CYBER DIMENSIONS

Disk Management in the Information Age

BARBARA J. MINNICK
DOUGLAS K. SMITH
Indiana State University
Terre Haute, Indiana

ABSTRACT. Teaching *file* management concepts has become common, but important *disk* management concepts should also be taught. Teachers and students should understand (a) the reasons for using directories/folders and proper use of directories/folders, (b) how fragmentation occurs and how to correct it, (c) how to partition large hard drives to reduce the size of the allocation unit and recover wasted space, and (d) how surface defects occur and can be detected before they cause problems.

Teaching file management concepts such as giving files meaningful names, copying files, renaming files, moving files, and deleting files has become common, but some important disk management concepts should also be taught. Whether students are working in a DOS or Windows environment, the following four important disk management concepts should be taught:

- using directories/folders and the effect of a full file allocation table (FAT)
- eliminating disk fragmentation
- checking and reducing the size of the allocation unit on hard disks
- scanning a disk for surface errors

The Importance of Directories/Folders and How to Use Them

How many files do your students create or use in a semester? Do they save them to the root directory/folder of a floppy or one section of a hard drive allocated for their use, or do they organize them into directories/folders as hard copies are organized into folders before placing them in a file drawer? (Note: Windows 95 refers to directories as "folders" and uses a file folder as the icon, so the analogy between electronic files and paper files is now even more appropriate.)

Whether students are using floppy disks or a hard disk for storage, proper use of directories/folders is very important. The importance of organizing files into directories/folders can be illustrated by (a) demonstrating the effect of a full file allocation table (sometimes referred to as FAT) and (b) demonstrating the difficulty of locating a specific file when it is placed in the root directory/folder along with hundreds of other files or placed in poorly organized and named directories.

To demonstrate the effect of a full file allocation table using the traditional DOS file-naming conventions of a 1–8 character file name and a 1–3 character file extension, one can copy 112 *small* files to a low-density 3.5" floppy disk or 224 *small* files to a high-density 3.5" floppy disk. Then one can use the DOS DIR command, the Windows 3.1 File Manager, or Windows 95 Windows Explorer to illustrate that there is still a lot of blank space on the diskette. Try to copy another file to the diskette by using the DOS copy command, File Manager, or Windows Explorer. Also, try to save a file to the diskette from a software application such as WordPerfect or Lotus 1-2-3. The error message displayed depends on the method that is used to try to add the file:

- The DOS COPY command displays: "Cannot make directory entry, 0 file(s) copied."
- File Manager for Windows 3.1 displays: "File Manager cannot create or replace filename: Access denied. Make sure the disk is not full or write protected."
- Windows Explorer for Windows 95 displays: "Cannot create or replace file name: The directory or file cannot be created."
- WordPerfect 6 for DOS displays: "Access denied."
- Lotus 1-2-3 for Windows displays: "Not enough disk space on drive a: Delete one or more files to increase available space or use a different disk."

Note none of the error messages say the file allocation table is full, and some indicate the problem might be a write-protected or full disk. The DOS message hints at the problem by stating that it cannot make a directory entry (i.e., there is no room in the file allocation table).

Now, delete one file from the disk and create a directory/folder on the disk. The directory/folder name is placed in the file allocation table in place of the deleted file name, and additional files can now be placed on the disk by placing the files inside the directory/folder. A directory/folder entry occupies only one position in the file allocation table, and the only limit on the number of files placed in a directory/folder is the amount of free space on the disk.

There is a limit to the number of file and directory/folder entries in the root directory/folder of every disk: Using 1–8 character file names, a low-density floppy has a limit of 112 entries, a high-density floppy 224 entries, and most hard drives 512 entries. If the longer file names allowed by Windows 95 are used, the file allocation table will hold fewer entries and several files must be deleted to make room for a new directory/folder. However, once a directory/folder is created, space on the disk is the only limit to how many files can be placed inside the directory/folder.

To demonstrate the difficulty of locating poorly placed files, have students locate a file when the file name and the date that the file was created are not known. Give them some information about the contents of the file and part of the file name; for example, that it contains information about a software demonstration of bitmapped clip art and that the letters "DEM" will appear somewhere in the file name, or that DEM will be used for the extension of the file name. Ask the students to locate the file on a floppy disk that contains a lot of files in the root directory/folder and several directories/folders, such as 319CLASS, DESKTOP, WORKSHOP, FALL96, with additional files in each directory/folder. In the root directory/folders and in the subdirectories/subfolders, place several files containing DEM somewhere in the name; for example, 319DEM.DOC, DEM-F96. DOC, DTP.DEM, DEMO.WK4, DEMO. DOC, DEMO5.

The students will have to locate each file with DEM in the file name or extension, and then open the file to see if it contains a demonstration about different types of clip art. Make sure students have access to the software used to create each file and point out the problems caused by a nonstandard or missing file name extension. Locating the file in this way is a tedious process and can be used to stress the importance of (a) meaningful file names that are easy to remember; (b) search or find techniques in DOS, Windows, or specific applications (for example, the advanced option of the find command in Windows 95 lets users search all files, regardless of the application used to create them, for files containing specified text); and (c) the importance of organizing directories/folders on a disk in the same way that paper files are organized in a filing cabinet. For example, if all demonstration files were placed in one directory/folder named DEMO, the scope of the search would be much smaller. Also, characters in the file name no longer have to be used to identify the file as a demo file, and the 1–8 character file name can be much more meaningful—for example, CLIPART.DOC or BITMAP.DOC. Also, Windows 95 allows longer directory/folder and file names to be used, so it is easier to have meaningful file names, but only the first 6 characters, plus the symbol ~ and a sequential number, will be shown in DOS, DOS applications, or the older Windows 3.1 applications. Also the longer file names cannot be used on non-Windows networks.

After demonstrating the importance of directories/folders, make sure that the students use directories/folders as they complete their assignments. If using more than one software application, they can be required to use more than one level of directories/folders. For example, it may be appropriate to use a WORDFILE directory/folder for their word processing files and a SSFILE directory/folder for their spreadsheet files. Then students could create a directory/folder for each assignment inside the first directory/folder level; for example, WORDFILE\PROJECT1, WORDFILE\PROJECT2, SSFILE\PROJECT1, SSFILE\PROJECT2. Students should also be required to copy files between directories/folders, back up their files to a floppy, and remove directories/folders when they are finished with them.

Disk Fragmentation and How to Defragment Files

File fragmentation refers to a condition in which the files on a disk are broken into scattered locations on the disk. Free space fragmentation refers to the empty space on a disk being broken into scattered parts. Together the two types of fragmentation are referred to as disk fragmentation.

Disk fragmentation occurs when files are constantly being erased from a disk and new files copied or saved to the disk. For example, when several small files are erased from a disk and then a large file is saved to the disk, the large file will be divided into several pieces and placed in the allocation units freed by the erased files. Also, if a file is enlarged by adding more information to it, the extra information will probably be placed in a noncontiguous allocation unit. Fragmentation is a problem because it (a) takes longer for a computer to read from and write to fragmented files, so system performance is slowed down, (b) causes more wear and tear on the disk, and (c) creates opportunities for chaining errors or lost allocation units. Because of these effects of fragmentation, "an understanding of fragmentation, and the resources available to eliminate its threat, are vital to restoring and maintaining peak system performance" (Butler, 1995).

Teachers can demonstrate how fragmentation occurs by using a floppy disk that contains a lot of small and medium-size files in the root directory/folder. The files need to be in the root directory/folder so the DOS CHKDSK command that works only on the current directory/folder can be used to show students whether fragmentation is present and what files are fragmented. First, delete several of the small files, and then copy several medium-to-large files to the disk. The process of erasing and copying files might have to be repeated several times before the disk is highly fragmented. Next, use the DOS CHKDSK *.* command or one of the other tools mentioned later to check the disk for fragmentation. After demonstrating *how* fragmentation occurs, discuss *why* fragmentation is not desirable (see above discussion).

The final step is to remove the fragmentation. For a long time, the only method available was to back up the disk, reformat it, and restore the files, but software products are now available that can defragment disks. The best software works online in the background and requires no intervention by the user; however, teachers might not have such software in the classroom. Various software solutions are (a) using the DOS DEFRAG command if DOS 6.2 or higher is available, (b) using the Windows 95 disk defragmenter in the System Tools folder, (c) using a separate file management utility such as Norton or Diskeeper if such a utility is available, or as a last resource, (d) using the COPY or XCOPY command to copy all the files to a blank disk. (NOTE: DISKCOPY will not work because it will make an identical twin of the original disk, thus retaining the fragmentation.) The first three solutions will also check the amount of fragmentation on the disk, and the software will make a recommendation concerning the need to defragment the disk.

To use the DEFRAG command,

- close all programs that are running, including Windows,
- check for lost allocation units by typing the command: CHKDSK/f at the DOS prompt and answer "Y" if the system asks if lost chains should be converted to files,
- start the defragment program by typing DEFRAG at the DOS prompt,
- use the UP or DOWN arrow key to select the drive to defragment, and then press ENTER to start defragmenting.

If a user has DOS 6.2 or higher and wants more information about the defragment program, he or she can type HELP DEFRAG at the DOS command prompt.

To use the disk defragmenter in Windows 95, click on the Start button and select Programs, Accessories, System Tools, Disk Defragmenter, and then select the drive to defragment. If Microsoft Plus! has been added to Windows 95, the System Agent in the System Tools folder can be used to schedule the disk defragmenter and other maintenance tools to run at a specific time, or whenever the computer is on but not being used, thus automating the maintenance process ("Keeping Your PC Maintained," 1995).

To ensure that students understand the concept of fragmented files, request them to

- check their student disk for fragmented files after they have been saving to and deleting from the disk for several weeks (or they can be asked to check a demo disk supplied by the instructor), and
- defragment the files on the disk by using one of the methods discussed earlier.

Allocation Units and How to Reduce Their Size

An allocation unit is the basic unit of storage on a disk, and each allocation unit can contain information about only one file. When a file is saved to a disk, it uses one or more allocation units, depending on the size of the file and the size of the allocation unit on that disk. For example, a small file that contains only 55 bytes or characters is stored in a reasonably economical manner on a floppy disk, but a lot of space is wasted on a large hard drive.

The DOS CHKDSK command can be used to illustrate the size of allocation units on various disks:

- CHKDSK A: will show that a low-density 3.5" floppy has an allocation size of 1,024 bytes and contains 713 allocation units. Thus, if a one-character file is saved to the disk, it will use 1,024 bytes of storage space; and if a 1,025-character file is saved to the disk, it will use 2,048 bytes of storage space.
- CHKDSK A: will show that a high-density 3.5" floppy has an allocation size of only 512 bytes and contains 2,847 allocation units.
- CHKDSK C: will show the allocation size on the hard disk.

It is important that students understand that the size of the allocation unit on hard disks doubles every time the size of the hard drive doubles, as shown in the following data (Maxtor Technical Assistance, 1996; *PartitionMagic User Guide*, 1995):

- A 16–127 Mb disk uses 2,048 byte (2K) allocation units
- A 128–255 Mb disk uses 4,096 byte (4K) allocation units
- A 256–511 Mb disk uses 8,192 byte (8K) allocation units
- A 512–1,023 Mb disk uses 16,384 byte (16K) allocation units
- A 1,024–2,048 Mb disk uses 32,768 byte (32K) units

Thus, the larger the hard drive, the less efficiently it stores small files. For example, on a gigabyte hard drive, 1,000 files that are each 750 bytes will consume 1,000 allocation units, or 32,768 bytes times 1,000, for a total of 32,768,000 bytes—the capacity of 46 low-density or 23 high-density floppies. However, if directories/folders are used (remember the file allocation table limit), all 1,000 files can be saved

- on *two* low-density floppies with 426 unused allocation units, because each floppy disk has 713 allocation units, and each allocation unit will hold up to 1,024 bytes—or in this case, one of the 750-byte files with space to spare, or
- on *one* high-density floppy with 847 unused allocation units, because the high-density disk has 2,847 allocation units, and each 750-byte file will require two of the 512 byte units.

The effect of the allocation unit on storage of files can be demonstrated by

- using the DIR command on a disk to determine how much space is used and how much space is free,
- copying a very small file to the disk, and
- using the DIR command a second time. Note the space used will increase by the size of the copied file, but the free space will decrease by the size of the allocation unit used to store the file.

Because hard drives are getting larger, it is important to know about allocation units and their effect on the space required for files. In Table 1, I show how required space increases with the

TABLE 1. Sample Allocation Units: Effect on Disk Space

Disk size (megabytes)	Cluster size (kilobytes)	Disk space wasted (%)
16–127	2	2
128–255	4	4
256–511	8	10
512–1,023	16	25
1,024–2,048	32	40

size of allocation unit, as estimated by *PartitionMagic User Guide* (1995).

Hard drives can be partitioned to reduce the size of the allocation unit. For example, a gigabyte hard drive could be divided into two 500-megabyte drives, thus decreasing the size of the allocation unit from 32,768 bytes (32K) to 8,192 (8K) bytes. According to Goldberg (1996, p.41), "a good rule of thumb is to keep partition sizes to less than 256 megabytes if you're storing primarily small files or between 256 megabytes and 512 megabytes if you'll also be storing lots of large files."

One way to partition a hard disk is to back up everything on the hard drive, create a startup or boot disk, use the DOS Fdisk command to partition the disk, reformat the disk, and restore the backed-up data to the hard drive. Partitioning a disk in this way is a time-consuming process, but it can save a lot of disk space! Fortunately, software is available that will simplify the process. PartitionMagic by PowerQuest Corporation allows users to add, resize, or reposition disk partitions by dragging them from one location and dropping them at another (Piven, 1995).

Surface Disk Defects and How to Detect Them

When a hard disk or floppy disk is formatted, any surface defects are located and the cluster containing the defect is marked so it will not be used. However, as a disk is used, additional defects can result from normal wear, contaminants in the air, or by the drifting of the read/write drive heads so the heads are over areas of the disk not originally tested for defects. These unknown surface defects are disasters waiting to happen, as important files can be lost when the drive is no longer able to read the data. The ScanDisk command in DOS 6, the ScanDisk system tool of Windows 95, or various utility software such as Norton's, can scan the surface of a disk and detect and isolate problem areas before they cause read/write problems.

If using Windows 95, simply type scandisk a: or scandisk c: at the DOS prompt, or click on the Start button and then select Programs, Accessories, System Tools, ScanDisk, and use the dialog box to select the drive you want to scan and the type of scan.

Disks (especially hard drives) should be scanned for errors on a regular basis, such as once a week, and students need to know why such scans are important and how to do them.

Conclusion

Teaching disk management for the information age will ensure that students understand (a) the reasons for using, and proper use of, directories/folders; (b) how fragmentation occurs and how to correct it; (c) how to partition large hard drives to reduce the size of the allocation unit and recover wasted space; and (d) how surface defects occur and how to detect them before they cause problems. This knowledge of disk management will enhance the computer literacy of teachers and students, helping them to find files faster, perform preventative maintenance on hard drives, and cope better with problems that arise as they use computers in the classroom, at home, and on the job.

REFERENCES

Butler, H. (1995). Disk fragmentation: Emerging as a multiplatform issue. [Online]. Available: http://www.decus.org/decus/pubs/magazine/fal195/disk.html.

Goldberg, C. J. (1996). Tuning up. *Entrepreneur, 24,* 41–42.

Keeping your PC maintained with Microsoft Plus! (1995). *Microsoft Magazine, 2,* 23–25.

Maxtor Technical Assistance [email to B. Minnick on June 19, 1996, from technical_assistance@maxtor.com]

PartitionMagic User Guide. (1995) Orem, UT: PowerQuest Corporation.

Piven, J. (1995). New technology eases pains of hard drive partitioning. *Computer Technology Review, 15,* 34, 40+.