About restoring erased files

When you erase a file using Explorer, Windows keeps a temporary copy of the file in the Recycle Bin that you can retrieve if you change your mind. However, Windows does not detect files erased or overwritten by applications running under Windows or erased from the DOS prompt.

Norton Utilities can help you get these files back. Norton Protection guards against losing the files the Recycle Bin doesn't protect. UnErase Wizard can even help you restore files not protected by Norton Utilities.

{button ,AL(`Introducing Norton Protection;Introducing UnErase Wizard;To retrieve an erased file',0,`',`')} More Info… Click here for more information.

To retrieve an erased file

1 Click here to open UnErase Wizard.

2 Choose what you want UnErase Wizard to look for:

- **Recently Deleted Files:** The 25 most recently deleted files UnErase Wizard knows about. It's best to try this first. If the file you want to retrieve is not listed, click Next to specify more search criteria. This option is available on Windows 95, Windows 98, and Windows Me.
- **All Protected Files:** Files in the Recycle Bin or under Norton Protection. If the file you want to retrieve is not listed, click Next to specify more search criteria.
- **Any Recoverable Files Matching Your Criteria:** Lets you specify precise search criteria to help UnErase Wizard locate the deleted file, or parts of the deleted file on the disk.
- **Find all Norton Protected Users files:** Searches for other users' protected files as well as your own. This option is available on Windows NT and Windows 2000.

3 When the file you want to recover is located, click the filename, and then click Recover.

4 Click Finish to close UnErase Wizard.

{button ,AL(`About restoring erased files',0,`',`')} More Info… Click here for more information.

About diagnosing and fixing problems

Trouble-free computing depends on the integrity of your computer, which requires an error-free hard disk and a correctly installed copy of Windows. Although most computers start out this way, over time Windows and hard drives are likely to degrade. This degradation, if not corrected, can ultimately lead to data loss.

Many Norton Utilities programs are on alert for the most common error conditions that can cause computer lock-ups and crashes, and in turn lead to computer problems and data loss. If a Norton Utilities program detects a potential problem, it can correct the condition automatically, or warn you, giving you the option to fix the problem if possible.

If you keep Norton System Doctor running all the time, it will spot problems early and recommend corrective action. Whenever you encounter a problem, use Norton Disk Doctor and Norton WinDoctor. When you feel like giving your system a comprehensive check-up, use Norton System Check.

{button ,AL(`To find and fix disk problems;To find and fix Windows problems;To find and fix computer problems',0,`',`')} More Info… Click here for more information.

To find and fix disk problems

Norton Disk Doctor scans your disks for problems, and offers to fix the problems it finds.

1 Click here to start Norton Disk Doctor.

2 If you want Norton Disk Doctor to automatically fix any errors it finds, click Automatically Fix Errors.

3 In the drive list, click the drive you want to diagnose.

You can select more than one drive to diagnose.

4 Click Diagnose.
Norton Disk Doctor checks the various components of your disk.

5 Follow the on-screen prompts as Norton Disk Doctor identifies and fixes any problems found on the disk.

{button ,AL(`Introducing Norton Disk Doctor;About diagnosing and fixing problems;To find and fix Windows problems',0,`',`')} More Info… Click here for more information.

To find and fix Windows problems

Norton WinDoctor scans your system for Windows problems, and offers to fix the problems it finds.

1 Click here to start Norton WinDoctor.

2 Follow the Wizard instructions to check for problems.

3 Click Finish.

Norton WinDoctor displays a list of the problems it finds.

4 You can choose to fix all or some of the problems:

- Click Repair All to automatically correct all of the problems in the list.
- Select a problem type and click Details to display all the problems of this type in the lower pane. Then, select one or more problems, and then click Repair All to automatically correct them.

To choose the solution you want applied to a problem type or a specific problem, click Repair and select Allow Me To Choose The Solution. Then select the solution to apply.

{button ,AL(`Introducing Norton WinDoctor;About diagnosing and fixing problems;To find and fix disk problems',0,`',`')} More Info… Click here for more information.

About speeding up your computer

Your computer's hard disk stores all of your files, applications, and the Windows operating system. Over time, the bits of information that make up your files get spread all over the disk. This is known as fragmentation, and is an inefficient way to store and retrieve information on a hard disk. The more you use your computer, the more fragmented it gets. Before long a fragmented hard disk can cause the entire system to slow down.

Speed Disk rearranges the bits of information on your disk for greatest efficiency.

{button ,AL(`To speed up your hard disk;To speed up your system using Norton Optimization Wizard',0,`',`')} More Info… Click here for more information.

To speed up your hard disk using Speed Disk

1. Start Speed Disk.
2. Click Start Optimizing to accept the recommendation.
   Speed Disk optimizes your disk drive.

By default, Speed Disk optimizes your primary hard disk drive, usually drive C. To optimize another drive, click its associated check box, and then click Start.

{button ,AL(`About speeding up your computer;To speed up your system using Norton Optimization Wizard',0,`',`')} More Info… Click here for more information.

About keeping software up-to-date

Some of the most basic components of your computer—your sound card, video card, and CD- ROM drive

—need the correct software to run properly.
Hardware manufacturers usually supply the necessary software with the hardware components they make. However, hardware manufacturers are constantly upgrading their software to make your hardware run even better.
The same is true for the programs you use. Software publishers often make improvements, sometimes called "patches," to your favorite programs and offer them to registered users at no charge. So many patches are released every week that keeping up with them can prove nearly impossible. Yet, without them you could be missing important improvements or crash-preventing bug fixes.

LiveUpdate technology included with Norton Utilities can keep Norton Utilities and your virus definitions files files up-to-date.

{button ,AL(`About LiveUpdate;To update your installed software',0,`',`')} More Info… Click here for more information.

To update your installed software

1 Click LiveUpdate at the top of the Norton Utilities main window.

2 Follow the on-screen instructions.

Note: Updating Norton Utilities or virus definitions files requires an Internet connection.

{button ,AL(`About LiveUpdate;About keeping software up-to-date',0,`',`')} More Info… Click here for more information.

Introducing Norton Utilities

Norton Utilities is an integrated suite of programs that can help you find and fix computer problems, improve your computer's performance, perform preventative maintenance tasks, and troubleshoot your system.

Norton Utilities keeps your computer in good condition, warns you of potential problems before they become apparent, minimizes your down time when something goes wrong, helps you to recover lost data, and provides the information you need when installing new hardware or software.

To access any of the Norton Utilities programs, double-click the Norton Utilities Integrator shortcut on your desktop and select the program to run.

{button ,JI(`nu.HLP>task',`NU_List_and_description_of_each_Norton_utility')} See a list and description of each Norton Utilities program

{button ,JI(`nu.HLP>task',`NU_Important_tasks_you_can_perform_with_Norton_Utilities')} See a list of important tasks you can perform with Norton Utilities

Important tasks you can perform with Norton Utilities

The list below shows the most important tasks Norton Utilities helps you perform. Click a task to get more details:

{button ,PI(`',`NU_System_Check_pop_up')} Giving your system a checkup
{button ,PI(`',`NU_Recovering_deleted_files_pop_up')} Restoring erased files
{button ,PI(`',`NU_diagnose_pop_up')} Diagnosing and fixing problems
{button ,PI(`',`NU_speeding_up_your_system_pop_up')} Speeding up your computer
{button ,PI(`',`NU_update_pop_up')} Keeping software up-to-date
{button ,PI(`',`NU_Getting_information_about_your_system_pop_up')} Learning about your system

---

{button ,AL(`About diagnosing and fixing problems;About keeping software up-to-date;About preparing your system against crashes;About restoring erased files;About speeding up your computer;About testing your computer hardware',0,`',`')} More Info… Click here for more information.

Norton Protection guards against losing the files the Recycle Bin doesn't protect. UnErase Wizard can help you restore erased files, even those that weren’t protected.

Norton Utilities helps speed up your system. Speed Disk defragments the files on your disk for greatest efficiency. Norton System Doctor monitors disk space with the Disk Space and Disk Slack Space sensors, and keeps an eye on fragmentation for you with the Disk Optimization sensor.

Norton Utilities is on alert for error conditions that lead to computer problems and data loss. If a problem is detected, Norton Utilities can correct the condition automatically, or warn you, giving you the option to fix the problem. Keep Norton System Doctor running all the time to monitor the health of your system. Whenever you encounter a problem, or want to give your system a quick check-up, use Norton Disk Doctor and Norton WinDoctor.

Norton Utilities can help you download and install updates to your software and hardware, using your Internet connection. LiveUpdate can help keep Norton Utilities and your virus definitions files up-to-date.

Norton Utilities provides you with detailed information about your system so you can make informed decisions. To review technical details about your system’s configuration, use System Information. To get information about your system's current conditions, use Norton System Doctor.

A list and description of each Norton Utilities program

Norton Utilities is a suite of programs designed to prevent and fix computer problems, optimize computer performance, and help you recover from disasters.

Click a Norton Utilities program in the list below to see a brief description:

{button ,PI(`',`NU_LiveUpdate_pop_up_description')} LiveUpdate	{button ,PI(`',`NU_Norton_System_Doctor_pop_up_description')} Norton System Doctor
{button ,PI(`',`NU_Norton_Disk_Doctor_pop_up_description')} Norton Disk Doctor	{button ,PI(`',`NU_WinDoctor_pop_up_description')} Norton WinDoctor
{button ,PI(`',`NU_Norton_Protection_pop_up_description')} Norton Protection	{button ,PI(`',`NU_System_Information_pop_up_description')} System Information
{button ,PI(`',`NU_Speed_Disk_pop_up_description')} Norton Speed Disk	{button ,PI(`',`NU_UnErase_Wizard_pop_up_description')} UnErase Wizard

---

{button ,AL(`Introducing Norton System Doctor;Introducing Rescue Disk;Introducing Speed Disk;About SpeedStart;About LiveUpdate;Introducing System Information;Introducing UnErase Wizard;Introducing Image;Introducing Norton Optimization Wizard;Introducing Norton Protection;Introducing Norton Registry Editor;Introducing Norton Registry Tracker;Introducing Norton WinDoctor;Introducing Norton Diagnostics;Introducing Disk Editor;Introducing Norton File Compare;Introducing Norton System Check;Introducing Norton WipeInfo;NUCD Intro',0,`',`')} More Info… Click here for more information.

LiveUpdate makes it easy to keep Norton Utilities and your virus definitions up-to-date by using your computer's modem or Internet connection to download updates directly from Symantec.

Diagnoses and repairs a variety of disk problems. It performs several tests, checking everything from the disk's partition table to its physical surface.

Enhances the Windows Recycle Bin, protecting deleted files. UnErase Wizard lets you recover the protected files.

Continuously monitors your system in the background to keep it free of problems and running at peak efficiency. Its sensors can alert you immediately when conditions require attention, recommend corrective actions, and even fix many problems automatically, without interrupting your work.

Optimizes your drives by defragmenting the files and free space on your disks to make file storage and retrieval more efficient.

Gives you detailed information about your computer, the peripherals—keyboard, mouse, printers, and multimedia devices

—attached to your computer, and your computer's Internet and network connections.

Diagnoses and repairs the most common types of Windows problems, keeping Windows running at peak efficiency. By fixing errors that might otherwise lead to data loss, Norton WinDoctor also protects your work.

Helps you recover deleted files that are protected by Norton Protection, the standard Windows Recycle Bin, or Novell's Salvage (for recovery of files on a network).

Could not open program

The program is not installed or has been moved from the folder to which it was installed.

To properly install the missing program, run Setup again.

The physical organization of your computer

A computer consists of:

- Central processing unit (CPU), which contains the logic necessary to make decisions and do arithmetic.
- Memory or primary storage devices, which hold information for the computer to process.
- Input/output (I/O) or peripheral devices from which the computer receives and to which it sends information. Some peripheral devices, such as monitors, display information. Others, such as keyboards, produce information. Still others are secondary storage devices, capable of accepting data from and transferring data to the computer.

All disk drives store data as groups of bits, the smallest pieces of information a computer recognizes. A bit is like a light bulb: it's either on or off. When a bit is on, it's considered to be set, or high, and its value is 1. When a bit is off, it's considered to be reset, or low, and its value is 0. A group of 8 bits constitutes a byte.

How the various disk drives and other memory media store these bits and bytes is one of the factors differentiating them. Some are magnetic devices, such as hard disk drives, floppy disk drives, and magnetic tape drives, while others are optical devices, such as CD-ROM drives.

Disk drives are also categorized by their physical characteristics. For example, some have removable disks and others contain non-removable disks. CDs, floppy disks, tape cartridge transports, SyQuest cartridge disks, optical disks used in write-once-read-many (WORM) drives, and Iomega Zip drive cartridge disks are all removable. The majority of hard disk drives used in PCs have non-removable, or fixed, disks. Non-removable doesn't mean the disk drive can't be removed from the PC, but rather that the disk itself can't be removed from the disk drive.

{button ,AL(`About CD-ROM drives;About CPU registers;About floppy disk drives;About hard disk drives;The memory components of your computer',0,`',`')} More Info… Click here for more information.

About hard disk drives

A hard disk drive is a fast, high-capacity storage unit that completely encloses a stack of platters in a protective enclosure. It is rigid, circular, and spins and is coated with a magnetic material.

The rigidity of the platters gives the hard disk drive its name. Each platter is attached to a center post called the spindle, which in turn is attached to a motor. The motor spins the platters at a constant speed. Common speeds are 3600, 4500, 5400, and 7400 rpm.

Attached to the comb are the heads, which read and write data from each side of each platter. The heads never touch the surface of the disk during operation. Instead, they ride on a cushion of air that not even a hair could squeeze through. The hard disk is hermetically sealed because even the tiniest dust particle caught between the head and the disk surface could damage the disk and cause data loss by scraping off magnetic particles.

After the hard disk is formatted, each side of each platter has logical concentric circles called tracks. All tracks residing the same distance from the spindle have the same track number and form what is called a cylinder. For example, taken together, all track 7s form a cylindrical ring around the spindle.

Cylinders, tracks, and read/write heads are numbered for reference. The outermost track is called track 0 and all track 0s reside in cylinder 0. Head 0 is the read/write head for the top surface of the top platter.

When the disk is not spinning, the cushion of air disappears and the head touches the disk. To help prevent damage, modern hard disks park themselves automatically when the computer is shut off by moving the heads to a safe cylinder known as the landing zone cylinder.

About sectors

Each track is divided into a number of arc-shaped segments, called sectors. The amount of data in a sector is the smallest unit read from or written to a magnetic disk.

Each track has the same number of sectors, but the length of the sectors varies from one track to another; the sectors of the outer tracks are physically longer than the sectors of the inner tracks.

Each sector contains:

- **Sector header**
Identifies the sector using a unique head, cylinder, and physical sector number. It also indicates if the sector is usable and, if not, may indicate another sector to use in its place. The last portion of the sector header contains a cyclic redundancy check (CRC) value, used to ensure the sector header is correct.
- **Gap**
Provides the read/write head the time it needs to switch from reading the sector header to writing in the data area. When reading data, the read/write head passes over the gap.
- **Data**
Contains 512 bytes of data when formatted under Windows or DOS. At the end of this area are error correction codes (ECCs), used to detect and correct errors.
- **Inter-sector gap**
Provides the read/write head the time it needs to switch from writing data to reading the next sector header. When reading data, the read/write head passes over the inter-sector gap.

Reading and writing

When the hard disk drive is accessed, the comb receives an electric signal from the drive's circuit board and moves the read/write head to the correct track. The computer writes or reads all the data from an entire cylinder without moving the heads again. The computer moves through the tracks in a cylinder by switching from head to head.

To write data, an electrical current is sent through a tiny coil of wire wrapped around a small iron core in the read/write head. The combination of the electrical current and the coil creates a magnetic field, which in turn magnetizes particles in the disk's coating. Each particle assumes one of the two possible positions and represents either a 1 or 0 bit. (This makes it easy to understand why exposing magnetic disks to large amounts of magnetism can disrupt the information stored on them.)

Reading data from a disk reverses this process. The magnetic particles on the disk surface induce currents in the read/write head's magnetic coil. The circuit board translates the changes in current back into electronic bits.

Because the sectors vary in size but contain the same number of bits, the bits on outer tracks are wider than those on inner tracks. The varying widths and the early designs for heads forced some drive manufacturers to compensate electronically. At an arbitrary spot on the disk, they changed the way in which signals from the head were interpreted or written, called prewrite compensation.

Even today, the longer tracks have wasted space. Some hard disks use a method called zone mapping to place more sectors on outer tracks than inner tracks. Other hard disks have a single platter that allows the circuit board the ability to emulate the presence of multiple platters and therefore more data space.

Various methods (or algorithms) have been developed for encoding data on the magnetic surface of a disk. They differ in capacity and in the reliability with which data is retrieved. The method is built into the disk drive and can't be changed. Two examples are Modified Frequency Modulation (MFM) and Run Length Limited (RLL). Hard disks using the RLL method pack more bits into less space than those using the MFM method. There are also variations of RLL, namely Advanced RLL (ARLL) and Enhanced RLL (ERLL).

{button ,AL(`About hard disk drives',0,`',`')} More Info… Click here for more information.

About floppy disk drives

The most widely used removable data-storage medium is the floppy disk. A floppy disk is a convenient place to save data so it can be moved from one computer to another. Floppy disks come in two sizes: 3½ inches and 5¼ inches.

The 3½-inch floppy disk comes in capacities that range from the low-density 720K disk to the extra high-density 2.88 MB disk. The capacity most commonly used is high density, 1.44 MB. The hard outer case and a metal (or hard plastic) piece, called the shutter, help to protect the disk, making it more reliable than the older 5¼-inch floppy disks. The shutter protects the cookie, the magnetically coated surface where data is stored. Although the cookie is not rigid, it is analogous to a platter in a hard disk.

Placing a 3½-inch floppy disk into the drive cocks a system of springs and levers, which slide the disk's shutter open to expose the cookie and bring the read/write heads into contact with it. Most of the floppy disk drives in use today have two read/write heads, one for the top surface of the disk and the other for the bottom.

When the floppy disk drive receives a request to read or write information, two motors are activated. The first motor, called the spindle motor, is located on the bottom of the drive. It rotates the cookie at approximately 300 rpm to pass data by the read/write heads. The second motor, a stepper motor (a common type of motor that moves in small steps), moves the read/write heads in and out to the proper location on the disk. When the read/write heads are in position and the cookie is spinning, the floppy disk drive executes read and write operations the same way a hard disk does.

Holes in the floppy disk case help (or hinder) the read and write processes. A small, square hole in the corner of a 3½-inch floppy disk case identifies it as high density. In addition, the words "High Density" (or "HD") appear on the case. In the other corner of the case, another small, square hole (with a write-protect tab that is moved to expose or hide the hole) determines whether the disk is read-only. When this hole is exposed, the drive identifies the floppy disk as read-only. This protects the data on the disk from changes and erasures because the computer can't write to it. For a 5¼-inch floppy disk, covering the write-protect notch at the edge of one side makes the disk read-only.

Some floppy disk drives use micro switches to detect information about the inserted disk. Others use an optical system that detects whether a beam of light passes through an open hole or gets blocked by a closed hole.

{button ,AL(`About hard disk drives',0,`',`')} More Info… Click here for more information.

About CD-ROM drives

CD-ROM drives are a variation of the devices used to play audio CDs. ROM means read-only memory. However, writeable CD technology also is available. Due to the large capacity of CDs (more than 600 MB), the CD is the perfect medium for distributing large software products, especially multimedia products. The most important difference between the CD-ROM drive and magnetic disk drives is the CD-ROM drive uses light, instead of magnetic fields, to read information from the CD surface.

The photodetector is the most important component in the CD-ROM drive. It contains the electronics necessary to interpret the CD surface as a string of bits. Within the detector is an electronic device called a laser diode. The diode emits a laser beam that travels through a series of lenses and mirrors to focus the beam on the CD surface. The laser beam is reflected back to the detector after hitting the CD surface. The optics within the photodetector redirect the reflected beam to a photodiode, which detects the amount of reflected light.

The CD surface (shiny aluminum film or foil protected by a plastic coating) contains a series of microscopic lands (flat areas) and pits. A land reflects the laser light, indicating a high (1) bit, whereas a pit disperses the light, indicating a low (0) bit.

Originally, CD-ROM drives rotated at the same rate as do audio CD players. To help speed up the process of reading data, manufacturers have developed CD-ROM drives that spin at higher rates. However, even the fastest of these drives can't yet read data as fast as a hard disk drive.

Unlike floppy disk or hard disk drives, CDs have a fixed physical sector size. They also have one spiral track. To read data, the CD-ROM drive must make the bits pass the detector at the same speed no matter where the detector is positioned. Because the bits at the center of the rotating disk are moving at a slower speed than those at the outer edge, the CD must spin faster when the detector is close to the center of the CD.

{button ,AL(`About floppy disk drives;About hard disk drives',0,`',`')} More Info… Click here for more information.

About disk controllers

A disk controller controls the operations of a disk. It enables the CPU and disks to communicate with each other. Various interfaces have been developed to define the rules for this communication. The following are interface types commonly used between disk controllers and hard disks:

- ST-506/412
- Enhanced Small Device Interface (ESDI)
- Integrated Drive Electronics (IDE))
- Small Computer System Interface (SCSI)
- Extended Integrated Drive Electronics (EIDE)

Of these interfaces, only ST-506/412 and ESDI are strictly oriented toward hard disks and require specialized controllers.

These interfaces vary not only in the amount of data they access, but also in their speeds. For example, ST-506/412 transfers about 5 to 7.5 megabits of data per second, while EIDE can transfer as much as 16.6 megabits per second.

SCSI drives are the fastest type of drive because SCSI controllers (or host adapters) have their own CPU for handling data transfers and other device related tasks, and consequently do not require resources from the main system CPU. Your system runs faster because your main CPU is free to do other chores. (This also explains why SCSI host adapters cost more than other kinds of drive controllers.)

Additionally, SCSI drives do not have drive wrap or sector translation problems (which are still a problem for EIDE drives).

Floppy disk drives use the slow disk controller interface they have used since their inception. They have a maximum transfer rate of 500 kilobits per second (kps), but rates of 350 kps are more typical.

CD-ROM drives can use SCSI, EIDE, or proprietary interfaces. The adapter cards for many CD-ROM drives use a subset of the SCSI standard that limits the card to a single device.

About speed

A number of factors that affect the speed at which data is accessed depend on the physical design of a disk drive and can't be modified. They include:

- Access time (which is a combination of seek time and latency)
- Transfer rate
- Rotation speed (which affects both the drive's latency and transfer rate)

Hard disk drives are significantly faster than floppy disk drives or CD-ROM drives. The speed of floppy disk drives has changed little over the years, but the speed of CD-ROM drives has been increasing.

- Hard disk drives have an access time of 8 to 15 milliseconds and a transfer rate of 5 to 16.6 MB per second.
- Floppy disk drives have an access time of 80 to 120 milliseconds and a transfer rate of .35 to .5 MB per second.
- CD-ROM drives have an access time of 160 to 300 milliseconds and a transfer rate of .15 MB per second for a single speed CD-ROM drive to .9 MB per second for 6x CD-ROM drives.

Access time

Access time is the amount of time it takes to locate information. For a magnetic disk drive, access time is the time needed to move the read/write head from the track it is on to the track where new data is to be read or written (seek time) plus the time needed for the requested sector to pass under the head (latency). For a CD-ROM drive, access time is the time needed to move the laser beam to the proper position along the CD's spiraling track plus the time for the sector to pass under the beam.

Since the head or laser beam can be anywhere in relation to the destination, the average seek time is used to rate drive performance. Most hard disk drives have an average seek time of less than 15 milliseconds, whereas even fast CD-ROM drives have average seek times 10 to 15 times longer.

The maximum latency for a magnetic disk is the time it takes for a complete rotation of a disk, since it's possible to miss a sector by the smallest fraction of a second. The average latency of a magnetic disk is exactly half the time of a complete rotation, assuming the sectors are requested at random.

The interleaving of sectors can reduce latency. For example, if a hard disk drive is reading sectors 1 through 4 in order, it may have less of a wait between sector reads if the sectors are not back-to-back on the disk. If the sectors are numbered 1, 3, 2, 4 instead of 1, 2, 3, 4, a slower computer waits a partial rather than a full revolution before reaching the next sector. For best performance, the interleave should carefully match the speed characteristics of the computer with those of the hard drive. Interleaving is performed during low-level formatting.

Many newer hard drives have their own memory buffers, where they can temporarily store an entire track or more of data. By allowing a drive to read the entire track at once, memory buffers eliminate latency. (Some CD-ROM drives also have memory buffers, enabling storage of as much as a megabyte of data. A memory buffer can substantially improve the performance of a CD-ROM drive.)

{button ,AL(`About CD-ROM drives;About hard disk drives;Low-level formatting',0,`',`')} More Info… Click here for more information.

Transfer rate

The transfer rate measures how fast data is transferred between the drive and computer memory in millions of bytes per second.

In today's more advanced hard disk drives, like SCSI and EIDE, transfer rates can approach or exceed 15 MB per second. The transfer rate for CD-ROM drives depends on the rotation speed as well as the use of data buffering.

{button ,AL(`About CD-ROM drives;About hard disk drives',0,`',`')} More Info… Click here for more information.

Rotation speed

The rotation speed affects both latency and the transfer rate. The higher the rotation speed, the faster the drive can access data. A hard disk's platters spin at a constant rate, commonly 5400, 7200, or 10,000 rpm. CDs spin at a variable speed, based on the position of the laser beam in relation to the edge of the CD. The range of speeds for a CD-ROM drive depends upon the model type: single, double (2x), triple (3x), quadruple (4x), or 6x speed, for example. Floppy disks rotate at about 300 rpm.

{button ,AL(`About CD-ROM drives;About floppy disk drives;About hard disk drives',0,`',`')} More Info… Click here for more information.

The logical organization of your computer

The operating system organizes and finds data on any disk drive. On a magnetic disk, reading and writing data requires a high degree of interaction with the computer's operating system. Because a CD is read-only, the operating system does not need to update information about what data is stored and where it is stored.

For speed and efficiency in accessing particular bytes on magnetic disks, the operating system constructs directories and indexes describing what's occupied, what's free, and what should never be used due to physical damage. This type of disk information is called the logical formatting.

To refer to a particular location on a disk, the disk controller uses the side or head number, track number, and sector number, so the PC must specify disk locations in this same manner. However, it is awkward for an operating system to communicate in the language the disk controller understands. For one thing, the number of tracks, sectors, and sides all change from one disk to another.

The operating system therefore locates data by using a sequential numbering scheme that enables it to maintain status information about every part of a disk. To reduce the overhead that would be required to monitor a disk at the sector level, the upper layers of the operating system deal instead with multi-sector units called clusters. The number of sectors in a cluster depends upon the size of the drive and is determined when the disk is formatted. (Clusters are also referred to as allocation units.)

The operating system organizes each logical disk into two main areas: the system area and the data area. The system area includes the boot sector, the file allocation table (FAT), and the root directory. (On FAT32 drives, the root directory is a cluster chain in the cluster heap, and the starting cluster of the chain is stored in the boot record.) The data area stores the files and folders.

{button ,AL(`About CD-ROM drives;About disk controllers;About sectors;Reading and writing',0,`',`')} More Info…

Click here for more information.

About partitioning

One organizational scheme the operating system recognizes is the division of a hard disk into multiple partitions. You can only partition hard disks.

A partition can be primary (capable of starting an operating system) or secondary (not capable of starting an operating system). You can have a maximum of four primary partitions and several secondary ones. Secondary partitions can be further subdivided into logical disks (also called volumes). Logical disks can give the impression that a single hard disk is actually multiple disks. For example, a hard disk might have three logical disks, typically referred to as drives C:, D:, and E:. In this case, C: may be the primary partition and store system data, while the logical disks D: and E:, on the secondary partition, store applications and data respectively. This makes it easy to backup only your data. Another example is a computer running more than one operating system. You may have two primary partitions, so you can run DOS and another operating system, such as Windows NT or OS/2.

Another reason to have multiple partitions on a hard disk is to reduce slack space on the disk. Because a drive can have a maximum of 64K clusters (at least until the arrival of FAT32, the larger the partition, the larger the cluster size. Repartitioning your disk into multiple smaller drives lets you use a smaller cluster size, which means less wasted disk space.

DOS supplies a program called FDISK that lets you create, view, alter, and delete partitions on your hard disk. It can also reserve space for partitions to be used by other operating systems. FDISK creates the partition table, which is the part of the master boot record that contains information about the partitions. The master boot record is stored at side 0, cylinder 0, sector 1 of every hard disk.

{button ,AL(`About FAT32',0,`',`')} More Info… Click here for more information.

Partition table

The partition table stores the following information:

- Type of operating system
- Starting location (head, sector, and cylinder)
- Size (number of sectors in the partition)
- Whether the partition is bootable
- Ending location (head, sector, and cylinder)
- Location of the other partitions

For the partition table to be valid, each partition must have a unique starting point and not overlap another partition. Additionally, only one partition at a time is marked as the active partition, the partition from which the operating system will be loaded. The program stored in the master boot record checks the partition table for the active partition and loads and executes the bootstrap loader in that partition's boot record.

{button ,AL(`Master boot record',0,`',`')} More Info…Click here for more information.

Master boot record

Although each logical disk has a boot record of its own, there is only one master boot record per hard disk. When starting up from a hard disk, the ROM BIOS bootstrap loads and executes the bootstrap loader in the master boot record before transferring control to a partition's boot record. The reason for this step is that the master boot record specifies which partition is the active partition. Not every partition may contain operating system files, so the system must determine the partition that does.

Boot managers alter the master boot record so that, instead of starting the system automatically, a menu is presented that lets you choose where the system starts from (for example, from a Windows 98 or Windows 2000 partition).

Note: If you use any drivers that load before DOS (such as Drive Space and Disk Manager, which alter the master boot record to load themselves), these drivers need to be on your bootable floppy disk (for instance, the floppy in your rescue disk set).

{button ,AL(`About boot sectors and boot records',0,`',`')} More Info… Click here for more information.

About boot sectors and boot records

The boot sector is the special sector at the beginning of a logical disk where the boot record is stored. The boot record on FAT32 drives requires three boot sectors, plus a copy of these sectors and a certain number of reserved sectors (usually 32) before the FATs begin. This reserved area (larger than that on FAT16 drives) allows copies of the boot record to be stored there, along with a sector that contains a free-space count and other file system information.

Every boot record contains information that helps the operating system determine how data is organized on the logical disk. The boot record includes the following information:

- Number of bytes per sector
- Number of sectors per cluster
- Number of sectors on the disk
- Number of sectors per track
- Number of heads on the disk
- Media descriptor byte, which indicates the type of the disk (such as a hard disk or a 3½-inch floppy disk with 1.44 MB capacity)
- File information, such as the number of sectors per FAT entry and the number of entries in the root directory

{button ,AL(`About FAT32;About floppy disk drives;About hard disk drives',0,`',`')} More Info… Click here for more information.

About formatting

Tracks and sectors are not physical structures you see when you look at the disk surface. Initially, a magnetic disk's surface is a randomly organized mass of scrambled iron oxide molecules. Preparing a magnetic disk for use involves the following processes that place tracks, sectors, partitions, boot records, and file allocation tables (FATs) on that surface.

- Low-level formatting prepares the disk surface.
- Partitioning creates the partitions and logical disks for a hard disk.
- High-level formatting prepares a logical disk for use.

When you format a floppy disk, both the low- and high-level formatting seems to occur as a single operation.

{button ,AL(`About boot sectors and boot records;About file allocation tables (FAT);About sectors;Partition table',0,`',`')} More Info… Click here for more information.

Low-level formatting

Low-level formatting (also known as physical or hard formatting) organizes the disk's surface into tracks and sectors. During formatting, sector identifiers are written to each track. This process establishes what is known as sector addressing, used to find each sector within a track. The sector addressing information is part of the sector header, which is one portion of a sector.

Low-level formatting is a process hard disk manufacturers now perform themselves. (However, you may be able to perform a physical format of an MFM or RLL drive, depending on its controller.)

High-level formatting

Once a disk has had a low-level format and been partitioned, the high-level format (also known as the logical or soft format) can be performed. High-level formatting places the structures on the disk that the operating system requires to access that disk: the boot record, FAT, and root directory. Each logical disk includes a boot record, whether or not the system boots from that disk.

These structures are set up either by using the DOS FORMAT command or by choosing FORMAT from a drive object's context menu within Windows. The DOS FORMAT command is still important when preparing a new or destroyed hard disk for use, since in this case Windows is probably unavailable.

The DOS FORMAT command also performs one other process, called bad sector mapping, which is time-consuming but important to data integrity. This process checks for bad sectors by reading the header of every sector. If a sector is bad, the cluster in which it resides is marked as bad in the FAT so the operating system won't use that cluster. (Windows 95, Windows 98 and Windows Me perform bad sector mapping through ScanDisk.)

File Allocation Table (FAT)

The file allocation table (FAT) is an integral part of many file operations. You can't create, read, or append data to a file without dealing with the FAT.

The FAT is used to track the locations of both files and free space on a logical disk. There are two copies of the FAT and they always follow the boot record on a disk. If one FAT becomes corrupt it can render an entire disk unusable, so, the other copy is used.

The FAT is a table that contains an entry for each cluster on a disk. The first FAT entry is a value representing the media descriptor byte, which is a copy of the one in the boot record. The second entry is reserved for special information. The rest of the entries correspond to the clusters on the disk. These entries contain one of four types of values to indicate the cluster's status:

- Cluster is available.
- Cluster is the last one in a file.
- If there is more than one cluster for the file, the value specifies the next cluster number for the file.
- Cluster is damaged.

When a FAT entry indicates a file has multiple clusters, a FAT chain is created for the file. Each FAT entry points to the next cluster of the file. It's this chaining of one FAT entry to another that's so vital to retrieving files.

(On FAT32, the upper 4 bits are reserved for future use by Microsoft.)

{button ,AL(`About FAT32;Data retrieval',0,`',`')} More Info… Click here for more information.

About FAT32

FAT32 is a file system from Microsoft that supports large hard drives and uses disk space more efficiently (especially on large disks) than earlier file systems.

The file allocation table (FAT) file system was originally created to store data on floppy disks (FAT12), and was later changed to allow it to handle data on fixed disks such as hard disks (FAT16). However, the FAT16 file system cannot support extremely large hard disks anything over 2 GB in size

so FAT32 was created to support these larger hard drives (up to 2 terabytes) and to provide more efficient use of disk space.

Disk space is allocated in clusters. That means that even if a file is smaller than a cluster, it still takes up a whole cluster, wasting any leftover space in the cluster. Or, suppose a file is large enough to fill 3-1/2 clusters, the file will still be allocated to four clusters. Since FAT32 drives use a smaller cluster size, typically FAT32 drives have less wasted (slack) space on the drive.

In addition, FAT32 systems can relocate the root directory and use the backup copy of the FAT instead of the default copy. Also, the boot record on FAT32 drives now backs up critical data, meaning FAT32 drives are less likely to fail. Since the root directory on a FAT32 drive is a cluster chain (instead of being a physical area on the disk), it can contain more entries than a FAT16 drive (up to 65,536) and it can be located anywhere on the drive.

About the root directory

The root directory is the table between the second copy of the FAT and the actual data area on the logical disk. It identifies the top-level contents of the logical disk. Each entry in the table is called a directory entry. The root directory acts as the foundation for the folder or directory structure. From the root directory, you can find any file or folder that exists on a disk. The root directory on a hard disk stores a maximum of 512 entries. (On FAT32 drives, the maximum number of entries is 65,536.)

Note: In Windows, directories and subdirectories other than the root directory are referred to as folders.

There are two good reasons for the root directory to be at the beginning of the disk. First, the operating system doesn't have to store the root directory location in order to retrieve it. Second, the root directory is the only file structure that always exists. Even if you delete every file and folder on a disk, you still have a root directory. Otherwise, you wouldn't be able to create new files and keep track of them.

On FAT32 drives, the root directory usually starts at cluster 2. If you optimize your drive with Speed Disk, the root directory is placed at the first usable cluster on the disk, which is usually cluster 2.

In Windows, an entry in the root directory may have one of two formats: a short filename format or a long filename (LFN) format.

{button ,AL(`About FAT32;About long filenames;About short filenames;Introducing Speed Disk',0,`',`')} More Info… Click here for more information.

About short filenames

Before the release of Windows 95, all directory entries used what is now known as the short filename format. Three types of directory entries used this format:

- File
- Subdirectory
- Volume label

Every directory entry is exactly 32 bytes, counted from 0 to 31. Here's a brief summary of the directory entry fields:

Byte Position	Field Name	Description
0-7	Filename	Name given to the entry.
8-10	File extension	File type identifier. Examples of common file extensions are .TXT, .DOC, and .EXE.
11	File attributes	Special properties of the entry, as explained in a separate section.
12	NT attribute	Reserved for use by Windows NT and Windows 2000.
13	Creation time (msec)	Milliseconds (thousandths of a second) after the minute the entry was created. This field is combined with the next one to provide a higher time resolution.
14-15	Creation time	Time the entry was created. This field is accurate to the minute.
16-17	Creation date	Date the file was created.
18-19	Last access date	Date the file was last accessed, either for reading or for writing.
20-21	Extended attribute	OS/2 field that points to other information about files, such as their icons and configuration information. Also called the EA handle, this field provides compatibility with OS/2.
22-23	Modified time	Time the file was last modified.
24-25	Modified date	Date the file was last modified.
26-27	Starting cluster	First cluster in the FAT chain for a file or folder. When retrieving a file or folder, the operating system must first find this cluster so the FAT chain for the file can be traversed.
28-31	File size	Number of bytes contained in the file.

{button ,AL(`About LFN directory entries;About long filenames',0,`',`')} More Info… Click here for more information.

About long filenames (LFN)

Before the introduction of Windows 95, the naming conventions for DOS and Windows files were restrictive. A filename had to follow the 8.3 format, that is, it could have a maximum of eight characters optionally followed by a file extension of up to three characters. A period separated the two parts.

In Windows, long filenames (including their extensions) can now contain up to 255 characters. As an extreme example, you could have a filename consisting of a period and a 254-character extension. (A long filename can contain more than one period. If it does, everything after the last period is considered the file extension.)

LFNs use the UNICODE character set, which requires two bytes for a single character. Short filenames use ASCII characters, which require one byte per character.

To support this new capability, the directory entries for LFNs are implemented differently from the other directory entry types. For example, the LFN "THIS IS MY VERY LONG FILENAME.TXT" takes three entries.

LFN entries are 32 bytes long. The interpretation of the bytes distinguishes a short filename entry from an LFN entry. The most important difference between LFN and the other types of directory entries is that a single long filename can require multiple LFN entries, since each LFN entry has room for only 13 characters of the filename. Here is a description of the fields for LFN entries.

Byte Position	**Field Name**	**Description**
0	Ordinal field	Hexadecimal number indicating the order of LFN entries, since LFNs may require more than one directory entry. The first entry is numbered 01h, the second 02h, and so forth. The last entry has its seventh bit, the most significant bit, set to 1.
1-10	Filename chars 1-5	First five filename characters within this LFN entry. This field must have an entry.
11	File attributes	Special properties of the entry, as explained in LFN directory entries.
12	Type	Reserved for future use. Currently it always contains 0, which denotes an LFN entry.
13	Checksum	Number verifying that the LFN directory entry is valid. The short filename for the file is the basis for this value.
14-25	Filename chars 6-11	Sixth through eleventh filename characters within this LFN entry. This field may be blank.
26-27	Cluster number	No longer used, always 0. This field is to maintain backwards compatibility with older versions of DOS and other software.
28-31	Filename chars 12-13	The twelfth and thirteenth filename characters within this LFN entry. This field may be blank.

{button ,AL(`About file attributes;About LFN directory entries;About short filenames',0,`',`')} More Info… Click here for more information.

About LFN directory entries

A long filename has one or more LFN (long file name) directory entries followed by its short filename directory entry. An LFN must have an associated short filename because only the short filename stores the file's starting cluster.

The short filename is a unique filename based on characters contained in the LFN. In addition to being the LFN's link to a file, the short filename allows Windows 3.x and DOS applications to access files.

For example, the LFN "THIS IS MY VERY LONG FILENAME.TXT" may have the short filename of THISIS~1.TXT. Windows generates this short filename. The tilde (~) and number ensure that no two long files have the same short filename. For example, if you create another file named "THIS IS MY LONG ESSAY.TXT", Windows assigns it a unique short filename of THISIS~2.TXT automatically.

Here is some other information about LFNs:

- Text that makes up the LFN is in reverse order. The first LFN directory entry stores the last characters in the filename.
- LFNs preserve the case used when the file was named. Short filenames do not.
- Even though LFNs preserve case, searching for, moving, opening, or otherwise manipulating files having LFNs is case-insensitive. For example, if you want to open a file named "Budget for the First Quarter," the file will still be opened if you specify the filename as "budget for the first quarter" or use any combination of uppercase and lowercase letters.
- An LFN becomes detached from its short filename if you move a file or rename the short filename using a pre-Windows 95 application or version of DOS. Use Norton Disk Doctor or Disk Editor to remove the LFN entries, or to reattach the LFNs to the short filename.

{button ,AL(`About file attributes;About long filenames;About short filenames',0,`',`')} More Info… Click here for more information.

About file attributes

File attributes are a combination of several directory properties. When the operating system interprets these properties, it knows whether to treat an entry as a file, directory, volume label, or long file name (LFN).

There are a total of six file attributes. Each attribute has a different bit position within the byte allocated for this field. A directory entry has a particular attribute if that attribute's bit is 1.

X Unused bits
These bits are always set to 0.

A Archive attribute
When this bit is 1, the entry has not been archived (backed up).

D Directory attribute
When this bit is 1, the entry is a folder.

V Volume label attribute
When this bit is 1, the entry denotes a volume label or LFN. However, an LFN also has 1 bit for the S, H, and R attributes described below.

S System attribute
When this bit is 1, the entry is a system file.

H Hidden attribute
When this bit is 1, the entry is not included in directory listings, such as what appears in the file pane of the Windows Explorer or when the DIR command is issued from the DOS prompt. In both cases, however, you have the option of overriding the exclusion of hidden entries.

R Read-only attribute
When this bit is 1, the entry can't be overwritten, deleted, or moved.

Notice there is no attribute to indicate that an entry is a file. Files without LFNs are entries where neither the directory nor volume label bit is set. Files with LFNs have LFN directory entries with their volume label, system, hidden, and read-only attributes set to 1. Windows uses this particular attribute combination to flag LFN directory entries because the combination would never be set by other software. This also provides compatibility with older applications.

{button ,AL(`About LFN directory entries;About long filenames',0,`',`')} More Info… Click here for more information.

Where data is stored

The last and largest portion of a logical disk is the data area, where all the files and folders you work with are located. A folder is like a file in form, but like the root directory in data content. To the operating system, a folder is a file whose contents happen to be the directory entries for all the files (and any folders) that are in that folder. The first two entries in a folder are always "." (which refers to this folder itself) and ".." (which refers to the parent folder).

By storing folders in the data area, the operating system can assign them space as needed, as if they were files. Like files, folders can expand, shrink, and be deleted. And by storing the starting cluster numbers for folders, the operating system can construct paths to files and other folders.

How data is retrieved

A directory entry identifies the starting cluster of a file and each file allocation table (FAT) entry for that file identifies its next cluster. The combination of these two pieces of information determines a chain of clusters that contains the file's data. The process of identifying this chain is called traversing the FAT chain.

Sometimes the directory entries and the FAT contain conflicting information. Such conflicts are usually the result of an interruption in normal processing while the FAT or a directory entry is being updated. Because the operating system performs the updates at different times, the FAT and the directory entries can become unsynchronized. The FAT is updated as you work on a file, whereas a directory entry is usually updated when you close the file.

When a file is deleted, its data is not really erased. The operating system marks the first byte of the filename with a special character to indicate the file has been erased, and then replaces the file's FAT entries with zeros. Because the data of the starting cluster number is not erased from the directory, UnErase Wizard can find and recover the first cluster, providing it hasn't been overwritten. However, if you have Norton Protection enabled, all deleted files (including some that are not protected by the Windows Recycle Bin) are moved to a hidden folder, greatly improving your chances of recovering deleted files.

Norton Disk Doctor checks to ensure that each used cluster in the FAT belongs to one, and only one, directory entry. If an entry in the FAT indicates a cluster is in use but that cluster is not part of a chain associated with any directory entry, Norton Disk Doctor reports it as a lost cluster. If an entry in the FAT indicates a cluster is part of a chain associated with more than one directory entry, Norton Disk Doctor reports a cross-linked file.

Norton Disk Doctor also compares the size of a file as reported by its directory entry to the size as determined by the number of clusters associated with it in the FAT. If the two sizes don't match, Norton Disk Doctor reports a file allocation error. Use Norton Disk Doctor to correct these and many other physical or logical disk problems.

{button ,AL(`About file allocation tables (FAT)',0,`',`')} More Info… Click here for more information.

Optimizing disk space and performance

Optimizing the performance and the space on your hard disk can make a big difference in the everyday functioning of your computer. Defragmenting files can substantially improve the performance of hard disks and using a disk cache can have a dramatic effect on the performance of both hard disks and CD-ROM drives. Increasing disk space on your hard disk gives you more room for a swap file and other data.

{button ,AL(`About hard disk drives;Defragmenting files;Disk caching;Increasing disk space',0,`',`')} More Info…
Click here for more information.

Defragmenting files

One factor affecting the performance of hard disk drives is file fragmentation. As you delete, modify, create, and copy files, they become fragmented or spread out over non-contiguous clusters.

Speed Disk solves this fragmentation problem by rearranging the files so they occupy contiguous clusters, thereby minimizing read/write head movement and optimizing access time. This reorganization process is known as defragmenting files. Other advantages of defragmenting files is the consolidation of unused space, better disk cache performance, and a better chance of recovering existing files if you delete them, or if they are damaged.

The Norton Utilities program Norton System Doctor includes the Disk Optimization sensor, which shows you the level of fragmentation for your hard disk.

{button ,AL(`About the Disk Optimization sensor;Optimizing disk space and performance;Disk caching;Introducing Speed Disk',0,`',`')} More Info… Click here for more information.

Disk caching

No matter how fast mechanical disk drives get, their data access speeds are always much slower than random access memory (RAM). To compensate, hardware and software vendors created disk caches. A disk cache sets aside RAM to store the data that applications are most likely to request from a drive. A disk cache can improve the effective data access speed of a disk drive by as much as a factor of ten.

There are four main types of disk caches:

- **Software disk caches** use main memory for the retrieval and temporary storage of disk data. They do not require any special hardware. VCACHE is an example of software disk cache.
- **On-board disk caches** use memory and a cache controller on the disk drive. Although they do not use any of the computer's RAM for caching operations, they are small and comparatively expensive.
- **Disk-caching controllers** use a controller card connected to the disk drive. This type of cache, like the on-board disk cache, uses its own memory rather than RAM and can be expensive. However, a disk-caching controller can offer better performance than an on-board disk cache, because it can overcome some of the limitations of other disk drive components.
- **Buffers** are memory chips on disk drives that temporarily store data but have limited impact on performance because they lack the ability to manage their contents. A track buffer retrieves an entire track of disk data when a single sector is read, so any subsequent requests for data on that track can be read from the buffer. Some disk drives come with buffers that can store much more than a track of data. CD-ROM drives often use buffers to hold the next sequential data the drive's photodetector encounters, on the assumption that it will be the next data requested.

You can use Norton System Doctor to monitor disk cache. Norton System Doctor provides three sensors that show you disk cache use and performance: the Cache Hits sensor, the Cache Memory Utilization sensor, and Disk Throughput sensor.

- Disk caching works less effectively if your disk is fragmented. Norton System Doctor includes the Disk Optimization sensor for monitoring to what degree your disk is fragmented. To defragment your disk, use Speed Disk.

{button ,AL(`About the Cache Hits sensor;About the Cache Memory sensor;About the Disk Throughput sensor;About the Disk Optimization sensor;Defragmenting files;Introducing Speed Disk',0,`',`')} More Info…

Click here for more information.

Read-ahead and write-behind caching

Read-ahead caching is a method of anticipating what data will be read next from a disk drive, and then reading it into memory, usually while the computer is idle. The result is reduced read/write head (or photodetector) movement and smoother operation.

Write-behind caching has the same result, but it involves holding in memory the data that is to be written to disk until the computer is idle. One problem with write-behind caching is the potential for data loss, which may occur if a computer is powered down before Windows indicates, "You may shut down." The cache is volatile memory, so you lose any data in the cache that was not written to disk before the computer was shut off.

{button ,AL(`Disk caching;About VCACHE',0,`',`')} More Info… Click here for more information.

About VCACHE

Windows has a virtual driver (VxD), called VCACHE, that handles disk caching. VCACHE is a replacement for the disk cache that came with previous versions of DOS and Windows (called SmartDrive). It has the advantage of being able to change the amount of memory it uses on the fly, something most caches loaded through DOS can't accomplish. When disk activity is high but the memory load is low, VCACHE increases its size to do as much processing as possible within RAM, thereby avoiding disk access. On the other hand, if memory usage is high (for example, if you're using a spreadsheet application to do calculations), VCACHE shrinks to maximize the amount of memory available for your application.

VCACHE has the ability to cache files residing on network volumes, comes free with Windows 95, Windows 98 and Windows Me, and requires no user setup. It uses the read-ahead and write-behind caching schemes. VCACHE is an example of software disk cache.

{button ,AL(`Disk caching;Read-ahead and write-behind caching',0,`',`')} More Info… Click here for more information.

Increasing disk space

Disk compression software (such as Microsoft’s DriveSpace) increases the amount of data that can be stored on a disk. (However, FAT32 volumes don’t support compressed drives.) This kind of software takes the data on a hard disk and looks for repeated sequences. It replaces these repeated sequences with abbreviations, using a sophisticated dictionary that provides a cross-reference between the repetitions and their abbreviations.

Here is a vastly simplified example: Disk compression software takes a text file with large amounts of space between portions of text and represents that space as a single space with the number of times it is repeated. When you open that file in a text editor, the disk compression software expands its notation back to the original large amounts of space.

When a disk compression product is installed, it usually takes all the existing data on a disk and compresses it into a single, hidden file that gets assigned its own drive letter. However, the installation program reserves a small amount of the original disk space for uncompressed system files. That way, the system can start up and load the software that handles the day-to-day interaction with the disk. In addition, most compression programs provide methods that make it easy for you to examine the contents of a compressed drive.

{button ,AL(`Optimizing disk space and performance',0,`',`')} More Info… Click here for more information.

The memory components of your computer

Computers store information based on size and how frequently it is accessed. The following is a list of the five most basic types of storage within a computer:

- The registers of the computer's central processing unit (CPU) are like a desktop. This is where work is performed and where people keep the information they need at their fingertips.
- The internal memory cache is like a desk drawer, where the next tier of often-needed information gets stored.
- The external memory cache is like the file cabinet in an office. It's still convenient, but it holds much more information than either the desktop or a desk drawer.
- Main memory (RAM) is like a file room down the hall (an even larger storage space, but not as readily accessible). Part of main memory is usually used for a type of cache called a disk cache.
- The hard disk is like an archive building across town, which stores the greatest amount of information but is the least convenient location. Windows uses part of the hard disk space for a work-in-progress area called a swap file.

The external memory cache is substantially larger than the internal memory cache, main memory is many times larger than the external memory cache, and the hard disk is huge compared to main memory. The internal memory cache is closest to the CPU registers and the hard disk is farthest from them. The farther a storage component is from the CPU registers, the longer the access time required for the CPU to retrieve information stored in that component. The shorter the access time, the greater the likelihood that the stored information is needed frequently.

In a broad sense, computer memory is any hardware or firmware that stores instructions or data for later retrieval. Four of the types of memory discussed so far (the CPU registers, the internal and external memory caches, and main memory) are often referred to as "primary storage media" or "primary storage devices," whereas the hard disk is an example of a "secondary storage device."

When people talk about how much "memory" their computers have, they're usually referring to main memory (RAM). Sometimes people get confused between this type of storage and the secondary storage offered by a hard disk. Main memory is volatile. It temporarily stores information related to the current processing the computer is performing. Typical sizes are 16 MB, 32 MB and 64 MB.

The hard disk provides long-term (although modifiable) storage for all your applications, documents, spreadsheets, databases, and other work, plus the programs required by the operating system. Typical sizes range from 1 GB, to 12 GB. When you use the Save command in an application, you're updating the information stored on the hard disk to reflect changes you've made. When you turn off your computer, the information in main memory gets wiped out, but the information on your hard disk remains intact.

{button ,AL(`About CPU registers;About external memory cache;About hard disks and swap files;About internal memory cache;About main memory',0,`',`')} More Info… Click here for more information.

About CPU registers

Inside the CPU are temporary storage areas called registers. Different functional units within a CPU use different registers to break down their work into a series of basic operations to be carried out. For example, the processor's control unit uses various registers while interpreting and executing instructions and the arithmetic unit uses other registers while performing calculations.

Because the registers are built into the processor, data can be transferred from one register to another at processor speeds. If all the work the processor is doing fits in the registers, then data access is immediate. However, because the registers hold only a very limited amount of information, they usually fill up quickly and the CPU must use its memory subsystem, the memory caches and main memory (RAM), to hold additional information.

Different processor chips use different register designs. These design differences can have a substantial impact on the actual performance of the registers.

{button ,AL(`The memory components of your computer',0,`',`')} More Info... Click here for more information.

About internal memory cache

Also called a primary cache, level-one cache, or L1 cache, an internal memory cache is built directly into the CPU, along with hardware circuitry called the cache controller. The internal memory cache stores frequently needed information (both program instructions and data) so the next time the processor needs any of that information, the cache can supply it more quickly than main memory (RAM).

This type of cache is relatively new. The Intel 386 SLC processor includes one, as do all 486 and Pentium processors. The 486 processor has an internal cache that stores both instructions and data, whereas Pentium processors have two separate internal caches, one for instructions and one for data. Separate caches are more effective because the access patterns for instructions and data are different. Instructions tend to be retrieved sequentially and are more frequently reused.

Like the CPU registers, an internal memory cache has a huge performance advantage over an external memory cache and main memory: the fact that it is internal to the processor chip itself means it is not subject to factors that limit the speed of external components (such as the speed of the bus and the distance to be traveled).

Because of physical limitations on how much cache can be built on a processor chip and because of their high cost, internal memory caches are small. Many have a capacity of only 8K. The 486 DX4 has a capacity of 16K and the Pentium processor has 8K on each of its two internal caches.

{button ,AL(`About cache performance;About caching schemes;The memory components of your computer',0,`',`')}
More Info… Click here for more information.

About external memory cache

Aside from the internal memory caches, the other type of memory cache is the external memory cache, sometimes referred to as secondary memory cache, level-two cache, or L2 cache. The external memory cache performs the same function as the internal memory cache: it makes frequently used information more quickly accessible than it would be if it had to be retrieved from main memory (RAM). If the internal memory cache does not contain information that the processor requests (or if there is no internal memory cache), the external memory cache intercepts the information request before it reaches main memory.

External memory caches were developed before internal memory caches. Almost every desktop computer now includes an external memory cache. Many portable computers do not. Like its internal counterpart, the external memory cache has its own cache controller.

The external memory cache can be mounted on the motherboard. Because it's a separate unit, it can't compete with the access speed of a cache that's integrated in the processor. However, the external cache has two major advantages over the internal type: larger capacity and lower cost. Typical external memory cache sizes are 128K and 256K. While caches within this size range can improve system performance dramatically, caches much larger than these provide diminishing returns: huge cost increases for comparatively small performance gains.

A bank of static random access memory (SRAM) chips makes up the external memory cache. SRAM chips are considerably faster than main memory (DRAM chips), but are also more expensive. Their design enables them to approach the processor's high speed yet prevents them from storing information as densely as main memory chips.

{button ,AL(`About cache performance;About caching schemes;The memory components of your computer',0,`',`')}
More Info… Click here for more information.

About main memory (RAM)

The computer's largest primary storage area is main memory, also known as random access memory (RAM). Before it can do any processing, the processor must load running applications and the data they require to this area from the disk. Main memory uses a type of chip called the DRAM chip.

Like SRAM (see external memory cache), DRAM includes "random access" in its name because all pieces of data it stores are equally accessible, no matter where the last accessed piece of data was located. Random access contrasts with sequential access, in which data is stored and retrieved in a linear fashion. (An example of a device that uses sequential access is the tape drive.)

SRAM and DRAM chips also share an important limitation: they provide only volatile storage of information. When you turn your computer off, you lose any information that resided in SRAM and DRAM that has not been written to disk.

DRAM chips have an additional disadvantage: their contents must be refreshed continually because the capacitors (which are similar to tiny rechargeable batteries that store the data) lose their charges every few milliseconds (thousandths of a second). This characteristic is what the "dynamic" part of the DRAM name means.

Older PCs frequently had their DRAM chips soldered onto the motherboard, but now DRAM chips are usually mounted on Single Inline Memory Modules (SIMMs).

Different DRAM chips offer different data access speeds, dependent primarily on their internal design and the external circuitry. The processor type and the CPU's speed determine the appropriate RAM speed. If you upgrade memory by installing slower RAM chips than your computer uses, wait states are necessary. If you install faster RAM chips, you'll increase the expense without affecting CPU or system speed.

Because SIMMs plug into sockets on the motherboard, installing more SIMMs is simple: make sure you get the type and upgrade quantity required by your computer. Turn off and unplug your computer before beginning installation. To prevent damage from electricity, also make sure you're grounded before touching any components inside your computer.

DRAM is comparatively inexpensive memory, but it is also much slower than SRAM. Not only are DRAM chips themselves intrinsically slower, but the time spent refreshing data further erodes DRAM speed.

Typically, part of a computer's main memory is dedicated to a special function: storing information that would otherwise be available only from the disk. This space in main memory is called a disk cache. Its purpose is to compensate for the huge speed discrepancy between main memory and disks. This discrepancy is far greater than the speed difference between an external memory cache and main memory.

{button ,AL(`About external memory cache;Disk caching;The memory components of your computer',0,`',`')} More Info… Click here for more information.

About hard disks and swap files

Compared to the primary storage components of a computer, the hard disk is gigantic. However, it's also incredibly slow. This combination of characteristics results in an interesting tension in the role the hard disk plays: on the one hand, computer designers go to great lengths to limit the number of disk accesses the CPU must make. On the other hand, a lot of that space is taken up by large applications. To accommodate these applications, Windows sets aside space for a swap file (sometimes referred to as a paging file).

Windows uses the swap file to store instructions and data that cannot fit in main memory (RAM). Since disks are so slow, Windows is designed to minimize the amount of swapping that occurs between the disk and main memory. For more information about how the swap file works, click More Info below and select a topic about virtual memory. Swap files are called paging files on Windows NT and Windows 2000.

Windows 3.1 enhanced mode gave you the choice of setting up either a permanent swap file, which was faster but required contiguous disk space that was never available for file storage, or a temporary swap file, which would get space allocated and de-allocated each time you started and exited Windows. Other swap file configuration options were available as well.

The Windows swap file is dynamic, which means it shrinks and grows to match changing demands on main memory and disk space. It also does not require contiguous disk space and comes already configured. Still, you can update the Windows system properties to specify a different disk drive to be used by the swap file or to change other swap file configuration settings.

- The more room you have available on your hard disk, the larger the Windows swap file can be, and so the more applications you can run simultaneously. Use Norton System Doctor to monitor disk space availability, physical memory usage, swap file size, and total virtual memory availability (through the Disk Space, Physical Memory, Virtual Memory, and Memory Load sensors, respectively).
- To make your system's swap file usage more efficient, set a minimum size for your swap file and have Speed Disk place it at the front of your disk. This will make your system run faster and keep the swap file defragmented. Also, you should never place your swap file on a compressed drive.

{button ,AL(`About CD-ROM drives;About floppy disk drives;About hard disk drives;About virtual memory;Introducing Norton System Doctor;Introducing Speed Disk',0,`',`')} More Info... Click here for more information.

About read-only memory (ROM)

As the name implies, read-only memory (ROM) stores information that can be read but not changed. ROM is the most common example of firmware, because software is embedded in ROM chips. Various components within a computer use ROM chips, including video, network, and other controller and expansion cards.

In addition, ROM is responsible for getting your system ready to go and for loading the operating system each time you start up your computer. Under DOS and Windows 3.x, ROM also played a significant role in handling the flow of data between the processor and the various input and output devices, for example, disk drives, printers, the video display, the keyboard, and the mouse. The detailed instructions governing these low-level operations are provided through software embedded in the ROM called the basic input/output system (BIOS). Individual system components, such as hard drive controllers, may have their own ROM BIOS chips.

However, under Windows, BIOS operations have been moved to the operating system itself, where they function much more efficiently. As a result, ROM chips no longer need to provide any BIOS functions for Windows applications, or for DOS applications running under Windows. (Some DOS applications, however, are written for operation under particular chip-based BIOS instructions. If necessary, use MS-DOS mode to run such applications, because MS-DOS mode still uses the ROM BIOS.)

In past years, ROM chips came from the factory with their programming already included in the chip circuitry. This practice proved to be expensive for computer manufacturers, because warehoused ROM chips would have to be discarded when bugs were found in the programming or when system designs changed. As a result, hardware companies developed programmable read-only memory (PROM) chips, which have their programming burned into them as a separate step that can be performed later in the computer manufacturing process.

Further innovations include these technologies:

- Erasable programmable read-only memory (EPROM)
- Electrically erasable programmable read-only memory (EEPROM)
- Flash memory
- Shadow

{button ,AL(`About DOS memory;The memory components of your computer',0,`',`')} More Info…Click here for more information.

Why all these memory components?

A computer could store everything in its CPU registers, in its main memory (RAM), and on its disks if it weren't for one major complication: the vast discrepancies in the speed of these three components. Because of these discrepancies, getting information to and from the CPU registers is much faster than getting information to and from main memory, which in turn is much faster than getting information to and from a disk.

CPU speed first outstripped the speed of the fastest main memory chips in 1982, when Intel introduced the 286 processor. Since then, the speed gap between these two components has grown wider still. Even a 25MHz processor is too fast for the fastest main memory chips. Yet as slow as main memory is compared to the CPU, main memory appears to be supersonic compared to the speed of a disk.

To give you an idea of the scale of these differences, consider this: the data access times for primary storage components are measured in nanoseconds (billionths of a second) whereas the access times for disk drives and other secondary storage devices are measured in milliseconds, or thousandths of a second.

Comparing the fastest access times of various types of storage indicates raw speed differences. Comparisons don't reflect how quickly information can actually be passed from one component to another. Many different factors affect the true performance of each component. For example, whereas the internal memory cache is capable of speeds that are ten times higher than those of the external memory cache, on average an internal memory cache actually performs about three times faster than an external memory cache.

Primary storage is astronomically faster than secondary storage. To prevent disk accesses from creating a processing bottleneck, a special part of main memory called the disk cache holds disk information. To compensate for the speed discrepancies between the CPU and main memory, the internal and external memory caches were developed. The faster the processor chip, the more dramatic the difference is between the speed of the registers and internal memory cache compared to the speed of the other components.

Speed is only one of the differences among the various types of memory. Size limitations and cost differences further complicate matters. For example, the internal memory cache can't be very large because space is extremely limited and this type of memory is also far more expensive than main memory. The least expensive and largest type of memory is main memory. These characteristics of main memory help compensate for the fact that it is also the slowest of the primary storage components.

In designing computer systems, manufacturers must constantly make decisions that represent trade-offs among performance, size, and cost constraints. One basic performance goal is to prevent the processor from communicating directly with main memory except as a last resort, because such communication always slows performance dramatically.

The System tab of the System Information program includes a benchmark feature that measures many of the speed factors that affect the performance of your system.

{button ,AL(`Disk caching;The memory components of your computer;Introducing System Information',0,`',`')}
More Info… Click here for more information.

More about caches

The basic theory behind caching revolves around two facts:

- Computers tend to use the same program instructions (and, to a lesser degree, the same data) repeatedly.
- The information a computer needs is often stored in adjacent, or at least fairly close, locations within memory.

Internal and external memory caches take advantage of these tendencies by trying to anticipate the information requirements of the computer.

Caches are transparent to the components with which they communicate. To the CPU, an internal memory cache or external memory cache looks like main memory (RAM).

{button ,AL(`About cache performance;About caching schemes;About external memory cache;About internal memory cache;Disk caching',0,`',`')} More Info… Click here for more information.

About cache performance

Measuring how often the processor finds the information it needs in a cache is a good way to assess the effectiveness of that cache. A cache hit means the required information is found, whereas a cache miss means that it is not. Once a cache fills with information, chances are excellent that the information the computer needs will be found in the cache.

When a cache hit occurs, the cache transfers the information to the processor with little or no delay. When a cache miss occurs, the cache must issue what are known as wait states to give the required information time to arrive from main memory (RAM).

To see how much of your processor's time is spent idle, use Norton System Doctor to activate the CPU Utilization sensor. To see how many cache hits your system is receiving, activate the Cache Hits sensor.

Effective caching can provide cache hit ratios above 90%, meaning that the requested information is found in the cache more than 90% of the time. Another way of expressing this ratio is that the computer has zero-wait-state performance 90% of the time.

{button ,AL(`About caching schemes;More about caches;Disk caching;About the CPU Utilization sensor;About the Cache Hits sensor',0,`',`')} More Info… Click here for more information.

About caching schemes

Because caches are comparatively small, they need to be discriminating about what information they keep and (when they fill up) what information they replace. Cache designers build the algorithms, or procedural rules, that determine when and what to store, when and what to discard, and how to organize the cache space. The resulting caching scheme is one of the most important factors affecting cache performance.

Many different caching algorithms (and variations on algorithms) exist. However, most caches use a form of the least-recently used (LRU) algorithm to determine what contents to discard when the cache fills up. For PCs, a pure LRU algorithm is prohibitively expensive because of the circuitry required. Therefore, most caches use approximations of the LRU algorithm.

In addition to an algorithm determining how a cache discards information, caches must define other aspects of their behavior:

- **Organization of contents**
Many internal memory caches use what is known as four-way set-associative caching, whereas many external memory caches use direct-mapped caching.
- **Handling of read requests**
When the CPU requests information from memory, the cache must have a means to intercept the request and, if it contains the information, respond. The most efficient caches correctly anticipate what information will be requested.
- **Handling of write requests**
The Pentium memory cache system uses the write-back caching algorithm. Among systems using other types of processors, write-through caching is the most widely employed writing scheme.

Cache designers consider a number of conflicting performance and cost factors. For example, the cache must contain the information actually requested or its cache hit rate will be unacceptably low. But a high cache hit rate means nothing if the process of locating and retrieving information is so slow that little time is actually saved. Furthermore, a very sophisticated cache implementation may be impractical because of its high price tag.

{button ,AL(`About cache performance;About external memory cache;About internal memory cache;Disk caching;More about caches',0,`',`')} More Info… Click here for more information.

About memory management

Part of the job of an operating system is to manage the use of the computer's memory. DOS was designed around the capabilities of the Intel 8086 processor, which could handle only 1 MB of memory. To compensate for this limitation, many different memory-management programs and standards (and the various versions of Windows itself) were developed and have provided some relief. All of these efforts, however, were ultimately hamstrung by the increasingly severe discrepancy between the capabilities of DOS and the capabilities of the hardware that DOS was supposed to control.

Windows   overcomes the 1 MB memory barrier of DOS without having to resort to tricks involving expanded memory or extended memory. Windows supports the 4 GB memory capabilities that processors have had ever since the introduction of the Intel 386. In addition to main memory (RAM), Windows manages the virtual memory provided through the Windows swap file. And, to provide backward compatibility with all the hardware, DOS programs, and older Windows applications currently in use, Windows still provides special handling of that first megabyte of computer memory, now referred to as DOS memory.

{button ,AL(`About DOS memory;About virtual memory',0,`',`')}   More Info…   Click here for more information.

About virtual memory

Virtual memory gives the illusion that a computer has more physical memory than it actually does. It is a technique that enables programs to share the computer's physical memory. Virtual memory increases the apparent size of main memory by augmenting it with hard disk space and using virtual addresses to identify actual memory locations. This comes from the mainframe world and has been supported in every Intel processor since the 386. (The 286 provided a simpler form of virtual memory management.)

The processor supports virtual memory directly. It allows different virtual address spaces to share the same physical memory. The built-in memory management functions greatly increase efficiency and reduce the overhead that otherwise would be involved if virtual memory were provided entirely at the operating system level.

However, DOS is fundamentally a real-mode operating system and therefore could not take advantage of the processor's virtual memory capabilities. Windows began incorporating virtual memory features in 1987. Newer versions of Windows expand on the capabilities of Windows 3.1, which offered virtual memory in its enhanced mode. As soon as Windows loads, it enables virtual memory and runs in 32-bit protected mode. Under Windows, if you don't have any hard disk space left, you don't have any memory.

To monitor the amount of free disk space on your system, activate the Disk Space sensor in Norton System Doctor.

The total amount of virtual memory available on a computer is the sum of main memory plus the disk space set aside for the Windows swap file.

When an application requests memory and main memory is full, Windows verifies that enough room is available in the swap file. If the swap file is full, but there is adequate disk space, Windows may increase the size of the swap file. Until the application actually reads from or writes to a particular part of its virtual memory space, Windows never has to allocate any corresponding physical memory. If the application needs access to information that has been swapped out to disk, Windows retrieves the information from the swap file back to main memory so the application can access it. Windows uses a least-recently-used (LRU) algorithm to decide what to keep in main memory and what to swap out to disk, but it does not attempt to anticipate future information needs.

Since disks are so slow, Windows is designed to minimize the amount of swapping that occurs between the disk and main memory. New 32-bit Windows applications never need any read-only information, which includes all their program instructions, to be swapped in and out. Instead, read-only information can be copied directly to main memory from the hard disk as needed, and discarded from main memory when no longer needed. As more 32-bit applications become available, this feature should reduce the amount of swapping required.

When you open an application and retrieve a document to work on, the required instructions and data are loaded in the various memory locations, with the swap file holding the information that won't fit in the primary storage locations.

If you open a second application and start working on a file in it, some of the instructions and data for the first application must be swapped out to make room for the instructions and data for the second application. Because of the higher demands on memory, the swap file increases in size.

To give each application time to do its processing, Windows multitasks. Windows 3.x used cooperative multitasking, which lets one task monopolize the processor until its program relinquished control so a different program could perform its processing. Windows now uses preemptive multitasking for 32-bit applications, whereby Windows itself decides how much time the processor is to spend on each task before control passes to a different task automatically.

When you use an application's Save command, you cause Windows to write the current data to the hard disk, not to the swap file, but to the mass storage area of the hard disk. When you close an application, the current data is written to the hard disk and all instructions and data for that application are marked for discarding from the swap file.

When organizing the 4 GB of virtual memory space, Windows reserves the addresses above 3 GB for its components. Addresses between 2 GB and 3 GB are for 16-bit Windows applications, shared application components, and shared system components. Addresses between 4 MB and 2 GB provide private space for 32-bit applications and the first MB is for DOS memory.

{button ,AL(`About hard disks and swap files;About memory management;Disk caching;About the Disk Space sensor',0,`',`')} More Info… Click here for more information.

About DOS memory

If you rarely run DOS applications and use only fairly recent, name-brand devices (such as network adapters, CD-ROM drives, sound cards, and multimedia hardware and software), you may never encounter any of the issues surrounding the use of the first megabyte of memory in your computer, known as DOS memory. But if you have older hardware or applications, run DOS-based games, have complex CONFIG.SYS and AUTOEXEC.BAT files even after installing Windows, or experience problems with certain devices or DOS programs, a knowledge of DOS memory helps you get the most out of your system.

The original IBM PC was designed around the capabilities (and limitations) of the Intel 8086 processor. DOS, in turn, was designed for the IBM PC, which set aside 384K of its 1 MB memory space for system use and future expansion. Because of the phenomenal success of IBM and compatible PCs, this original design has been supported by every version of DOS and by programs designed to run in DOS. A PC must comply with this original memory specification (and with many other design characteristics) to claim IBM compatibility.

DOS calls the 384K reserved area upper memory and the remaining 640K conventional memory. DOS also recognizes two other types of memory: expanded memory and extended memory.

{button ,AL(`About conventional memory;About expanded memory;About extended memory;About memory management;About upper memory',0,`',`')} More Info… Click here for more information.

About conventional memory

At the time that the original PC was designed, its 1 MB of main memory (RAM) was considered a large amount because the previous generation of processors (and therefore the existing microprocessor-based programs) could handle only 64K. The notion was that programs written to run under DOS would not need more than the 640K of conventional memory that DOS made available to them. In fact, many DOS applications were developed that confined themselves to this memory area.

To run, a DOS application must have contiguous space in conventional memory. However, DOS applications share conventional memory with a number of components, including some BIOS operations and DOS internal commands. The amount of space actually available for DOS applications to run, therefore, has always been at a premium.

Traditionally, DOS applications' main competition for conventional memory has come from the CONFIG.SYS and AUTOEXEC.BAT files, which can load device drivers, TSRs, and programs in conventional memory. Under DOS, various memory-management techniques were developed to move some of this memory usage to other types of memory. In its enhanced mode, Windows 3.1 replaced some device drivers with virtual device drivers and DLLs, which use extended memory.

Windows 95, Windows 98 and Windows Me further reduce the demands on conventional memory by replacing many more of the standard device drivers and by replacing many of the BIOS operations with protected-mode facilities within the operating system itself. As a result, much of the 640K of conventional memory is now actually available to DOS applications. However, Windows also needs a little conventional memory for its own operations and every running Windows application consumes a small amount of conventional memory.

{button ,AL(`About extended memory;About DOS memory',0,`',`')} More Info… Click here for more information.

About upper memory

The 384K of memory immediately above conventional memory is referred to as upper memory. The original IBM PC reserved all of this memory area (between 640K and 1 MB) for ROM, display adapters, other system uses, and future system use. However, much of this space is not needed for these purposes in typical PCs. The unused portions of upper memory are called upper memory blocks (UMBs).

Various memory managers have taken advantage of UMBs by filling them with device drivers and TSRs from conventional memory. A driver or TSR cannot be larger than the UMB to which it is moved, so this remapping requires matching UMB sizes with device driver and TSR sizes. (It also requires a 386 or higher processor.) Because DOS applications require contiguous space in conventional memory, UMBs cannot be used to increase the memory available to them.

Another use for UMBs is to provide expanded memory.

{button ,AL(`About conventional memory;About expanded memory;About DOS memory;About read-only memory',0,`',`')} More Info… Click here for more information.

About expanded memory

As DOS applications began outgrowing the space available to them in conventional memory, a hardware trick, called memory mapping, was used to allow programs to access expanded memory (EMS).

Initially, expanded memory consisted of memory management software as well as a separately installed memory card. Later, expanded memory emulators eliminated the need for a separate card by enabling other types of memory, such as extended memory, to be manipulated like expanded memory.

Expanded memory addresses are remapped so they appear to be within upper memory. When a program requests information that is in expanded memory, an expanded memory manager maps that information to addresses in the page frame.

This access to a separate memory store gives the processor more than 1 MB of memory while in real mode. By changing the mapping, comparatively large amounts of information can be switched in and out of the page frame as needed.

For computers using 8086 and 8088 processors, expanded memory is the only way programs can get around the 640K limitation of conventional memory. However, only programs that have been designed for expanded memory can take advantage of it. Windows makes expanded memory available automatically to those DOS applications that use it, so EMS is outdated on systems running Windows.

If your CONFIG.SYS file includes "noems" in the line that loads EMM386.EXE (a device driver that uses extended memory to emulate expanded memory), Windows cannot provide access to expanded memory.

{button ,AL(`About conventional memory;About DOS memory;About upper memory',0,`',`')} More Info… Click here for more information.

About extended memory

The 286 processor increased the PC's memory capacity from 1 MB to 16 MB. The 386 increased memory capacity even further, to 4 GB. All of the main memory (RAM) in a computer that is above the first megabyte is called extended memory (XMS). Although DOS cannot access it, extended memory opened up a vast world of new possibilities.

To avail themselves of this memory windfall, software developers began designing more powerful programs capable of switching to protected mode for some of their processing. One of these programs was Windows 3.0, which incorporated a specification that standardized the way extended memory was to be used, so programs would not interfere with each other's processing. This specification is called the DOS Protected Mode Interface (DPMI). It enables DOS applications to use up to 16 MB of extended memory. DOS applications that adhere to the DPMI standard can run under Windows 95, Windows 98, and Windows Me (as well as under Windows 3.x). Windows 95, Windows 98, and Windows Me have access to all of the extended memory that is available.

Extended memory can be used to emulate expanded memory. Unlike expanded memory, extended memory gives programs direct access to its memory.

Another use for extended memory is to free additional space in conventional memory. The first 64K (actually, 64K minus 16 bytes) of extended memory can be used much like the unused blocks of upper memory: a device driver or TSR can be moved from conventional memory to this extended memory region, called the high memory area (HMA). However, only one driver or TSR can be moved to this region, even if more than one can fit.

{button ,AL(`About conventional memory;About expanded memory;About DOS memory',0,`',`')} More Info…
Click here for more information.

About Windows and DOS memory

After you've installed Windows 95, Windows 98 or Windows Me, your computer still initializes itself in real mode when you turn it on. During startup, Windows loads a special version of DOS unique to Windows, loads the real-mode device drivers in the CONFIG.SYS file, executes the commands in the AUTOEXEC.BAT file, and saves an image of the real-mode state of the system at startup time. Only then does Windows switch to protected mode.

When you start a DOS application from Windows, the operating system creates what is known as a DOS virtual machine (DOS VM), in which the DOS application runs. This VM inherits the real-mode attributes stored at system startup. It is a faithful simulation of an original IBM PC running DOS, using only the physical memory addresses for the first megabyte of memory. To imitate the original PC's processor, Windows uses virtual mode. To imitate the hardware and software, Windows uses virtual drivers and DLLs where possible. Otherwise it uses the commands in the AUTOEXEC.BAT and CONFIG.SYS files.

Windows eliminates the need for the most common drivers and TSRs formerly loaded through the CONFIG.SYS and AUTOEXEC.BAT files. These real-mode components used space in conventional memory, whereas their protected-mode replacements can load in any part of memory. As a result, DOS applications have more room available in which to run. However, Windows does not have drivers for every device developed for DOS, and there are cases where real-mode drivers or TSRs cannot be safely replaced with protected-mode drivers or DLLs. You therefore may still need to load device drivers through CONFIG.SYS and start TSRs or similar programs through AUTOEXEC.BAT.

To customize the behavior of a DOS VM, change its properties by right-clicking the icon and selecting Properties, or click Properties on the DOS window's Window menu. For example, you can create a custom CONFIG.SYS or AUTOEXEC.BAT file to be used only with this DOS application, run a batch file before the application starts, prevent the application from writing to the memory used by the DOS system area (to reduce the chance that the application will crash your system), or specify the maximum amount of each type of memory the program uses.

If a DOS application requires complete control over the system (Disk Editor is an example of such an application) you should be able to run the application in MS-DOS mode rather than in a DOS VM. In MS-DOS mode, Windows closes all open applications and running processes, removes all but a stub of itself from memory, loads a real-mode version of DOS, and turns control of the system over to the application. When you close the application, Windows uses its stub to restart.

{button ,AL(`About DOS memory;About memory management',0,`',`')} More Info… Click here for more information.

Browser not found

Norton Utilities could not find a default Web browser on your computer. You must have a Web browser installed to view websites. If one is installed, start it manually to visit the website mentioned in the help topic.

Introducing LiveUpdate

Symantec's LiveUpdate technology makes it easy to keep Norton Utilities up-to-date. LiveUpdate uses your computer's modem or Internet connection to download updates directly from Symantec.

Run LiveUpdate once a month to see if there are updates to Norton Utilities. Run LiveUpdate from the Norton Utilities Integrator to update all your Norton Utilities.

{button ,AL(`Technical Support',0,`',`')} More Info… Click here for more information.

Select to have Norton Disk Doctor diagnose and repair one or more disks automatically whenever Windows is started.

Select to have Norton System Doctor begin monitoring critical computer information whenever Windows is started.

Select to prevent the Norton Utilities splash screen from displaying each time a Norton Utilities program is started. The splash screen is the colored graphics screen displaying the logo for Symantec Norton Utilities.

Select to prevent Norton Utilities program introduction screens from displaying each time a Norton Utilities program is started. The Norton Utilities program introduction screens provide brief descriptions of the programs when the programs are started.

This option disables or enables all program introduction screens. If the check box has a gray check mark, only some of the program introductions have been disabled. You can disable selective screens by selecting the Don't Show Again check boxes on the individual screens.

To get information about your system

No matter what your level of expertise, questions about your system arise from time to time. For example, if you call a vendor for technical support, they're likely to ask about your system's BIOS, bus type, processor, ports, video, and multimedia capabilities.

System Information gives you quick, one-stop access to basic or detailed information about your system. It also can conduct benchmark tests of your system and disks to help you gauge your system's performance.

1 Click here to open System Information.

2 Click a tab to display a category of information.

Removes all traces of selected files or folders from your disk. It can also wipe the free space on your disk, ensuring that previously deleted information is not left on your disk.

Give your system a comprehensive checkup with Norton System Check. Norton System Check gives you all the power of Norton Utilities with one click. It not only finds problems, it also suggest how to fix them. You can run Norton System Check manually, or except on Windows 95, schedule it to perform periodic maintenance on your computer.

Gives you all the power of Norton Utilities with one click. It not only finds problems, it also suggests how to fix them. You can run Norton System Check manually, or except on Windows 95, schedule it to perform periodic maintenance on your computer.

About running Norton Utilities directly from the CD

Several of the Norton Utilities can be run directly from the CD. This is useful if you have a problem on a computer that does not have Norton Utilities installed. The following Norton Utilities can be run directly from the CD:

- Norton Disk Doctor
- Norton WinDoctor

{button ,JI(`nu.HLP>maintwo',`IDH_To_run_Norton_Utilties_directly_from_the_CD')} More Info…

To run Norton Utilities directly from the CD

1 Put your installation CD into the CD-ROM drive.

2 When the first screen appears, click Launch Utilities From CD.

Do not click Install Norton Utilities. Doing so will overwrite information on your hard disk, which could prevent recovery of erased files.

3 When the Norton Utilities CD screen appears, click the Norton Utility you want to run.

{button ,JI(`nu.HLP',`IDH_About_running_Norton_Utilties_directly_from_the_CD')} More Info…

Options: General Settings

Use this tab to schedule Norton System Check. You can run Norton System Check at a regular time, or every time you start your computer.

This dialog box contains the following:

Display Splash Screens

Select this option to see the graphical introductory screens as each of the Norton Utilities opens.

Display Program Introductions

Select this option to see the Welcome To screens that describe the operation of each of the Norton Utilities.

Replace Scan Disk with Norton Disk Doctor

Select this option to use Norton Disk Doctor instead of Windows Scan Disk when Windows encounters a disk problem.

Options: Startup Programs

Use this tab to set the Norton Utilities applications that you want to start when Windows starts. You can select any of the following programs:

- Norton Disk Doctor
- Norton System Doctor
- Norton Protected Recycle Bin
- Norton Speed Start (Windows 95 only)

Options: Recycle Bin

Use this tab to set the appearance and double-click action of the Recycle Bin:

Double-clicking item opens

You can choose what happens when you double-click the Recycle Bin icon on your desktop.

Appearance

You can set appearance of the Recycle Bin icon and the text that appears under it.

You must Enable Protection on the Norton Protection tab before you can enable Show Norton Protection Status.

Options: Norton Protection

Use this tab to set Norton Protection options.

Drive Settings

You can determine which drives are protected, and for how long files are saved.

System Settings

- Exclude certain types of files from being protected..
- Empty the files from Norton Protection.
- Remove Norton Protection from memory, and disable it from starting with Windows.

Displays graphical introductory screens as Norton Utilities programs start.

Displays screens with introductory information as Norton Utilities programs start.

Uses Norton Disk Doctor instead of Windows Scan Disk when Windows encounters a disk problem.

Starts Norton Disk Doctor when you start Windows.

Starts Norton System Doctor when you start Windows.

Starts Norton Protected Recycle Bin when you start Windows.

Starts Norton SpeedStart when you start Windows.

Displays the Norton WipeInfo icon on the desktop.

Sets the number of times to repeat the entire wipe procedure.

Writes the value 0, or any other value you choose, to replace the data being wiped.

Specifies the value to be written by WipeInfo. Can be any value from 0 to 255.

A 7-pass procedure that conforms to the Sanitization procedure specified in DoD document 5220-22-M, National Industrial Security Program Operating Manual.

Sets the number of times to repeat writing 1's and 0's.

Resets all entries to their preset values.

Checks for problems that affect performance, such as disk fragmentation.

Checks for problems with Windows, such as invalid shortcuts and registry problems.

Checks for disks for problems with the integrity of the file system.

Performs preventative maintenance, such as checking for adequate disk space and to see if your Rescue Disks are out of date.

Select the frequency at which you want System Check to run.

Set the time you want System Check to run.

Select the day of the week on which you want System Check to run.

Select the day of the month on which you want System Check to run.