

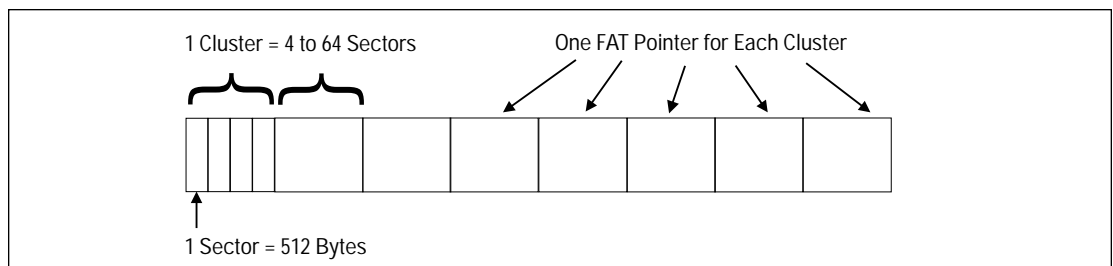
# The FAT Dilemma: Analysis and Solutions for Disk Storage Efficiency

Now that you've got that high capacity hard drive that you've always wanted, you may be wondering what happened to all the space you thought you had. It's not your hard drive. You've got the latest BIOS or BIOS translation utility. So what's wrong? It may be your FAT.

## What is a File Allocation Table (FAT)?

DOS, Windows 3.x and Windows 95 track files stored on the hard disk through directory structures and an index called a file allocation table (FAT). This table contains up to 65,520 entries and keeps track of which disk sectors are bad, in use, or available for assignment. With the advent of very high capacity hard drives, the limitations of FAT and its Windows 95 successor, VFAT, pose a dilemma for the efficient use of storage. Your disk drive could be losing many megabytes to FAT's built-in file inefficiencies.

Using 16-bit pointers, the FAT allocates file space on your hard disk in clusters. A **cluster** is a group of one or more contiguous sectors and is the smallest unit a file system can allocate. A **sector** is the smallest unit of disk space that the hardware can read or write and it is equal to 512 bytes. These sectors form a volume (an entire hard drive or a partition) that is organized into a file system.



There are normally 4, 8, 16, 32 or 64 sectors per cluster. Depending on the capacity of the hard drive, the smallest cluster sizes are 2K, the largest, 32K. For reasons of compatibility between DOS, Windows 3.x and Windows 95, the FAT file system has a limit of approximately 65,520 FAT entries. Therefore, as hard drive capacities have doubled in size, cluster sizes have doubled. The breakpoints in cluster sizes specified by the FDISK command are:

Partition Size*	Cluster Size	Sectors per Cluster
0 - 127 MB	2K	4
128 - 255 MB	4K	8
256 - 511 MB	8K	16
512 - 1023 MB	16K	32
1024 - 2048	32K	64

\* These numbers are reported in binary megabytes (1,048,576 bytes), not decimal megabytes (1,000,000 bytes). FDISK uses binary megabytes while hard drive capacities are frequently expressed in decimal megabytes.

### Some FAT Math:

16-bit FAT pointers =  $2^{16}$  = 65,536 pointer values minus those values reserved for special use = 65,520 (the maximum number of locations that can be specified by the FAT)

### The 2 GB Barrier

$65,520 \times 64 \text{ sectors per cluster} \times 512 \text{ bytes per sector} = 2,146,959,360 \text{ bytes per partition (2.1 GB)}$

While BIOS upgrades and translating software can break high capacity barriers, they cannot exceed the ultimate 2.1 GB per volume barrier imposed by the DOS/Windows-based FAT system nor the associated breakpoint cluster sizes and sectors per cluster.

Cluster size and sectors per cluster greatly influence overall storage efficiency. Because operating systems store a file on the hard disk using

# The FAT Dilemma

only whole clusters, disk space can be wasted when the file size is considerably smaller than the cluster size. Imagine a 2K file being stored in a 32K cluster. That's 30K of lost space. Hence, the larger the cluster size, the greater the likelihood of wasted disk space.

## Solution: Partition

So what is the solution? Increase the number of partitions? Certainly for drive capacities that exceed the 2.1 GB barrier, there is no other choice for DOS/Windows-based systems but to create partitions or virtual drives. The question then becomes - how many? While the FAT is most efficient when drive volumes are formatted for 2K cluster sizes (or 128 MB capacity), this would be impractical for a high capacity drive. The inefficiency of large clusters may be the trade-off for fewer but larger virtual disks.

Determine the average length of your work files and the amount of space needed by your applications. You should consider partitioning your 2 GB drive into 1 GB partitions, perhaps even smaller. Choose a reasonable number of volumes to minimize wasted space and remember these guidelines.

- The smaller the partition size, the smaller the cluster size, and, therefore, a more efficient use of disk space.
- Minimizing the size of clusters maximizes the number of clusters, which can slow file access and increase disk fragmentation.

### To estimate the amount of wasted disk space on your drive:

1. Type CHKDSK C: to generate a report of disk space usage and perform the following calculation.

	Total # of allocation units
<b>subtract:</b>	<u># of available allocation units</u>
<b>result:</b>	# of allocation units used

<b>then take:</b>	# of allocation units used
<b>multiply by:</b>	bytes in each allocation unit
<b>final result:</b>	# of bytes of space allocated

2. Type DIR C:\S/A to determine the cumulative length of all files on disk (except those with special attributes).
3. Take the "final result" from step 1 above and the "cumulative length" from step 2 and calculate the wasted space due to cluster size.

	# of bytes of space allocated
<b>subtract:</b>	<u>the cumulative length of files</u>
<b>result:</b>	cumulative cluster waste



WESTERN DIGITAL

WESTERN DIGITAL CORPORATION  
8105 IRVINE CENTER DRIVE  
IRVINE, CALIFORNIA 92718

TELEPHONE: 714.932.5000  
FOR SERVICE AND LITERATURE, CALL 800.832.4778  
OR 714.932.4900  
INTERNET: <http://www.wdc.com/>  
AMERICA ONLINE: WDC  
\*HARDCORE NETWORKS\* WDC