How a computer does multitasking

Most computers can do only one thing at a time. They have a single microprocessor, and they process information one at a time. But they operate very quickly. In fact, microcomputers today typically operate at tens of millions of operations per second.

Here's the problem. Even though computers operate very quickly, doing one thing at a time is often too constraining for the way people work. We are used to doing more than one thing at a time, and that's how we want our computers to work. And that's where multitasking comes in. it allows the computer to do several things at once effectively.

Say, for instance, that you are working in an office, making up mailing labels for a newsletter. You want the computer to print the mailing labels, but you don't want to sit around staring at the ceiling while the labels print out. You need to write some special letters to include with some of the newsletters while the labels are printing out, you want the computer to do two things: print and accept input.

There are several ways that the computer can perform these two takss at once. If, for example, the mailing list is small enough to fit in the printer's memory, the computer can simply read the entire list to the printer to start the printing process, and then turn its full attention to your letter. If the mailing list is too large for the printer to hold at once, then the computer will need to send the first block of names and addresses to the printer and while it is printing, to let you write the letter. When the printer is done with the first block of mailing labels, it will send another block of labels to the printer. The process is invisible to you, because it occurs so quickly (between the characters you type on the screen) that you never notice it.

Interestingly, if the computer is multitasking between two programs instead of a program an I/O task such as printing, it will generally split up its time differently. If the two tasks have the same priority, the computer will divide its time with equal slices, giving the first task perhaps 100 processing cycles, and then switching to the other task, giving it also 100 processing cycles, and so on.

In most cases you will not see any of this or the effects of multitasking unless you are doing so many things at once that the computer causes some of the processes to slow down. That is when the issue of priority becomes important. If the two tasks do not have the same priority, you can give the one that requires more processing time (or needs to complete faster) more processing time. Multitasking systems generally allow you to assign priority levels to any task. So, if you want one task to take 75 percent of the computer's time, any other tasks running at the same time will have to share the remaining 25 percent of the computer's time.

You'll also need to be careful of how many tasks you ask the computer to do at the same time. It is possible to overload even the fastest computer systems. If, for instance, the multitasking computer is a network file server and too many users want access to it at the same time, then the whole network can slow down. This is also true if a single system tries to do too many things at once. Multitasking actually creates its own processing overhead. So if you have many tasks to complete, in some cases you may be better of running a new task first and the rest later.

One thing that you should keep an eye on is the new technology being released in microprocessors. Many of the new processors are much faster than the existing ones. In fact chips such as Pentium's give us performance leaps of 10x and more in some applications.

Multitasking and networking are prime candidates for the faster machines. Also, more and more systems are being developed that include many processors. With this kind of power it is inevitable that processing limitations for common applications such as ones we see today will virtually disappear. Multitasking allows you to assign priorities to different tasks.