

SECTORS, CLUSTERS, AND LARGE HARD DRIVES

About a year ago, I quoted a prediction that hard drive prices would fall to 10 cents per megabyte in January 1996. At this writing (26Nov95), it is still not known if they will reach that mark, but the December 1995 issue of Computer Shopper carried an ad for a Fujitsu 1090-megabyte drive for $209, which works out to about 19 cents a meg. The point is, prices are dropping, and a lot of hams are upgrading to larger drives. With that upgrade comes a decision as to how to partition the drive. One big partition, or do you create several smaller ones (drives C:, D:, E:, and so on)? In some cases, the decision is made for you; if your computer has a standard BIOS, the largest single partition you can have is 504 megabytes.

Why? What is all this about clusters, sectors, heads and stuff, and why does it place limitations on the size of a C: partition? To answer this, let us begin by reviewing terminology. You probably already remember that a bit is like a single toggle switch - either on or off - and a byte is like a bank of 8 toggle switches, each one of which can be on or off. A single character, the letter A for example, requires a particular pattern of on or off positions in the bank of 8 toggle switches. The actual pattern is:

0 1 0 0 0 0 0 1

which turns out to be simultaneously both a binary number and a sort of "picture" of 8 switches and whether each is on or off. Starting from the right (because that's how binary numbers are read), the 1st and 7th switches are ON, while all the rest are off. We humans hate to work with binary notation because the numbers are so hard to read, so we convert to a different numbering system (hexadecimal). The hex representation of the letter A is 65h (the lower case h just tells that the number is in the hexadecimal system). Nevertheless, 65h is a number that can be converted to a binary number, and that binary number is a direct "picture" of 8 toggle switches, and whether each is on or off.

Now, it turns out that 512 bytes is a sort of magic number in the world of hard drives. It is the smallest "chunk" of information that the hardware can read from or write to the hard drive at one time. To help you relate to the size of this number, if you type a paragraph on a 3" X 5" card, leaving a nice white margin all around, the letters and spaces between them occupy about 512 characters and spaces. This index card worth of information that can be written to or read from a hard drive in one fell swoop is called a SECTOR.

So a sector is 512 bytes long, - by a hardware limitation. But there is also a software limitation. Whenever DOS creates a file, it must allocate a certain number of sectors to that file, as it creates it. This group of neighboring sectors (next to each other on the drive platter) is known as an ALLOCATION UNIT or a CLUSTER. If your hard drive is below 16 megabytes in size, a cluster consists of just one sector, 512 bytes. That means that, if you create a file named STAN with just this in it: WB9RQR (6 bytes long), it will occupy 512 bytes of disk space. The unused 506 bytes (sometimes called CLUSTER OVERHANG) are locked away; they are empty but already allocated to the file STAN and cannot be used for anything else unless the file is erased. On the other hand, if your hard drive is between 16 and 128 megabytes in size, each cluster occupies 2,048 bytes or 4 sectors. The file STAN, containing only 6 bytes, would have a cluster overhang of 2,042 bytes. Even larger hard drives have larger clusters, up to 64 kilobytes in size (128 sectors of 512 bytes each). Of course, cluster overhang increases proportionally. Cluster overhang will be the subject of a later article.

Why does cluster size vary? Because DOS has a filing system that keeps track of every file on your drive by giving each cluster a unique number. The filing system itself (File Allocation Table or FAT) has a size limit; it cannot exceed 65,536 bytes (64 kilobytes). So, if the FAT is limited in size, something else has to be larger on larger hard drives. The number that gets larger is the number of sectors per cluster; cluster size increases.

Well then, if there are limits on the FAT (64 kilobytes) and limits on the clusters (64 kilobytes), then there must be some limit on the physical tracks (cylinders), heads and sectors that a hard drive can hold. There are, and the result is that the following MAXIMUM values are in place in a standard BIOS for any hard drive:

CYLINDERS	HEADS	SECTORS
1,024	16	63

If you multiply 1,024 cylinders X 16 heads X 63 sectors X 512 bytes per sector, the result is 528,482,304 bytes or 504 megabytes. That's it, the magic number over which DOS cannot go. That is it. If you want a single C: partition on that new drive, it can only be 504 megabytes in size.

That is not to say you cannot partition that new drive which is 1,080 megabytes when formatted to a C: and D: drive. Note, however, that since each can only be 504 megabytes in size, a C: and D: will only occupy 1,008 megs; you will loose 1,080 - 1,008 = 72 megabytes. Perhaps C:, D: and E: partitions would be better, so you don't wind up with any unused space. Remember also, that if you have a CD ROM currently designated as drive D:, it will have to become drive F:, if you partition the new hard drive as C:, D: and E:.

Confusing, isn't it? You bet. On top of that, some hard drive manufacturers are adding another confusing factor. Remember that a kilobyte is 1,024 bytes, and a megabyte is 1,048,576 bytes? Therefore, 100 megabytes is 104,857,600 bytes. Well, suppose you are a hard drive manufacturer and you make a new model that holds exactly 100,000,000 bytes after it is formatted. Can you call it a 100-megabyte drive? No, absolutely not, since 100,000,000 is less than 95% of 104,857,600. However, it would sound better in advertising if it could be called a 100 megger. So some manufacturers have started advertising their drive's capacities using decimal notation (100 megabytes) instead of hexadecimal (95 megabytes), and it is difficult to tell which companies are being honest and which are not. Furthermore, some manufacturers are advertising drives based on the unformatted capacity, which can be a number considerably larger than the actual usable space on the drive. Use care in examining advertising claims. Be sure you know exactly what you are getting. You want to know what a drive will hold after formatting, and you want that number in accurate, hexadecimal notation. Happy computing!