates directory structure, but does not copy files. Does not
include empty directories or subdirectories. /T /E includes
empty directories and subdirectories.
/U Updates the files that already exist in destination.
/K Copies attributes. Normal Xcopy will reset read-only attributes.
/Y Overwrites existing files without prompting.
/-Y Prompts you before overwriting existing files.
/N Copy using the generated short names.

9.8.10 SYS

Copies MS-DOS system files and command interpreter to a disk you specify.

SYS [drive1:][path] drive2:

[drive1:][path] Specifies the location of the system files.
drive2: Specifies the drive the files are to be copied to.

9.8.11 FORMAT

Formats a disk for use with MS-DOS.

FORMAT drive: [/V[:label]] [/Q] [/F:size] [/B | /S] [/C]
FORMAT drive: [/V[:label]] [/Q] [/T:tracks /N:sectors] [/B | /S] [/C]
FORMAT drive: [/V[:label]] [/Q] [/1] [/4] [/B | /S] [/C]
FORMAT drive: [/Q] [/1] [/4] [/8] [/B | /S] [/C]

/V[:label] Specifies the volume label.
/Q Performs a quick format.
/F:size Specifies the size of the floppy disk to format (such as 160, 180, 320, 360, 720, 1.2, 1.44, 2.88).
/B Allocates space on the formatted disk for system files.
/S Copies system files to the formatted disk.
/T:tracks Specifies the number of tracks per disk side.
/N:sectors Specifies the number of sectors per track.
/1 Formats a single side of a floppy disk.
/4 Formats a 5.25-inch 360K floppy disk in a high-density drive.
/8 Formats eight sectors per track.
/C Tests clusters that are currently marked "bad."

9.8.12 FDISK

Configures a hard disk for use with MS-DOS.

FDISK [/STATUS] /X

/STATUS Displays partition information.
/X Ignores extended disk-access support. Use this switch if you receive disk access or stack overflow messages.

9.8.13 CHKDSK

Checks a disk and displays a status report.

CHKDSK [drive:][[path]filename] [/F] [/V]

[drive:][path] Specifies the drive and directory to check.
filename Specifies the file(s) to check for fragmentation.
/F Fixes errors on the disk.
/V Displays the full path and name of every file on the disk.

Type CHKDSK without parameters to check the current disk.

Instead of using CHKDSK, try using SCANDISK. SCANDISK can reliably detect and fix a much wider range of disk problems.

9.8.14 EDIT

MS-DOS Editor Version 2.0.026 Copyright (c) Microsoft Corp 1995.

EDIT [/B] [/H] [/R] [/S] [/

- /B - Forces monochrome mode.
- /H - Displays the maximum number of lines possible for your hardware.
- /R - Load file(s) in read-only mode.
- /S - Forces the use of short filenames.
- /- /? - Displays this help screen.
- [file] - Specifies initial files(s) to load. Wildcards and multiple filespecs can be given.

9.8.15 PING

sage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]

[-r count] [-s count] [[-j host-list] | [-k host-list]]

[-w timeout] destination-list

Options:

- t Ping the specified host until stopped
 - To see statistics and continue - type Control-Break;
 - To stop - type Control-C
- a Resolve addresses to hostnames
- n count Number of echo requests to send
- l size Send buffer size
- f Set Don't Fragment flag in packet
- i TTL Time To Live
- v TOS Type Of Service
- r count Record route for count hops
- s count Timestamp for count hops
- j host-list Loose source route along host-list
- k host-list Strict source route along host-list

-w timeout Timeout in milliseconds to wait for each reply

9.8.16 TYPE

Displays the contents of text files.

TYPE [drive:][path]filename

9.9 RESOURCES

Windows 98 Installation and Configuration Handbook, Rob Tidrow, Que Books

Windows 2000 Professional Resource Kit, Microsoft Press

Windows NT Workstation 4.0 Resource Kit, Microsoft Press

Inside MS-DOS 6.22 3rd Edition, Mark Minasi, Mark, New Riders Publishing

9.10 SUMMARY

In this chapter we discussed the differences between the FAT16, FAT32, and NTFS file systems. We also discussed the way the different file systems deal with long filenames and 8.3 names, and the best way to avoid problems with this conversion. Finally, we talked about internal and external DOS commands and the proper way to use them.

9.11 REVIEW QUESTIONS

1. What is the command used to format a hard drive, designated as the drive c:, and make the drive bootable.

2. What does chkdsk do?

3. List three advantages and three disadvantages of using the FAT16 file system.

4. What is the short filename of the following files, using NTFS.

Test File sample 1.doc _____

Test File sample 2.doc _____

Test File sample 3.doc _____

Test File sample 4.doc _____

Test File sample 5.doc _____

Test File sample 6.doc _____

5. What is the difference between internal and external DOS commands?

6. What file systems does the Windows NT 4.0 support?

7. Which operating systems support the FAT32 file system?

8. Which DOS commands copies entire files or directories with different options?

9. What is the command to list the files within a directory and only show a page full at a time?

10. What command do you use to delete an entire directory, its subdirectories, and all the files?