PLEASE TYPE, OR PRINT LEGIBLY, IN BLACK INK - THIS PAGE MAY BE SCANNED

RAMSEY LIBRARY RESERVE MATERIALS FORM

PHOTOCOPIES

THIS ITEM TO BE PLACED ON:

____ TRADITIONAL RESERVE (ALLOW 48 HOURS TO PROCESS)

_✓_ ELECTRONIC RESERVE (ALLOW 1 WEEK TO PROCESS)

TITLE AS YOU WANT IT LISTED: "Types of Software"

DATE 2/5/04 _✓_ FALL

INSTRUCTOR M.L.MANNS ____ SPRING

COURSE NUMBER Mgmt 386 ____ SUMMER

COURSE NAME Systems & Info Mgmt

INSTITUTION UNCA

Please Note: This form must accompany each reserve article

In compliance with copyright law, the 1st page of any reserve article *must* contain the bibliographic information requested below, in full. Fill out a separate sheet for each item you wish to be placed on reserve.

( ) Title of Periodical _____

Title of Article _____

Author _____ Year _____

Volume _____ Issue _____ Pages _____

(✓) Title of Book Management Information Systems

Article Title or Chapter Types of Software & Programming Languages

Author Laudon & Laudon

Edition/Date of Publication 6th Publisher Prentice Hall

Pages 194-212

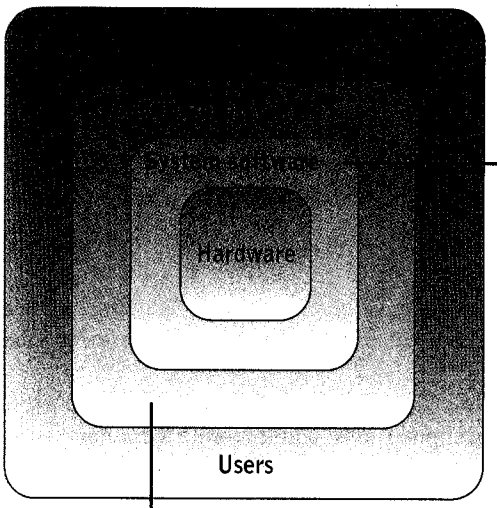
Laudon, K.C. & Laudon, J.P. (2000)
Management Information Systems. 6th ed.
New Jersey: Prentice Hall Publishing Co.

"Types of Software"

7.1 What Is Software?

software The detailed instructions that control the operation of a computer system.

Software is the detailed instructions that control the operation of a computer system. Without software, computer hardware could not perform the tasks we associate with computers. The functions of software are to (1) manage the computer resources of the organization, (2) provide tools for human beings to take advantage of these resources, and (3) act as an



SYSTEM SOFTWARE

Operating System

Schedules computer events
Allocates computer resources
Monitors events

Language Translators

Interpreters
Compilers

Utility Programs

Routine operations (e.g., sort, list, print)
Manage data (e.g., create files, merge files)

APPLICATION SOFTWARE

Programming languages
Assembly language
FORTRAN
COBOL
BASIC
PASCAL
C
"Fourth-generation" languages and PC software tools

Figure 7-1 The major types of software. The relationship between the system software, application software, and users can be illustrated by a series of nested boxes. System software—consisting of operating systems, language translators, and utility programs—controls access to the hardware. Application software, such as the programming languages and "fourth-generation" languages, must work through the system software to operate. The user interacts primarily with the application software.

intermediary between organizations and stored information. Selecting appropriate software for the organization is a key management decision.

Software Programs

A software **program** is a of statements or instructions to the computer. The process of writing or coding programs is termed *programming,* and individuals who specialize in this task are called *programmers.*

The **stored program concept** means that a program must be stored in the computer's primary storage along with the required data in order to execute, or have its instructions performed by the computer. Once a program has finished executing, the computer hardware can be used for another task when a new program is loaded into memory.

program A series of statements or instructions to the computer.

stored program concept The idea that a program cannot be executed unless it is stored in a computer's primary storage along with required data.

Major Types of Software

There are two major types of software: system software and application software. Each kind performs a different function. **System software** is a set of generalized programs that manage the resources of the computer, such as the central processor, communications links, and peripheral devices. Programmers who write system software are called *system programmers.*

Application software describes the programs that are written for or by users to apply the computer to a specific task. Software for processing an order or generating a mailing list is application software. Programmers who write application software are called *application programmers.*

The types of software are interrelated and can be thought of as a set of nested boxes, each of which must interact closely with the other boxes surrounding it. Figure 7-1 illustrates this relationship. The system software surrounds and controls access to the hardware. Application software must work through the system software in order to operate. End users work primarily with application software. Each type of software must be specially designed to a specific machine to ensure its compatibility.

system software Generalized programs that manage the resources of the computer, such as the central processor, communications links, and peripheral devices.

application software Programs written for a specific application to perform functions specified by end users.

7.2 System Software

System software coordinates the various parts of the computer system and mediates between application software and computer hardware. The system software that manages and controls the activities of the computer is called the **operating system.** Other system software consists of computer language translation programs that convert programming languages into machine language and utility programs that perform common processing tasks.

operating system The system software that manages and controls the activities of the computer.

Functions of the Operating System

One way to look at the operating system is as the system's chief manager. Operating system software decides which computer resources will be used, which programs will be run, and the order in which activities will take place.

An operating system performs three functions. It allocates and assigns system resources; it schedules the use of computer resources and computer jobs; and it monitors computer system activities.

Allocation and Assignment

The operating system allocates resources to the application jobs in the execution queue. It provides locations in primary memory for data and programs and controls the input and output devices such as printers, terminals, and telecommunication links.

Scheduling

Thousands of pieces of work can be going on in a computer simultaneously. The operating system decides when to schedule the jobs that have been submitted and when to coordinate the scheduling in various areas of the computer so that different parts of different jobs can be worked on at the same time. For instance, while a program is executing, the operating system is scheduling the use of input and output devices. Not all jobs are performed in the order they are submitted; the operating system must schedule these jobs according to organizational priorities. On-line order processing may have priority over a job to generate mailing lists and labels.

Monitoring

The operating system monitors the activities of the computer system. It keeps track of each computer job and may also keep track of who is using the system, of what programs have been run, and of any unauthorized attempts to access the system. Information system security is discussed in detail in Chapter 16.

Multiprogramming, Virtual Storage, Time Sharing, and Multiprocessing

How is it possible for 1000 or more users sitting at remote terminals to use a computer information system simultaneously if, as we stated in the previous chapter, most computers can execute only one instruction from one program at a time? How can computers run thousands of programs? The answer is that the computer has a series of specialized operating system capabilities.

Multiprogramming

multiprogramming A method of executing two or more programs concurrently using the same computer. The CPU executes only one program but can service the input/output needs of others at the same time.

The most important operating system capability for sharing computer resources is **multiprogramming.** Multiprogramming permits multiple programs to share a computer system's resources at any one time through concurrent use of a CPU. By concurrent use, we mean that only one program is actually using the CPU at any given moment but that the input/output needs of other programs can be serviced at the same time. Two or more programs are active at the same time, but they do not use the same computer resources simultaneously. With multiprogramming, a group of programs takes turns using the processor.

Figure 7-2 shows how three programs in a multiprogramming environment can be stored in primary storage. The first program executes until an input/output event is read in the pro-

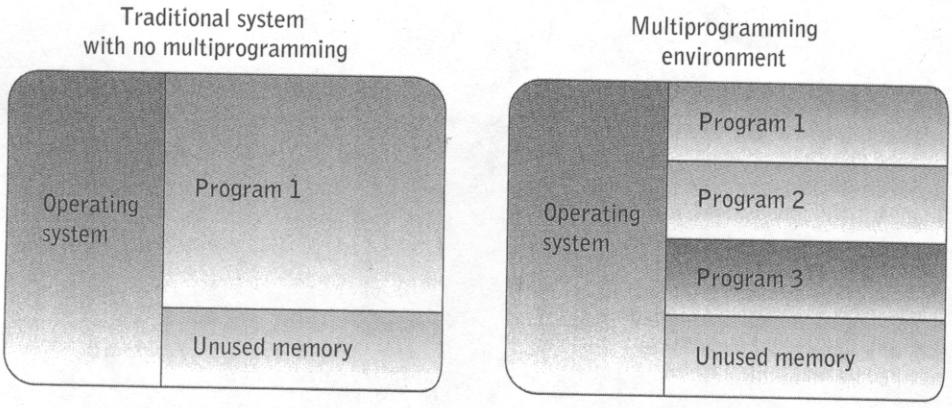


Figure 7-2 Single-program execution versus multiprogramming. In multiprogramming, the computer can be used much more efficiently because a number of programs can be executing concurrently. Several complete programs are loaded into memory. This memory management aspect of the operating system greatly increases throughput by better management of high-speed memory and input/output devices.

gram. The operating system then directs a channel (a small processor limited to input and output functions) to read the input and move the output to an output device. The CPU moves to the second program until an input/output statement occurs. At this point, the CPU switches to the execution of the third program, and so forth, until eventually all three programs have been executed. In this manner, many different programs can be executing at the same time, although different resources within the CPU are actually being utilized.

The first operating systems executed only one program at a time. Before multiprogramming, when a program read data off a tape or disk or wrote data to a printer, the entire CPU came to a stop. This was a very inefficient way to use the computer. With multiprogramming, the CPU utilization rate is much higher.

Multitasking

Multitasking refers to multiprogramming on single-user operating systems such as those in older personal computers. One person can run two or more programs or program tasks concurrently on a single computer. For example, a sales representative could write a letter to prospective clients with a word processing program while simultaneously using a database program to search for all sales contracts in a particular city or geographic area. Instead of terminating the session with the word processing program, returning to the operating system, and then initiating a session with the database program, multitasking allows the sales representative to display both programs on the computer screen and work with them at the same time.

multitasking The multiprogramming capability of primarily single-user operating systems, such as those for older PCs.

Virtual Storage

Virtual storage handles programs more efficiently because the computer divides the programs into small fixed- or variable-length portions, storing only a small portion of the program in primary memory at one time. If only two or three large programs can be read into memory, a certain part of main memory generally remains underutilized because the programs add up to less than the total amount of primary storage space available. Given the limited size of primary memory, only a small number of programs can reside in primary storage at any given time.

Only a few statements of a program actually execute at any given moment. Virtual storage breaks a program into a number of fixed-length portions called **pages** or into variable-length portions called *segments*. Each of these portions is relatively small (a page is approximately 2 to 4 kilobytes). This permits a very large number of programs to reside in primary memory, inasmuch as only one page of each program is actually located there (see Figure 7-3).

All other program pages are stored on a peripheral disk unit until they are ready for execution. Virtual storage provides a number of advantages. First, the central processor is utilized more fully. Many more programs can be in primary storage because only one page of each program actually resides there. Second, programmers no longer have to worry about the size of the primary storage area. With virtual storage, programs can be of infinite length and small machines can execute a program of any size (admittedly, small machines will take longer than big machines to execute a large program).

virtual storage A way of handling programs more efficiently by the computer by dividing the programs into small fixed- or variable-length portions with only a small portion stored in primary memory at one time.

page A small fixed-length section of a program, which can be easily stored in primary storage and quickly accessed from secondary storage.

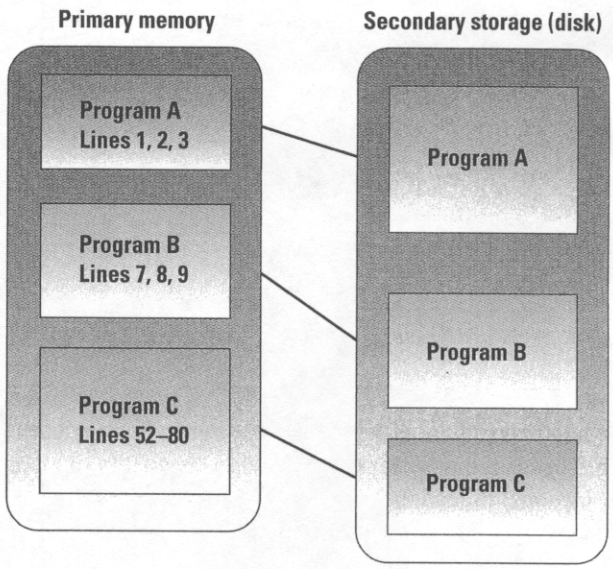


Figure 7-3 Virtual storage. Virtual storage is based on the fact that, in general, only a few statements in a program can actually be utilized at any given moment. In virtual storage, programs are broken down into small sections called pages. Individual program pages are read into memory only when needed. The rest of the program is stored on disk until it is required. In this way, very large programs can be executed by small machines, or a large number of programs can be executed concurrently by a single machine.

Time Sharing

time sharing The sharing of computer resources by many users simultaneously by having the CPU spend a fixed amount of time on each user's program before proceeding to the next.

Time sharing is an operating system capability that allows many users to share computer processing resources simultaneously. It differs from multiprogramming in that the CPU spends a fixed amount of time on one program before moving on to another. In a time-sharing environment, thousands of users are each allocated a tiny slice of computer time (2 milliseconds). In this time slot, each user is free to perform any required operations; at the end of this period, another user is given a 2-millisecond time slice of the CPU. This arrangement permits many users to be connected to a CPU simultaneously, with each receiving only a tiny amount of CPU time. But because the CPU is operating at the nanosecond level, a CPU can accomplish a great deal of work in 2 milliseconds.

Multiprocessing

multiprocessing An operating system feature for executing two or more instructions simultaneously in a single computer system by using multiple central processing units.

Multiprocessing is an operating system capability that links together two or more CPUs to work in parallel in a single computer system. The operating system can assign multiple CPUs to execute different instructions from the same program or from different programs simultaneously, dividing the work between the CPUs. Whereas multiprogramming uses concurrent processing with one CPU, multiprocessing uses simultaneous processing with multiple CPUs.

Language Translation and Utility Software

source code Program instructions written in a high-level language that must be translated into machine language to be executed by the computer.

compiler Special system software that translates a high-level language into machine language for execution by the computer.

object code Program instructions that have been translated into machine language so that they can be executed by the computer.

interpreter A special translator of source code into machine code that translates each source code statement into machine code and executes them, one at a time.

When computers execute programs written in languages such as COBOL, FORTRAN, or C, the computer must convert these human-readable instructions into a form it can understand. System software includes special language translator programs that translate high-level language programs written in programming languages such as BASIC, COBOL, and FORTRAN into machine language that the computer can execute. This type of system software is called a *compiler* or *interpreter.* The program in the high-level language before translation into machine language is called **source code.** A **compiler** translates source code into machine code called **object code.** Just before execution by the computer, the object code modules are joined with other object code modules in a process called *linkage editing.* The resulting load module is what is actually executed by the computer. Figure 7-4 illustrates the language translation process.

Some programming languages such as BASIC do not use a compiler but an **interpreter,** which translates each source code statement one at a time into machine code and executes it. Interpreter languages such as BASIC provide immediate feedback to the programmer if a mistake is made, but they are very slow to execute because they are translated one statement at a time.

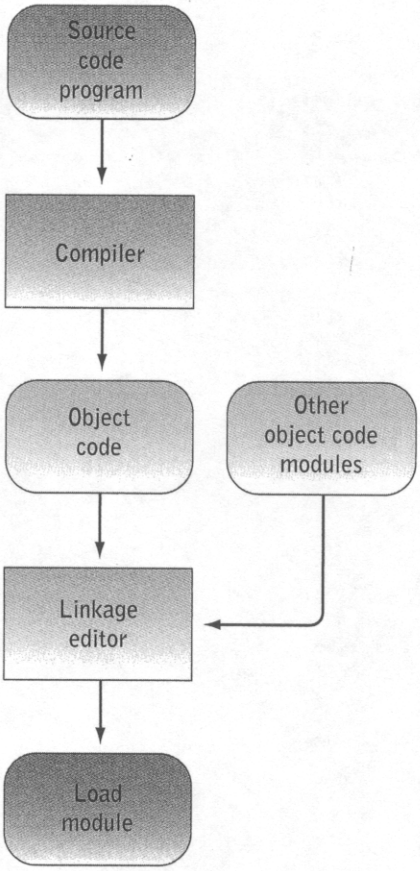


Figure 7-4 The language translation process. The source code, the program in a high-level language, is translated by the compiler into object code so that the instructions can be "understood" by the machine. These are grouped into modules. Prior to execution, the object code modules are joined together by the linkage editor to create the load module. It is the load module that is actually executed by the computer.

An assembler is similar to a compiler, but it is used to translate only assembly language (see Section 7.3) into machine code.

System software includes **utility programs** for routine, repetitive tasks, such as copying, clearing primary storage, computing a square root, or sorting. If you have worked on a computer and have performed such functions as setting up new files, deleting old files, or formatting diskettes, you have worked with utility programs. Utility programs are prewritten programs that are stored so that they can be shared by all users of a computer system and can be used rapidly in many different information system applications when requested.

utility program System software consisting of programs for routine, repetitive tasks, which can be shared by many users.

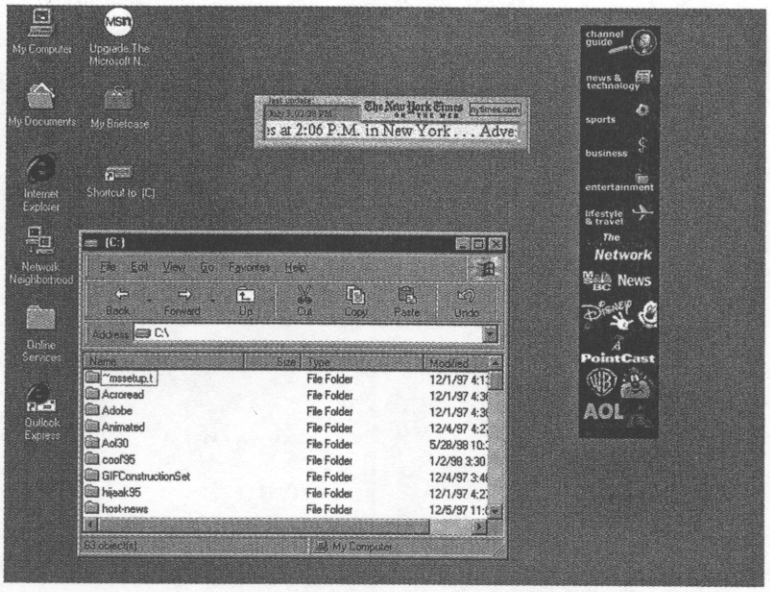
Graphical User Interfaces

When users interact with a computer, even a PC, the interaction is controlled by an operating system. The user interface is the part of an information system that users interact with. Users communicate with an operating system through the user interface of that operating system. Early PC operating systems were command-driven, but the **graphical user interface,** often called a **GUI,** makes extensive use of icons, buttons, bars, and boxes to perform the same task. It has become the dominant model for the user interface of PC operating systems and for many types of application software.

graphical user interface (GUI) The part of an operating system users interact with that uses graphic icons and the computer mouse to issue commands and make selections.

Older PC operating systems such as DOS, described in the following section, are command-driven, requiring the user to type in text-based commands using a keyboard. For example, to perform a task such as deleting a file named DATAFILE, the user must type in a command such as *DELETE C:\DATAFILE.* Users need to remember these commands and their syntax to work with the computer effectively. An operating system with a graphical user interface uses graphic symbols called *icons* to depict programs, files, and activities. Commands can be activated by rolling a mouse to move a cursor about the screen and clicking a button on the mouse to make selections. Icons are symbolic pictures and they are also used in GUIs to represent programs and files. For example, a file could be deleted by moving the cursor to a

Microsoft's Windows 98 is a powerful operating system with a graphical user interface and capabilities to integrate the user's desktop with the information resources of the Internet.



trash icon. Many graphical user interfaces use a system of pull-down menus to help users select commands and pop-up boxes to help users select among command options. Windowing features allow users to create, stack, size, and move around boxes of information.

Proponents of graphical user interfaces claim that they save learning time because computing novices do not have to learn different arcane commands for each application. Common functions such as getting help, saving files, or printing output are performed the same way. A complex series of commands can be issued simply by linking icons. On the other hand, GUIs may not always simplify complex tasks if the user has to spend too much time first pointing to icons and then selecting operations to perform on those icons (Morse and Reynolds, 1993). Graphic symbols themselves are not always easy to understand unless the GUI is well designed. Existing GUIs are modeled after an office desktop, with files, documents, and actions based on typical office behavior, making them less useful for nonoffice applications in control rooms or processing plants (Mandelkern, 1993). Users may be more productive if the interface is less generic and more customized to specific tasks (Satzinger and Olfman, 1998).

DOS Operating system for 16-bit PCs based on the IBM personal computer standard.

Windows A graphical user interface shell that runs in conjunction with the DOS PC operating system. Supports multitasking and some forms of networking.

Windows 98 Version of the Windows operating system that is more closely integrated with the Internet and that supports hardware technologies such as MMX, digital video disk, videoconferencing cameras, scanners, TV tuner-adapter cards, and joysticks.

Windows 95 A 32-bit operating system with a streamlined graphical user interface and multitasking, multithreading, and networking capabilities.

PC Operating Systems

Like any other software, PC software is based on specific operating systems and computer hardware. A software package written for one PC operating system generally cannot run on another. Table 7.1 compares the leading PC operating systems: Windows 98 and Windows 95, Windows NT, Windows CE, OS/2, UNIX, Linux, the Macintosh operating system, and DOS.

DOS was the most popular operating system for 16-bit PCs. It is used today with older PCs based on the IBM PC standard because so much available application software was written for systems using DOS. (PC-DOS is used exclusively with IBM PCs. MS-DOS, developed by Microsoft, is used with other 16-bit PCs that function like the IBM PC.) DOS itself does not support multitasking and limits the size of a program in memory to 640 K.

DOS is command-driven, but it can present a graphical user interface by using Microsoft **Windows,** a highly popular graphical user interface shell that runs in conjunction with the DOS operating system. Windows supports limited forms of multitasking and networking but shares the memory limitations of DOS. Early versions of Windows had some problems with application crashes when multiple programs competed for the same memory space.

Microsoft's **Windows 98** and **Windows 95** are genuine 32-bit operating systems. A 32-bit operating system can run faster than DOS, which could only address data in 16-bit chunks, because it can address data in 32-bit chunks. Both Windows 98 and Windows 95 provide a streamlined graphical user interface that arranges icons to provide instant access

Table 7.1 Leading PC Operating Systems

Operating System	Features
Windows 98 and Windows 95	32-bit operating system with a streamlined graphical user interface. Has multitasking and powerful networking capabilities and can be integrated with the information resources of the Web.
Windows NT (Windows 2000)	32-bit operating system for PCs, workstations, and network servers not limited to Intel microprocessors. Supports multitasking, multiprocessing, intensive networking.
Windows CE	Pared-down version of the Windows operating system for handheld computers and wireless communication devices.
OS/2	Operating system for IBM PCs that can take advantage of the 32-bit microprocessor. Supports multitasking and networking.
UNIX	Used for powerful PCs, workstations, and minicomputers. Supports multitasking, multi-user processing, and networking. Is portable to different models of computer hardware.
Linux	Free, reliable alternative to UNIX and Windows NT that runs on many different types of computer hardware and provides source code that can be modified by software developers.
Mac OS	Operating system for the Macintosh computer. Supports networking and multitasking and has powerful multimedia capabilities. Supports connecting to and publishing on the Internet.
DOS	Operating system for IBM (PC-DOS) and IBM-compatible (MS-DOS) PCs. Limits program use of memory to 640 K.

to common tasks. They can support software written for DOS but can also run programs that take up more than 640 K of memory. Windows 98 and 95 feature multitasking, multithreading (the ability to manage multiple independent tasks simultaneously), and powerful networking capabilities, including the capability to integrate fax, e-mail, and scheduling programs.

Windows 98 is faster and more integrated with the Internet than Windows 95; it includes support for new hardware technologies such as MMX, digital video disk (DVD—see Chapter 6), videoconferencing cameras, scanners, TV tuner-adapter cards, and joysticks. It provides capabilities for optimizing hardware performance and file management on the hard disk and enhanced 3-D graphics. The most visible feature of Windows 98 is the integration of the operating system with Web browser software. Users will be able to work with the traditional Windows interface or use the Web browser interface to display information. The user's hard disk can be viewed as an extension of the World Wide Web so that a document residing on the hard disk or on the Web can be accessed the same way. Small applet programs (see the discussion of Java in Section 7.4) on the Windows desktop can automatically retrieve information from specific Web sites whenever the user logs onto the Internet. These applets can automatically update the desktop with the latest news, stock quotes, or weather. Windows 98 also includes a group collaboration tool called NetMeeting (see Section 7.3) and Front Page Express, a tool for creating and storing Web pages.

Windows NT (for New Technology) is another 32-bit operating system developed by Microsoft with features that make it appropriate for applications in large networked organizations. It is used as an operating system for high-performance workstations and network servers. Windows NT shares the same graphical user interface as the other Windows operating systems, but it has more powerful networking, multitasking, and memory-management capabilities. Windows NT can support existing software written for DOS and Windows, and it can provide mainframe-like computing power for new applications with massive memory and file requirements. It can even support multiprocessing with multiple CPUs. Windows NT is not tied to computer hardware based on Intel microprocessors.

Windows NT Powerful operating system developed by Microsoft for use with 32-bit PCs and workstations based on Intel and other microprocessors. Supports networking, multitasking, and multiprocessing.

There are two versions of Windows NT—a Workstation version for users of standalone or client desktop computers and a Server version designed to run on network servers and provide network management functions. Windows NT Server includes tools for creating and operating Web sites. Microsoft renamed its recent release of Windows NT **Windows 2000.**

Windows 2000 Recent release of Windows NT for corporate computing.

Windows CE Portable and compact operating system designed to run on small hand-held computers, personal digital assistants, or wireless communication devices.

Windows CE has some of the capabilities of Windows, including its graphical user interface, but it is designed to run on small handheld computers, personal digital assistants, or wireless communication devices such as pagers and cellular phones. It is a portable and compact operating system requiring very little memory. Non-PC and consumer devices can use this operating system to share information with Windows-based PCs and to connect to the Internet.

OS/2 Powerful operating system used with 32-bit IBM/PCs or workstations that supports multitasking, networking, and more memory-intensive applications than DOS.

OS/2 is a robust 32-bit operating system for powerful IBM or IBM-compatible PCs with Intel microprocessors. OS/2 is used for complex, memory-intensive applications or those that require networking, multitasking, or large programs. OS/2 provides powerful desktop computers with mainframe-operating-system capabilities, such as multitasking and supporting multiple users in networks, and it supports networked multimedia and pen computing applications.

OS/2 supports applications that run under Windows and DOS and has its own graphical user interface. There are now two versions of OS/2. OS/2 Warp is for personal use. It can accept voice-input commands and run Java applications without a Web browser (see Sections 7.3 and 7.4). OS/2 Warp Server has capabilities similar to Windows NT for supporting networking, systems management, and Internet access.

UNIX Operating system for PCs, minicomputers, and mainframes, which is machine independent and supports multi-user processing, multitasking, and networking.

UNIX is an interactive, multi-user, multitasking operating system developed by Bell Laboratories in 1969 to help scientific researchers share data. Many people can use UNIX simultaneously to perform the same kind of task, or one user can run many tasks on UNIX concurrently. UNIX was developed to connect various machines together and is highly supportive of communications and networking. UNIX was designed for minicomputers but now has versions for PCs, workstations, and mainframes. It is often used on workstations and server computers. UNIX can run on many different kinds of computers and can be easily customized. Application programs that run under UNIX can be ported from one computer to run on a different computer with little modification. UNIX also can store and manage a large number of files.

UNIX is considered powerful but very complex, with a legion of commands. Graphical user interfaces have been developed for UNIX. UNIX cannot respond well to problems caused by the overuse of system resources such as jobs or disk space. UNIX also poses some security problems because multiple jobs and users can access the same file simultaneously. Vendors have developed different versions of UNIX that are incompatible, thereby limiting software portability.

Linux Reliable and compactly designed operating system that is an offshoot of UNIX which can run on many different hardware platforms and is available free or at very low cost. Used as alternative to UNIX and Windows NT.

Linux is a UNIX-like operating system that runs on Intel, Motorola, Digital Alpha, SPARC, and Mips processors. Linux can be downloaded from the Internet free of charge or purchased for a small fee from companies that provide additional tools for the software. Because it is free, reliable, compactly designed, and capable of running on many different hardware platforms, it has become popular during the past few years among sophisticated computer users and businesses as an alternative to UNIX and Windows NT. Major application software vendors are starting to provide versions that can run on Linux. The source code for Linux is available along with the operating system software, so that it can be modified by software developers to fit their particular needs.

Open-source software Software that provides free access to its program code, allowing users to modify the program code to make improvements or fix errors.

Linux is an example of **open-source software,** which provides all computer users with free access to its source code so that they can modify the code to fix errors or to make improvements. Open-source software such as Linux is not owned by any company or individual. A global network of programmers and users manages and modifies the software, usually without being paid to do so. The Window on Organizations describes how organizations are starting to benefit from this new operating system.

Mac OS Operating system for the Macintosh computer that supports multitasking, has access to the Internet, and has powerful graphics and multimedia capabilities.

Mac OS, the operating system for the Macintosh computer, features multitasking, powerful multimedia and networking capabilities, and a mouse-driven graphical user interface. New features of this operating system allow users to connect to, explore, and publish on the Internet and World Wide Web; use Java software (see Section 7.4); and load Chinese, Japanese, Korean, Indian, Hebrew, and Arabic fonts for use in Web browser software (see Section 7.3). A new search capability called Sherlock provides a standard interface for efficiently searching for files on the Internet as well as on the user's own hard drive.

Should Businesses Switch to Linux?

Burlington Coat Factory, the $1.8 billion clothing discounter based in Burlington, New Jersey, decided to take the plunge with Linux and is installing this new operating system on 1150 computers in its 250 stores. Why would such a large company opt for a new shareware operating system that can be downloaded free from the Internet?

According to Mike Prince, Burlington's CIO, Linux was attractive both for its price and its performance. It's free and "runs like the wind." Prince also believes Linux is more stable than Windows NT, and will be less costly to support. Burlington is known as a company that has been comfortable embracing new technology, including network computers and Java as well as Linux. The company also has used UNIX for many years and was using Linux on development workstations for about a year before installing it in its stores.

Burlington's previous in-store systems were based on aging technology—Sun Microsystems' SPARC workstations running the SunOS 4.1 operating system. Its client computers for back-office and inventory applications were either radio-frequency handheld scanners or dumb terminals. Burlington's point-of-sale system, which will not change, uses old PCs running MS-DOS. Prince is replacing the dumb terminals with Pentium PCs but hasn't made up his mind about whether to scrap the SPARC workstations entirely or install Linux on them. When Burlington completes its upgrade, the new hardware should cost between $1.15 million and $1.8 million, but the cost of Linux will only be a few hundred dollars. Burlington expects to save thousands of dollars in each store by not buying a commercial operating system.

The low cost, fast performance, and reliability of Linux also made it attractive to Jay Jacobs, Inc., another retailer based in Seattle, which is installing Linux servers in all of its 120 stores. The Linux servers will be tracking purchases by customer as well as by item. Bill Lawrence, the firm's Chief Financial Officer, thinks that Linux will provide fast, UNIX-like performance for less cost than the slower Windows NT environment. By using Linux instead of another operating system, the company is saving $666 per store, amounting to $80,000.

On the other hand, both Burlington and Jay Jacobs are not relying solely on Linux. Jay Jacobs is using the more established UNIX and Windows NT operating systems at its corporate headquarters. Burlington is keeping Windows NT for desktop productivity applications such as Microsoft Excel and Word that aren't available on Linux. Burlington is sticking with UNIX servers from Sequent Computer Systems to house and manipulate its corporate data. Other retailers have primarily selected Windows NT when upgrading the operating systems for their stores.

Until more kinds of applications are developed for Linux, this operating system is being used primarily on specialized departmental servers providing Web, e-mail, or printing services or to run custom applications that only require a simple interface. Retailers such as Burlington, which run very few third-party applications, are in a stronger position to select more obscure software platforms. Businesses are also waiting for computer hardware vendors to provide more services and software so that Linux can run easily on their machines.

Burlington is using Red Hat Software's version of Linux, an inexpensive commercial version available on CD-ROMs that offers technical support. Red Hat's version is the market leader but it represents only one of a number of different versions of Linux that are currently in use. Since Linux has no single owner, the software is updated by a large group of programmers around the globe. Unlike Windows, which is controlled by Microsoft, anyone can find errors and make changes to Linux code, raising the danger that Linux could splinter into many slightly different versions as did UNIX. The Linux Standards Base is working on rules to keep different Linux versions compatible. Different versions of Linux would discourage its widespread adaption in business.

To Think About: Should a company select Linux as its operating system for its major business applications? What management, organization, and technology factors would have to be addressed when making that decision?

Sources: David Orenstein, "Burlington Commits to Linux in 250 Stores," Computerworld, February 15, 1999 and "Retailer Sets Big on Linux," Computerworld, February 8, 1999; and Ray Lane, "Standardizing Linux," The Industry Standard, March 8, 1999.

7.3 Application Software

Application software is primarily concerned with accomplishing the tasks of end users. Many different languages can be used to develop application software. Each has different strengths and drawbacks.

Generations of Programming Languages

To communicate with the first generation of computers, specialized programmers wrote programs in **machine language**—the 0s and 1s of binary code. Programming in 0s and 1s (reducing all statements such as add, subtract, and divide into a series of 0s and 1s) made early programming a slow, labor-intensive process.

machine language A programming language consisting of the 1s and 0s of binary code.

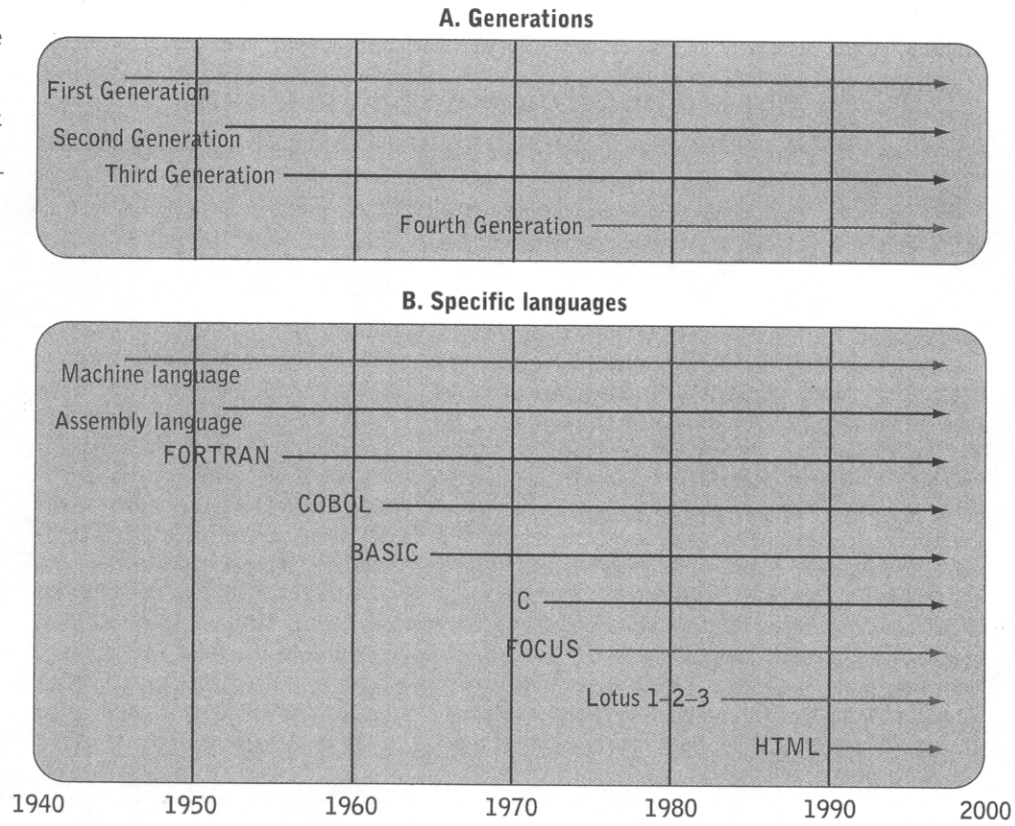


Figure 7-5 Generations of programming languages. As the capabilities of hardware increased, programming languages developed from the first generation of machine and second generation of assembly languages of the 1950s to 1960s, through the third-generation, high-level languages such as FORTRAN and COBOL developed in the 1960s and 1970s, to today's fourth-generation languages and tools.

As computer hardware improved and processing speed and memory size increased, computer languages changed from machine language to languages that were easier for humans to understand. Generations of programming languages developed to correspond with the generations of computer hardware. Figure 7-5 shows the development of programming languages during the past 50 years as the capabilities of hardware have increased. The major trend is to increase the ease with which users can interact with hardware and software.

Machine language was the first-generation programming language. The second generation of programming languages occurred in the early 1950s with the development of assembly language. Instead of using 0s and 1s, programmers could substitute language-like acronyms and words such as *add, sub* (subtract), and *load* in programming statements. A language translator called a *compiler* converted the English-like statements into machine language.

From the mid-1950s to the mid-1970s, the third generation of programming languages emerged. These languages, such as FORTRAN, COBOL, and BASIC, allowed programs to be written with regular words using sentence-like statements. These languages are called **high-level languages** because each statement generates multiple statements when it is translated into machine language. Programs became easier to create and became more widely used for scientific and business problems.

high-level language Programming languages in which each source code statement generates multiple statements at the machine-language level.

Beginning in the late 1970s, fourth-generation languages and tools were created. These languages dramatically reduced programming time and made software tasks so easy that many could be performed by nontechnical computer users without the help of professional programmers. Software such as word processing, spreadsheets, data management, and Web browsers became popular productivity tools for end users.

Popular Programming Languages

Most managers need not be expert programmers, but they should understand how to evaluate software applications and be able to select programming languages that are appropriate for their organization's objectives. We will now briefly describe the more popular high-level languages.

```
AR 5, 3
```

Figure 7-6 Assembly language. This sample assembly language command adds the contents of register 3 to register 5 and stores the result in register 5.

```
READ (5,100) ID, QUANT, PRICE
TOTAL = QUANT * PRICE
```

Figure 7-7 FORTRAN. This sample FORTRAN program code is part of a program to compute sales figures for a particular item.

```
MULTIPLY QUANT-SOLD BY UNIT-PRICE GIVING SALES-TOTAL.
```

Figure 7-8 COBOL. This sample COBOL program code is part of a routine to compute total sales figures for a particular item.

Assembly Language

Like machine language, **assembly language** (Figure 7-6) is designed for a specific machine and specific microprocessors. Each operation in assembly corresponds to a machine operation. Assembly language makes use of certain mnemonics (e.g., *load, sum*) and assigns addresses and storage locations automatically. Although assembly language gives programmers great control, it is costly in terms of programmer time; it is also difficult to read, debug, and learn. Assembly language is used primarily today in system software.

assembly language A programming language developed in the 1950s that resembles machine language but substitutes mnemonics for numeric codes.

FORTRAN

FORTRAN (FORmula TRANslator) (Figure 7-7) was developed in 1956 to provide an easier way of writing scientific and engineering applications. FORTRAN is especially useful in processing numeric data. Many kinds of business applications can be written in FORTRAN, and contemporary versions provide sophisticated structures for controlling program logic. FORTRAN is not very good at providing input/output efficiency or in printing and working with lists. The syntax is very strict and keying errors are common, making the programs difficult to debug.

FORTRAN (FORmula TRANslator) A programming language developed in 1956 for scientific and mathematical applications.

COBOL

COBOL (COmmon Business Oriented Language) (Figure 7-8) came into use in the early 1960s. It was developed by a committee representing both government and industry. Rear Admiral Grace M. Hopper was a key committee member who played a major role in COBOL development. COBOL was designed with business administration in mind, for processing large data files with alphanumeric characters (mixed alphabetic and numeric data), and for performing repetitive tasks such as payroll. It is poor at complex mathematical calculations. Also, there are many versions of COBOL, and not all are compatible with each other.

COBOL (COmmon Business Oriented Language) Major programming language for business applications because it can process large data files with alphanumeric characters.

BASIC

BASIC (Beginners All-purpose Symbolic Instruction Code) was developed in 1964 by John Kemeny and Thomas Kurtz to teach students at Dartmouth College how to use computers. Today it is a popular programming language on college campuses and for PCs. BASIC can do

BASIC (Beginners All-purpose Symbolic Instruction Code) A general-purpose programming language used with PCs and for teaching programming.

almost all computer processing tasks from inventory to mathematical calculations. It is easy to use, demonstrates computer capabilities well, and requires only a small interpreter. The weakness of BASIC is that it does few tasks well even though it does them all. It has no sophisticated program logic control or data structures, which makes it difficult to use in teaching good programming practices. Different versions of BASIC exist.

Pascal

Pascal A programming language used on PCs and used to teach sound programming practices in computer science courses.

Named after Blaise Pascal, the seventeenth-century mathematician and philosopher, **Pascal** was developed by the Swiss computer science professor Niklaus Wirth of Zurich in the late 1960s. Pascal programs can be compiled using minimal computer memory, so they can be used on PCs. With sophisticated structures to control program logic and a simple, powerful set of commands, Pascal is used primarily in computer science courses to teach sound programming practices. The language is weak at file handling and input/output and is not easy for beginners to use.

C and C++

C A powerful programming language with tight control and efficiency of execution; is portable across different microprocessors and is used primarily with PCs.

C++ Object-oriented version of the C programming language.

C is a powerful and efficient language developed at AT&T's Bell Labs in the early 1970s. It combines machine portability with tight control and efficient use of computer resources, and it can work on a variety of different computers. It is used primarily by professional programmers to create operating system and application software, especially for PCs.

C++ is a newer version of C that is object-oriented (see Section 7.4). It has all the capabilities of C plus additional features for working with software objects. C++ is used for developing application software.

Other Programming Languages

Other important programming languages include Ada, LISP, Prolog, and PL/1.

Ada A programming language that is portable across different brands of hardware; is used for both military and nonmilitary applications.

LISP Programming language used for artificial-intelligence applications. Stands for LISt Processor.

Prolog Programming language for artificial-intelligence applications.

PL/1 (Programming Language 1) A programming language developed by IBM for both business and scientific applications.

- **Ada** was developed in 1980 by the U.S. Defense Department to serve as a standard for all of its applications. Named after Ada, Countess of Lovelace, a nineteenth-century mathematician, it was designed to be executed in diverse hardware environments. Ada is used for both military and nonmilitary applications because it can operate on different brands of computer hardware.
- **LISP** (designating LISt Processor) and **Prolog** (designating PROgramming LOGic) are used for artificial-intelligence applications. LISP, created in the late 1950s, is oriented toward putting symbols such as operations, variables, and data values into meaningful lists. Prolog was introduced about 1970 and also is well-suited to manipulating symbols. It can run on a wider variety of computers than LISP.
- **PL/1 (Programming Language 1)** is a powerful general-purpose programming language developed by IBM in 1964. It can comfortably handle both mathematical and business problems, but it has not replaced COBOL or FORTRAN because organizations have already invested so heavily in COBOL and FORTRAN systems.

Fourth-Generation Languages and PC Software Tools

fourth-generation language A programming language that can be employed directly by end users or less-skilled programmers to develop computer applications more rapidly than conventional programming languages.

Fourth-generation languages consist of a variety of software tools that enable end users to develop software applications with minimal or no technical assistance or that enhance the productivity of professional programmers. Fourth-generation languages tend to be nonprocedural or less procedural than conventional programming languages. Procedural languages require specification of the sequence of steps, or procedures, that tell the computer what to do and how to do it. Nonprocedural languages need only specify what has to be accomplished rather than provide details about how to carry out the task. Thus, a nonprocedural language can accomplish the same task with fewer steps and lines of program code than a procedural language.

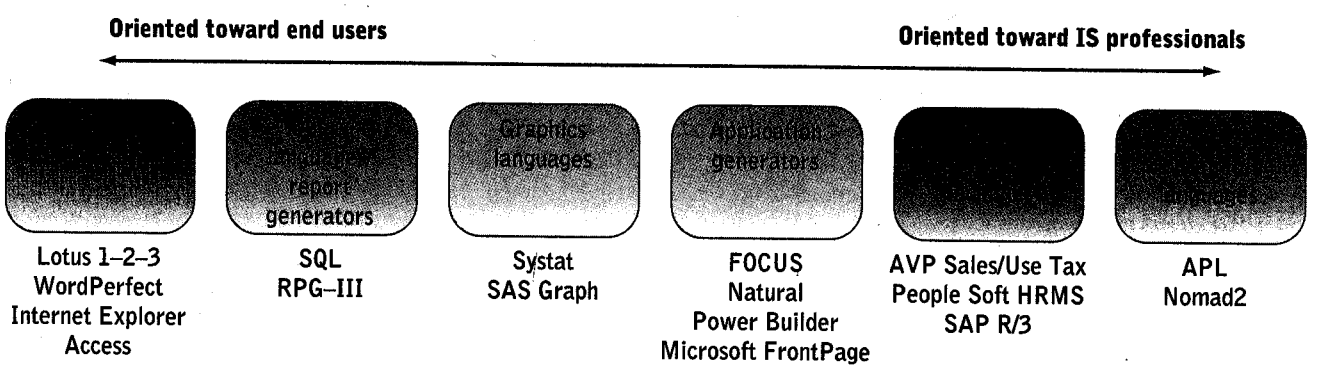


Figure 7-9 Fourth-generation languages. The spectrum of major categories of fourth-generation languages; commercially available products in each category are illustrated. Tools range from those that are simple and designated primarily for end users to complex tools designed for information systems professionals.

There are seven categories of fourth-generation languages: query languages, report generators, graphics languages, application generators, very high-level programming languages, application software packages, and PC tools. Figure 7-9 illustrates the spectrum of these tools and some commercially available products in each category.

Query Languages

Query languages are high-level languages for retrieving data stored in databases or files. They are usually interactive, on-line, and capable of supporting requests for information that are not predefined. They are often tied to database management systems (see Chapter 8) or some of the PC software tools described later in this section. For instance, the query

query language A high-level computer language used to retrieve specific information from databases or files.

SELECT ALL WHERE age >40 AND name = "Wilson"

requests all records where the name is "Wilson" and the age is more than 40. Chapter 8 provides more detail on Structured Query Language (SQL), which has become a standard query language.

Available query tools have different kinds of syntax and structure, some being closer to natural language than others (Vassiliou, 1984–85). **Natural language** software allows users to communicate with the computer using conversational commands that resemble human speech. Natural language development is one of the concerns of artificial intelligence (see Chapter 14). Some consider the movement toward natural language as the next generation in software development.

natural language Programming language that is very close to human language.

Report Generators

Report generators are facilities for creating customized reports. They extract data from files or databases and create reports in many formats. Report generators generally provide more control over the way data are formatted, organized, and displayed than query languages. The more powerful report generators can manipulate data with complex calculations and logic before they are output. Some report generators are extensions of database or query languages.

report generator Software that creates customized reports in a wide range of formats that are not routinely produced by an information system.

Graphics Languages

Graphics languages retrieve data from files or databases and display them in graphic format. Users can ask for data and specify how they are to be charted. Some graphics software can perform arithmetic or logical operations on data as well. SAS and Systat are examples of powerful analytical graphics software.

graphics language A computer language that displays data from files or databases in graphic format.

application generator Software that can generate entire information system applications; the user needs only to specify what needs to be done, and the application generator creates the appropriate program code.

Application Generators

Application generators contain preprogrammed modules that can generate entire applications, greatly speeding development. A user can specify what needs to be done, and the application generator will create the appropriate code for input, validation, update, processing, and reporting. Most full-function application generators consist of a comprehensive, integrated set of development tools: a database management system, data dictionary, query language, screen painter, graphics generator, report generator, decision support/modeling tools, security facilities, and a high-level programming language. Application generators now include tools for developing full-function Web sites.

very high-level programming language A programming language that uses fewer instructions than conventional languages. Used primarily as a professional programmer productivity tool.

Very High-Level Programming Languages

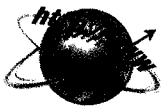
Very high-level programming languages are designed to generate program code with fewer instructions than conventional languages such as COBOL or FORTRAN. Programs and applications based on these languages can be developed in much shorter periods of time. Simple features of these languages can be employed by end users. However, these languages are designed primarily as productivity tools for professional programmers. APL and Nomad2 are examples of these languages.

software package A prewritten, precoded, commercially available set of programs that eliminates the need to write software programs for certain functions.

Application Software Packages

A **software package** is a prewritten, precoded, commercially available set of programs that eliminates the need for individuals or organizations to write their own software programs for certain functions. There are software packages for system software, but the vast majority of package software is application software.

Application software packages consist of prewritten application software that is marketed commercially. These packages are available for major business applications on mainframes, minicomputers, and PCs. Although application packages for large complex systems must be installed by technical specialists, many application packages, especially those for PCs, are marketed directly to end users. Systems development based on application packages is discussed in Chapter 12.



PC Software Tools

Some of the most popular and productivity-promoting software tools are the general-purpose application packages that have been developed for PCs, especially word processing, spreadsheet, data management, presentation graphics, integrated software packages, e-mail, Web browsers, and groupware.

word processing software Software that handles electronic storage, editing, formatting, and printing of documents.

Word processing software. **Word processing software** stores text data electronically as a computer file rather than on paper. The word processing software allows the user to make changes in the document electronically in memory. This eliminates the need to retype an entire page to incorporate corrections. The software has formatting options to make changes in line spacing, margins, character size, and column width. Microsoft Word and WordPerfect are popular word processing packages. Figure 7-10 illustrates a Microsoft Word screen displaying text, spelling and grammar checking, and major menu options.

Most word processing software has advanced features that automate other writing tasks: spelling checkers, style checkers (to analyze grammar and punctuation), thesaurus programs, and mail merge programs, which link letters or other text documents with names and addresses in a mailing list. The newest versions of this software can create and access Web pages.

spreadsheet Software displaying data in a grid of columns and rows, with the capability of easily recalculating numerical data.

Spreadsheets. Electronic **spreadsheet** software provides computerized versions of traditional financial modeling tools such as the accountant's columnar pad, pencil, and calculator. An electronic spreadsheet is organized into a grid of columns and rows. The power of the electronic spreadsheet is evident when one changes a value or values because all other related values on the spreadsheet will be automatically recomputed.

Spreadsheets are valuable for applications in which numerous calculations with pieces of data must be related to each other. Spreadsheets also are useful for applications that require

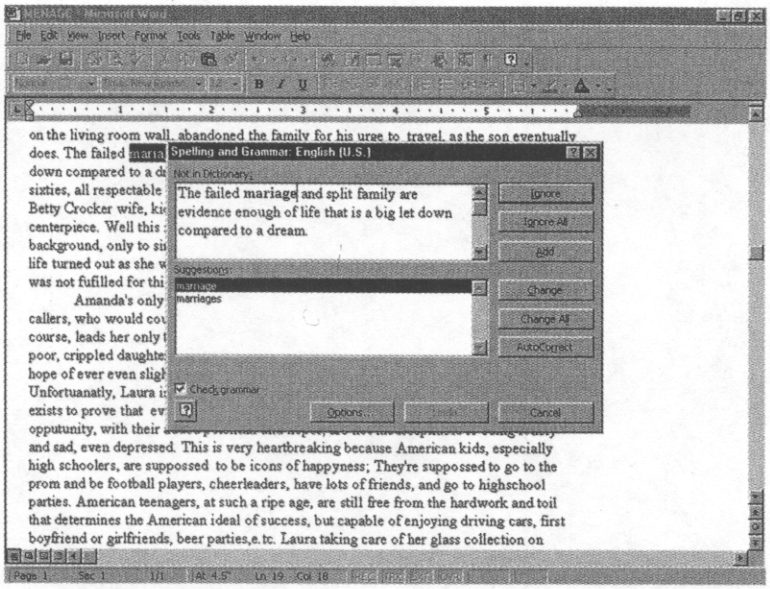


Figure 7-10 Text and the spell-checking option in Microsoft Word. Word processing software provides many easy-to-use options to create and output a text document to meet a user's specifications. Source: Courtesy of Microsoft.

modeling and what-if analysis. After the user has constructed a set of mathematical relationships, the spreadsheet can be recalculated instantaneously using a different set of assumptions. A number of alternatives can easily be evaluated by changing one or two pieces of data without having to rekey in the rest of the worksheet. Many spreadsheet packages include graphics functions that can present data in the form of line graphs, bar graphs, or pie charts. The most popular spreadsheet packages are Microsoft Excel and Lotus 1-2-3. The newest versions of this software can read and write Web files.

Figure 7-11 illustrates the output from a spreadsheet for a breakeven analysis and its accompanying graph.

Data management software. Although spreadsheet programs are powerful tools for manipulating quantitative data, **data management software** is more suitable for creating and manipulating lists and for combining information from different files. PC database management packages have programming features and easy-to-learn menus that enable nonspecialists to build small information systems.

data management software Software used for creating and manipulating lists, creating files and databases to store data, and combining information for reports.

Data management software typically has facilities for creating files and databases and for storing, modifying, and manipulating data for reports and queries. A detailed treatment of data management software and database management systems can be found in Chapter 8. Popular database management software for the personal computer includes Microsoft Access, which has been enhanced to publish data on the Web. Figure 7-12 shows a screen from Microsoft Access illustrating some of its capabilities.

Presentation graphics. Presentation graphics software allows users to create professional-quality graphics presentations. This software can convert numeric data into charts and other types of graphics and can include multimedia displays of sound, animation, photos, and video clips. The leading presentation graphics packages include capabilities for computer-generated slide shows and translating content for the Web. Microsoft PowerPoint, Lotus Freelance Graphics, and Aldus Persuasion are popular presentation graphics packages.

presentation graphics Software to create professional-quality graphics presentations that can incorporate charts, sound, animation, photos, and video clips.

Integrated software packages and software suites. Integrated software packages combine the functions of the most important PC software packages, such as word processing, spreadsheets, presentation graphics, and data management. This integration provides a more general-purpose software tool and eliminates redundant data entry and data maintenance. For example, the breakeven analysis spreadsheet illustrated in Figure 7-11 could

integrated software package A software package that provides two or more applications, such as word processing and spreadsheets, providing for easy transfer of data between them.

Total fixed cost	19,000.00
Variable cost per unit	3.00
Average sales price	17.00
Contribution margin	14.00
Breakeven point	1,357

Custom Neckties Pro Forma Income Statement

Units sold	0.00	679	1,357	2,036	2,714
Revenue	0	11,536	23,071	34,607	46,143
Fixed cost	19,000	19,000	19,000	19,000	19,000
Variable cost	0	2,036	4,071	6,107	8,143
Total cost	19,000	21,036	23,071	25,107	27,143
Profit/Loss	(19,000)	(9,500)	0	9,500	19,000

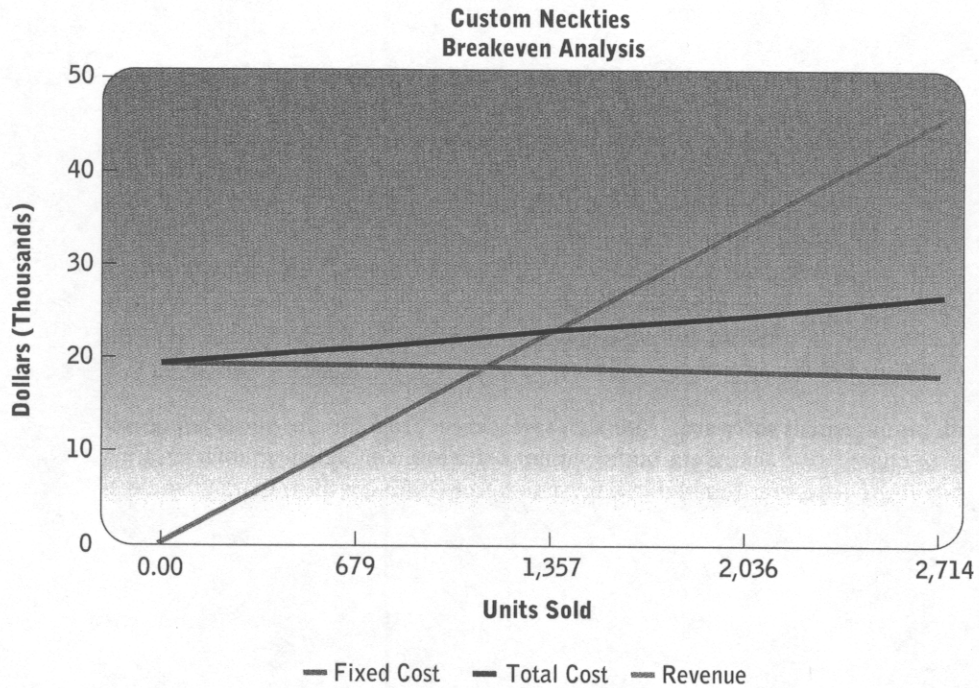


Figure 7-11 Spreadsheet software. Spreadsheet software organizes data into columns and rows for analysis and manipulation. Contemporary spreadsheet software provides graphing abilities for clear visual representation of the data in the spreadsheets. This sample breakeven analysis is represented as numbers in a spreadsheet as well as a line graph for easy interpretation.

be reformatted into a polished report with word processing software without separately keying the data into both programs. Integrated packages are a compromise. Although they can do many things well, they generally do not have the same power and depth as single-purpose packages.

Integrated software packages should be distinguished from software suites, which are collections of applications software sold as a unit. Microsoft Office is an example. This software suite contains Word word processing software, Excel spreadsheet software, Access database software, PowerPoint presentation graphics software, and Outlook, a set of tools for e-mail, scheduling, and contact management. Office 2000 contains additional capabilities to support collaborative work on the Web, including in-line discussions about documents and the ability to automatically notify others about changes to documents. Documents created with Office tools can be viewed with a Web browser and published on a Web server. Software suites have some features of integrated packages, such as the ability to

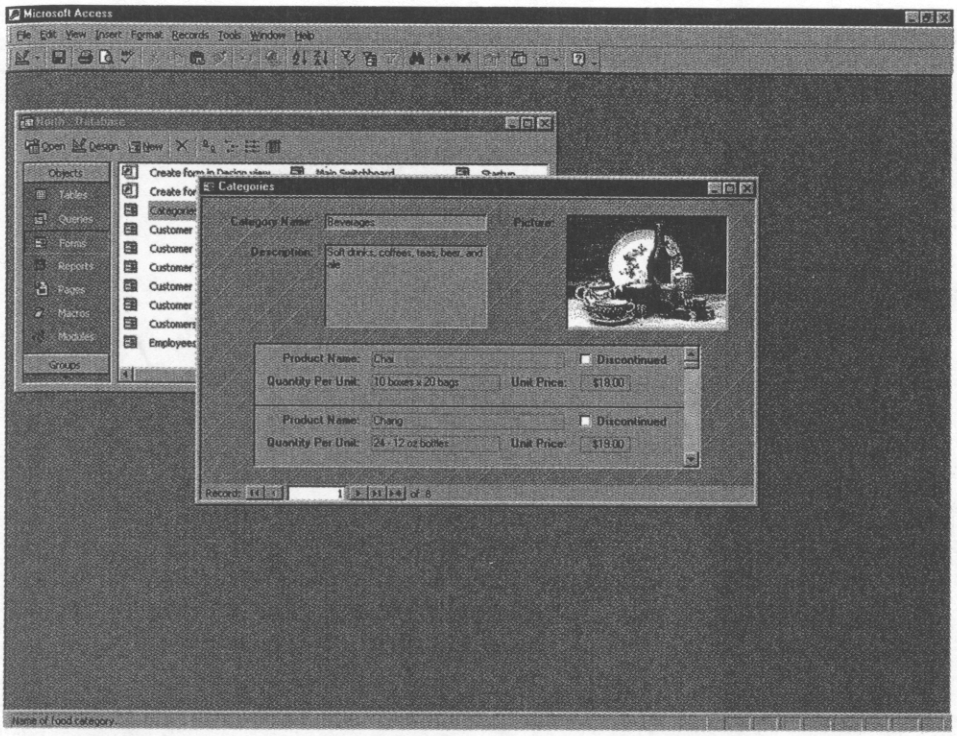


Figure 7-12 Data management software. This screen from Microsoft Access illustrates some of its powerful capabilities for managing and organizing information.

share data among different applications, but they consist of full-featured versions of each type of software.

E-mail software. Electronic mail (e-mail) is used for the computer-to-computer exchange of messages and is an important tool for communication and collaborative work. A person can use a networked computer to send notes or lengthier documents to a recipient on the same network or a different network. Many organizations operate their own electronic-mail systems, but communications companies such as MCI and AT&T offer these services, along with commercial on-line information services such as America Online and Prodigy and public networks on the Internet.

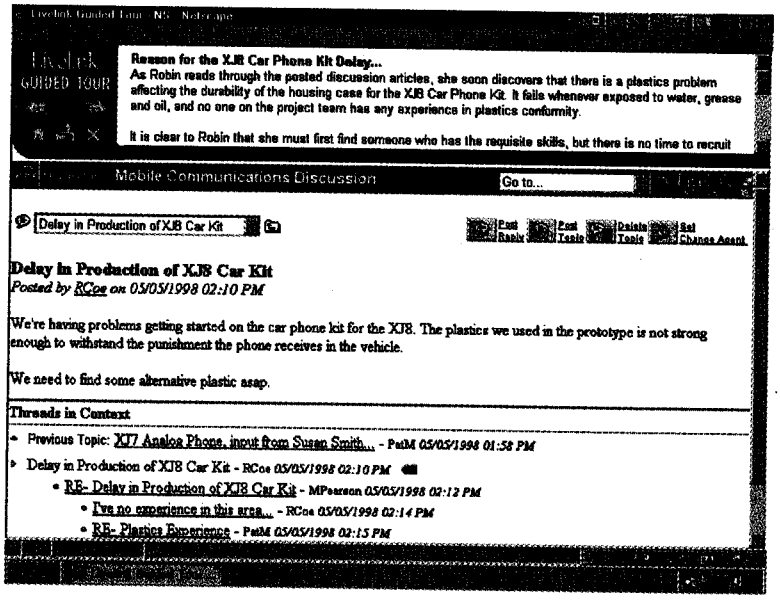
electronic mail (e-mail) The computer-to-computer exchange of messages.

Web browsers and the PC software suites have e-mail capabilities, but specialized e-mail software packages such as Eudora are also available for use on the Internet. In addition to providing electronic messaging, many e-mail software packages have capabilities for routing messages to multiple recipients, message forwarding, and attaching text documents or multimedia to messages.

Web browsers. Web browsers are easy-to-use software tools for displaying Web pages and for accessing the Web and other Internet resources. Web browser software features a point-and-click graphical user interface that can be employed throughout the Internet to access and display information stored on computers at other Internet sites. Browsers can display or present graphics, audio, and video information as well as traditional text, and they allow you to click on-screen buttons or highlighted words to link to related Web sites. Web browsers have become the primary interface for accessing the Internet or for using networked systems based on Internet technology. You can see examples of Web browser software by looking at the illustrations of Web pages in each chapter of this text.

Web browser An easy-to-use software tool for accessing the World Wide Web and the Internet.

The two leading commercial Web browsers are Microsoft's Internet Explorer and Netscape Navigator, which is also available as part of the Netscape Communicator software suite. They include capabilities for using e-mail, file transfer, on-line discussion groups and bulletin boards, along with other Internet services. Newer versions of these browsers contain support for Web publishing and workgroup computing. (See the following discussion of groupware.)



Groupware facilitates collaboration by enabling members of a group to share documents, schedule meetings, and discuss activities, events, and issues. Illustrated here are capabilities for following a threaded discussion.

groupware Software that provides functions and services that support the collaborative activities of work groups.

Groupware. Groupware provides functions and services to support the collaborative activities of work groups. Groupware includes software for information-sharing, electronic meetings, scheduling, and e-mail and a network to connect the members of the group as they work on their own desktop computers, often in widely scattered locations. Table 7.2 describes the capabilities of groupware.

thread A series of messages in on-line discussions on a specified topic that have been posted as replies to each other. Each message in a thread can be read to see how a discussion has evolved.

Groupware enhances collaboration by allowing the exchange of ideas electronically. All the messages on a topic can be saved in a group, stamped with the date, time, and author. All of these messages can be followed in a **thread** to see how a discussion has evolved. (A thread is a series of messages in an on-line discussion that have been posted as replies to each other.) Any group member can review the ideas of others at any time and add to them, or individuals can post a document for others to comment on or edit. Members can post requests for help, allowing others to respond. Finally, if a group so chooses, members can store their work notes on the groupware so that all others in the group can see what progress is being made, what problems occur, and what activities are planned.

The leading commercial groupware product has been Lotus Notes from the Lotus Development Corporation. The Internet is rich in capabilities to support collaborative work. Recent versions of Microsoft Internet Explorer and Netscape Communicator include groupware functions, such as e-mail, electronic scheduling and calendaring, audio and data conferencing, and electronic discussion groups and databases (see Chapters 10 and 14). Microsoft's Office 2000 software suite includes groupware features using Web technology. Powerful Web-based groupware features can also be found in products such as Opentext's Livelink.

Table 7.2 Groupware Capabilities

Group writing and commenting
Electronic mail distribution
Scheduling meetings and appointments
Shared files and databases
Shared time lines and plans
Electronic meetings and conferences