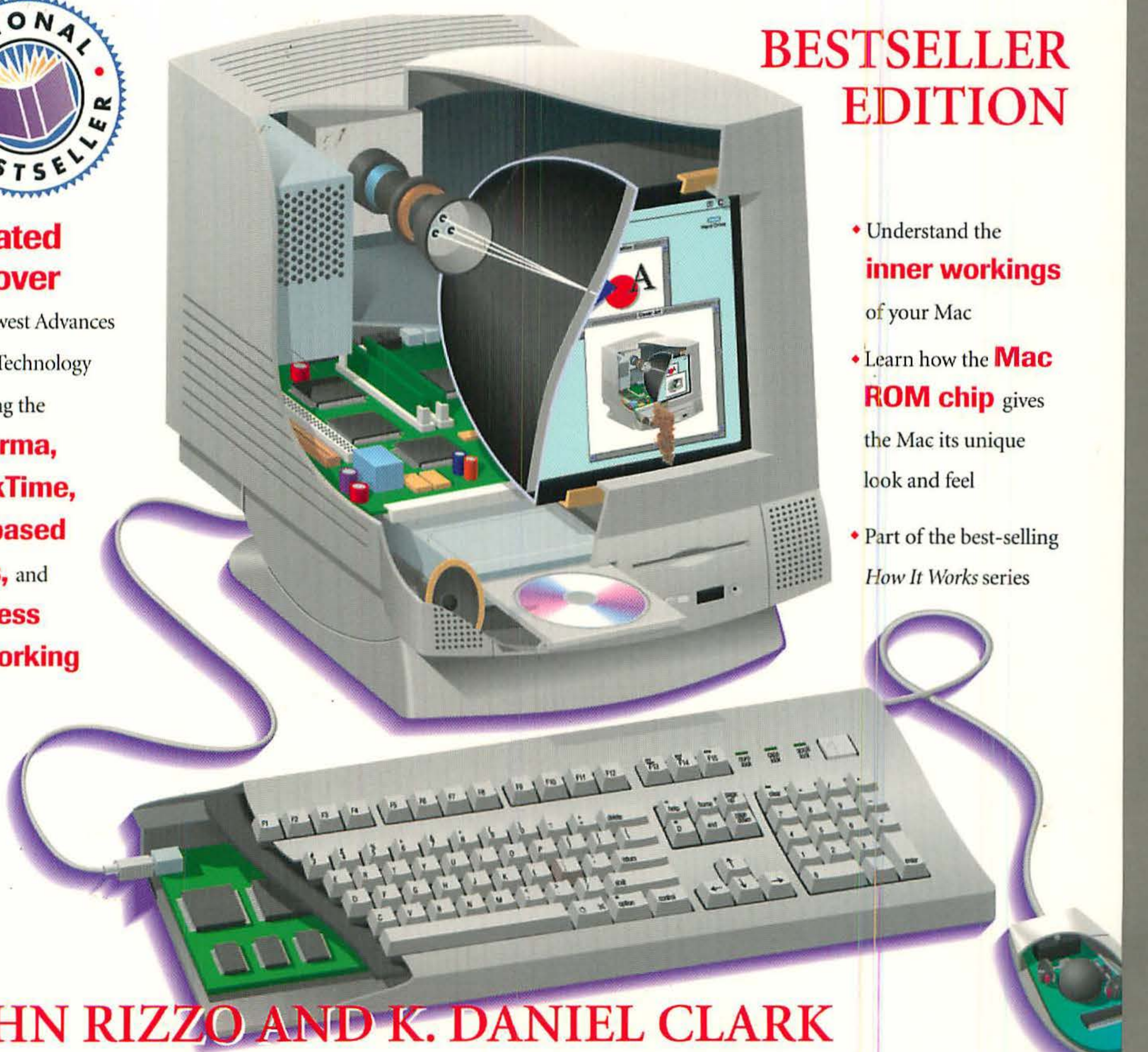
HOW MACS WORK

NATIONAL BESTSELLER

BESTSELLER EDITION



Updated to Cover

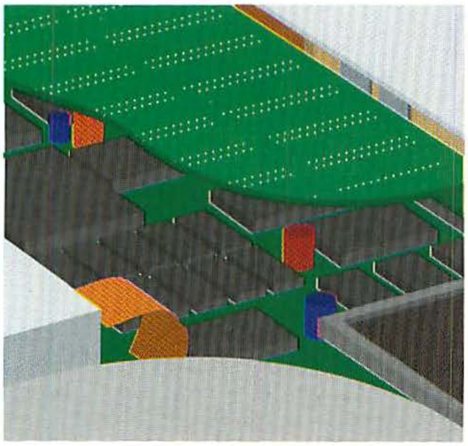
The Newest Advances in Mac Technology Including the **Performa, QuickTime, PCI-based Macs,** and **Wireless Networking**

- Understand the **inner workings** of your Mac
- Learn how the **Mac ROM chip** gives the Mac its unique look and feel
- Part of the best-selling *How It Works* series

JOHN RIZZO AND K. DANIEL CLARK

How Macs Work

Bestseller Edition



How Macs Work

Bestseller Edition

John Rizzo and K. Daniel Clark

Ziff-Davis Press
An imprint of Macmillan Computer Publishing USA
Emeryville, California

Senior Development Editor	Valerie Haynes Perry
Copy Editor	Nicole Clausing
Technical Reviewer	Paul R. Freedman
Project Coordinator	Barbara Dahl
Cover Design	Regan Honda
Cover Illustration	K. Daniel Clark
Book Design	Bruce Lundquist
Technical Illustration	K. Daniel Clark
Word Processing	Howard Blechman
Page Layout	M.D. Barrera
Indexer	Carol Burbo

Ziff-Davis Press, ZD Press, and the Ziff-Davis Press logo are trademarks of, and are licensed to Macmillan Computer Publishing USA by Ziff-Davis Publishing Company, New York, New York.

Ziff-Davis Press imprint books are produced on a Macintosh computer system with the following applications: FrameMaker®, Microsoft® Word, QuarkXPress®, Adobe Illustrator®, Adobe Photoshop®, Adobe Streamline™, MacLink®Plus, Aldus® FreeHand™, Collage Plus™.

Ziff-Davis Press, an imprint of

Macmillan Computer Publishing USA

5903 Christie Avenue

Emeryville, CA 94608

(510) 601-2000

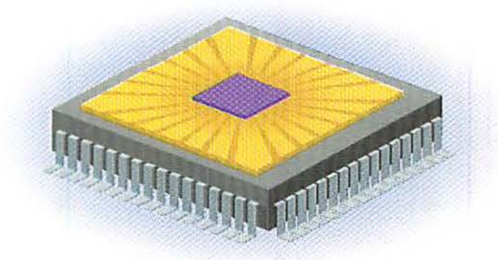
Published 1993. Second edition 1996.

PART OF A CONTINUING SERIES

ISBN 1-56276-401-2

Manufactured in the United States of America

10 9 8 7 6 5 4 3

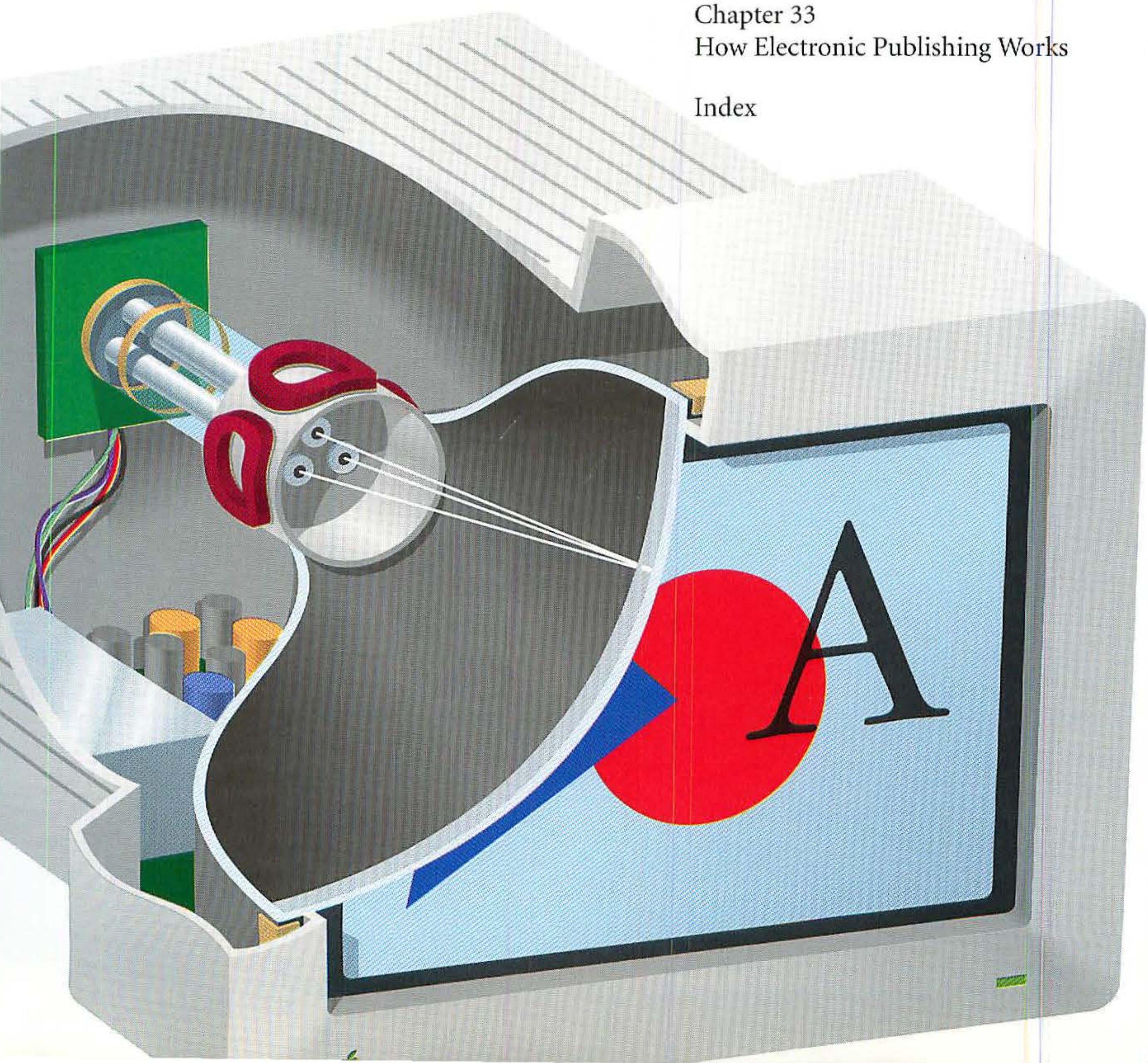


For my wife, Christine, who takes time out from her own work to help and support me with mine.

—John Rizzo

For my family. They graciously give me the tools, talent, and freedom to do my best.

—K. Daniel Clark



A

WHEN I began writing the second edition of *How Macs Work*, it had been three years since I had the pleasure of working with K. Dan Clark, who illustrated both editions. We first worked together on the original edition because we had created "How It Works" sidebars for *MacUser* magazine articles. While most book publishers farm out the illustrations to one or more contract artists, Dan and I worked together on *How Macs Work*, hashing out the proto-sketches that Dan later turned into the graphical interface of this book. Three years later, we picked up where we had left off, delving into new Mac secrets.

Uncovering the mechanisms that Apple had taken great care to hide from the user took some detective work. We had to tear apart a lot of hardware, most of which came from Apple and the folks at *MacUser* magazine. But looking inside the hardware only gave us part of the story. I tracked down the creative people at Apple, who proved helpful in answering obscure questions and pointing us in the right direction. For the second edition, I didn't need to talk to as many people because I found answers on Apple's many World Wide Web sites and its eWorld Technical Library. I also looked for clues in other technical literature, such as Addison Wesley's *Inside Macintosh* series, Scott Knaster's classic Mac programming books, as well as *The Adobe PostScript Language Reference Manual.*

However, I knew from the start that it would take more to explain the inner workings of the Mac than the engineering block diagrams found in the technical literature. It wasn't too difficult to come up with ways to depict hardware, but we were meshing hardware and software concepts together. We had to figure out how to represent software, which the user never sees, in a drawing. Dan usually had an answer when all I could come up with was a rectangle.

When it came time to assemble the gathered evidence, we depended on a crack team of experts. Jeff Green edited most of the first edition and helped us conceptualize much of it. Valerie Haynes Perry helped us wrap up the first edition and was the task master for this second edition. Kim Haglund helped us with the logistics of the art on the first edition; Paul R. Freedman, our technical forensics expert for the second edition, helped us piece together the last remaining bits of the story.

There are some other people not directly involved with this project that we would like to thank. Dan is grateful to Mike Yapp for showing him how to draw from the inside, and to Lisa Orsini for teaching him the language of illustration. I'd like to thank my mother for letting me take over her kitchen with a rented Mac and a modem tied to her telephone line while I finished the second edition. This book shows that you can accomplish a great deal with a Macintosh, and that you should never bring a computer on vacation with you.

—John Rizzo
San Francisco
1996

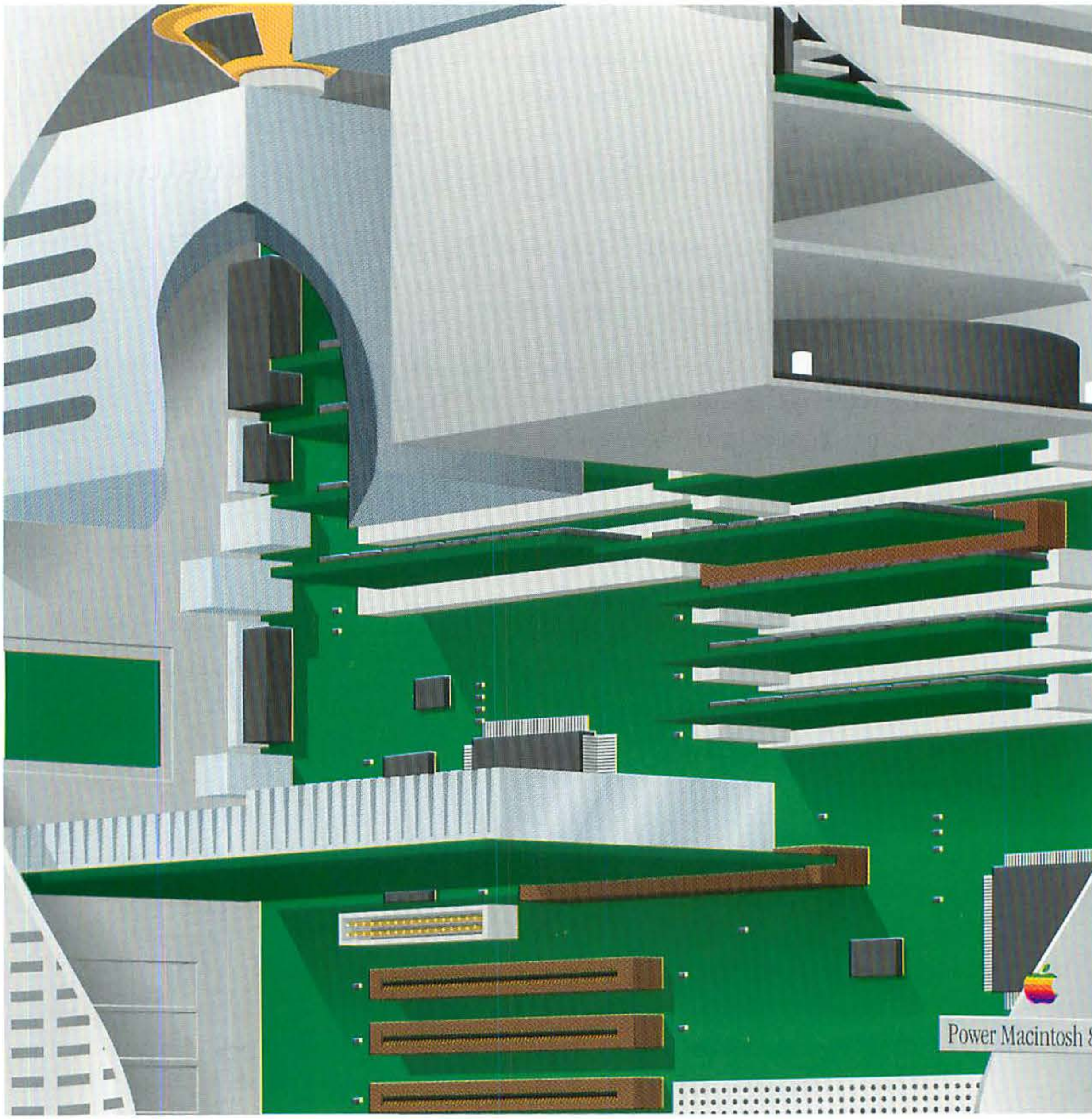
EVER wonder what happens when you pull down a menu? Insert a floppy disk? Move a mouse? Ever wonder what makes a Mac so easy to use? The answer lies behind the windows, inside the microchips, and among the microscopic components that make up the bits and bytes we call data. A Mac is more than windows and a mouse. It's a complex system of interrelated circuits and software, designed symbiotically from the ground up to work together to make the Mac do what you want it to.

The Mac platform, now well into its second decade, has taken many shapes and sizes during its journeys around the globe, under the sea, and in space aboard NASA's shuttle. Both the machine and the software have changed dramatically since 1984 and are still evolving into more powerful and flexible forms. So it is with *How Macs Work.* When we wrote the first edition in 1993, terms that are common today, such as Internet and multimedia, were considered nerdy jargon. This second edition covers important advances in Mac computing, including Mac OS 8, new PowerPC microprocessors, video and virtual reality software, and Mac clones—non-Apple, Macintosh-compatible PCs that were unimaginable a few years ago.

Even with the new processors and parts, today's Macs can still run much of the software designed a decade ago. This has to do with the basic philosophy of the Mac design, which is a constant. It's a philosophy of modular integration, and of enabling a user to do his or her own work instead of the work of learning a computer system. The user interface is not something slapped on top of the system, but is an integral part of the system itself. We follow this philosophy throughout this book.

How Macs Work is not a technical manual. We have tried to make this book Mac-like. Through communicating in a mostly graphical form, we have endeavored to create an easy-to-use book. The text is written in conversational language, and a conscious effort has been made to avoid acronyms and technical jargon whenever possible. However, this book is full of information, covering every aspect of the Macintosh, from the internal electronics to the networks outside. It goes into many details never before presented in layperson's terms. Concepts of software and hardware that are often presented in two different technical languages have been integrated together here. We feel you can't separate the two, because that is how Macs work.

You may want to start with Part 1, a look inside the different types of Macs and the parts that make them up. From here you can jump to an area that interests you, much as you point and click on the Mac screen. (Unfortunately, you'll have to turn the pages yourself.) If you read this book linearly, you'll first find the core Mac processes, and gradually move out toward the peripherals outside of the Mac case. We end, appropriately, with the desktop and electronic publishing processes that made the Mac popular and created this book.


Power Macintosh 8

P A R T

I

INSIDE THE MAC

UPON first glance, the inside of a Macintosh is a mysterious jumble of silicon, metal, and plastic. Microscopic electronic circuits and macroscopic mechanical subsystems joined by winding cabling work together to provide the icons and menus of the Mac desktop. The fact that many users have never seen the Mac's internal components should not be surprising—the Mac is designed to let you do everything you need to with the keyboard, mouse, and exterior ports. Unless you are upgrading or repairing a Mac, there is little reason to open one up. There are no switches to set, and most of the internal hardware can be reconfigured through software with the click of a mouse. Plug in, turn on, and tune in.

On the other hand, there isn't much preventing you from opening up a Mac to gaze at its guts—you can open most Mac models without having to lift a screwdriver. If you were to open up several different Mac models (or read the next few chapters), you'd begin to notice some of the same parts in different places. These components form a basic set that is common to the dozens of models that have evolved from the original 128K Macintosh.

The central component of any personal computer is the *logic board*, which contains the thinking parts of a Mac. The logic board's name comes from the fact that it consists of digital circuitry, which operates on the mathematical principle called Boolean logic. (In the first Macs, another circuit board called the *analog board* contained the analog power circuitry that powered the logic board.) The Mac logic board's digital circuitry controls the central processing unit (CPU), random-access memory (RAM), and operating system code built into a microchip called read-only memory (ROM).

The CPU is the commander of the logic board and of the Mac. Some Macs use the older 68000 family of CPU microprocessor chips, while most new Macs use the much faster PowerPC family of processors. No CPU, however, can contain all the software and data it needs to run a computer, so the Mac's RAM acts as an extension of the CPU, temporarily storing code used by the CPU. The Mac's ROM plays a different role, acting as a permanent storage mechanism for some of the core Mac OS code that makes a Mac a Mac.

Additionally, the Mac logic board contains the controller chips that create and use sound and graphics. Logic board controller circuitry is also responsible for communications with peripheral devices such as modems, printers, and scanners, and for networking with other computers. Still other controller chips run the logic board's expansion slots, which enable you to plug in new circuitry that adds functionality to your Mac. Because the add-in cards are often called *daughter cards*, you'll often hear the logic board referred to as the *motherboard*.

The logic board also connects you to every part of the Mac. You tell the Mac what to do using the keyboard and mouse. The Mac tells you what it is doing with the video display and speakers. Inside the Mac, a *hard-disk drive* stores your work and software for safekeeping when your Mac is turned off, and for quick access when it's turned on. A *CD-ROM drive*, now standard in most desktop Macs, reads multimedia data from permanently encoded optical discs and plays your favorite audio CDs as well.

While these components are common to most Mac models, there are obvious differences in the shape, size, and speed of Macs of the past and present. Classic-style Macs, minitowers, laptops, "pizza boxes," and standard desktop models each have some unique components that differentiate them from one another. For instance, PowerBooks have battery packs that enable you to use them while traveling. Large tower models have extra-large power supplies that can run many internal disk drives and expansion cards. This makes them well suited for use as video or multimedia production workstations or as network or Internet servers.

Macs became even more differentiated in 1995 when Apple began to license its technology to other manufacturers. Though every Macintosh clone uses the same PowerPC microprocessor chips found in Apple's Macs, the non-Apple Macs brought new ways of solving hardware design problems to the market. The first generation Mac clones contain some components no Apple Mac ever had, but still contain some of the same logic board components, including the ROM, found in every Mac.

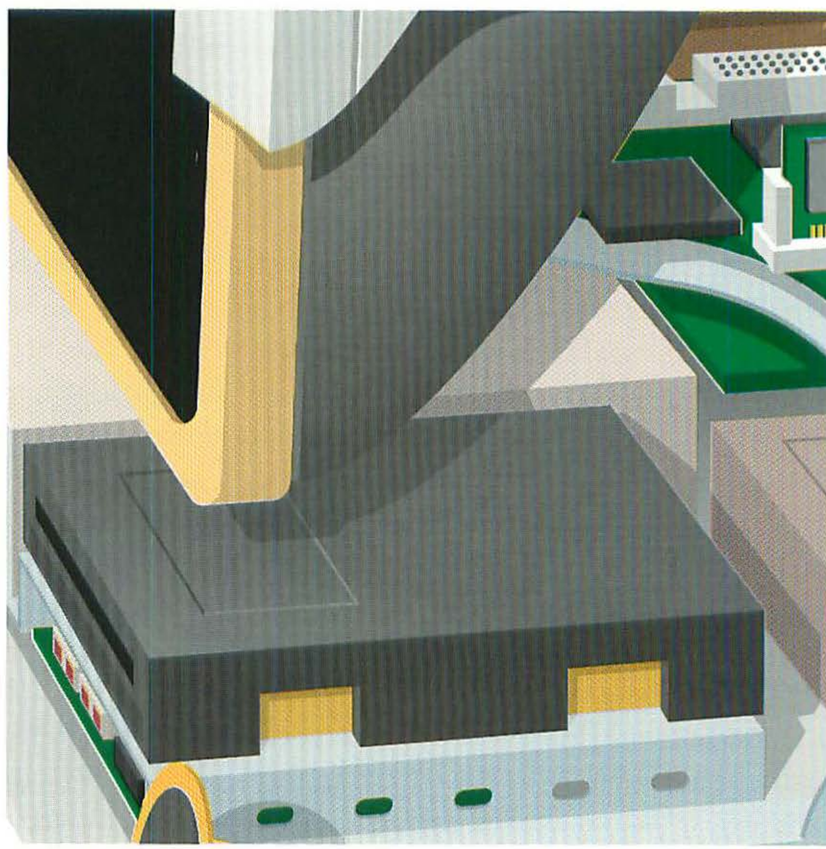
This is not the case with the second generation of Mac compatible technology, the Power PC Platform (formerly called the Common Hardware Reference Platform, or CHRP). Unlike the first generation of Mac clones, no Apple chips, including ROMs, are required in the PowerPC Platform machines. Anyone can build a Power PC Platform machine if they follow the specifications designed by Apple, IBM, and Motorola. One benefit of the Power PC Platform is that it can run multiple operating systems, including Apple's upcoming Mac OS 8 and Windows NT. To do this, Power PC Platform machines will use components found inside Macs and PCs, but will run on a PowerPC CPU.

Of course, the more Mac clones that become available, the more new shapes and configurations we are likely to see. We can group the well-over 100 types of Macs that have been produced so far into three basic groups: All-in-one Macs, desktop models, and laptops. Most of the Macs that exist today are desktop models, but the others are different enough to warrant a look in the next few chapters. We will also take a look inside Mac clones, current and future, in Chapter 4, "How Mac Clones Work."

Regardless of their shape, cost, or manufacturer, Macs in all these lines can run the same software and have a high degree of hardware compatibility with each other as well. We'll start our tour of the inside of a Mac with a uniquely Apple model line most resembling the original 128K Mac—the all-in-one Mac.

CHAPTER 1

Inside the All-in-One Mac



WHEN someone says Macintosh, the first image that comes to mind is that of the old Classic-style Macs. Their unique all-in-one design made them popular on the sets of movies and television shows and with millions of users. Starting with the original 128K Mac, the all-in-one line lasted ten years, through the Mac Plus, SE, SE/30, and several Mac Classic models. The all-in-one design offered easy setup without the tangle of cables that hang out the back of most computers.

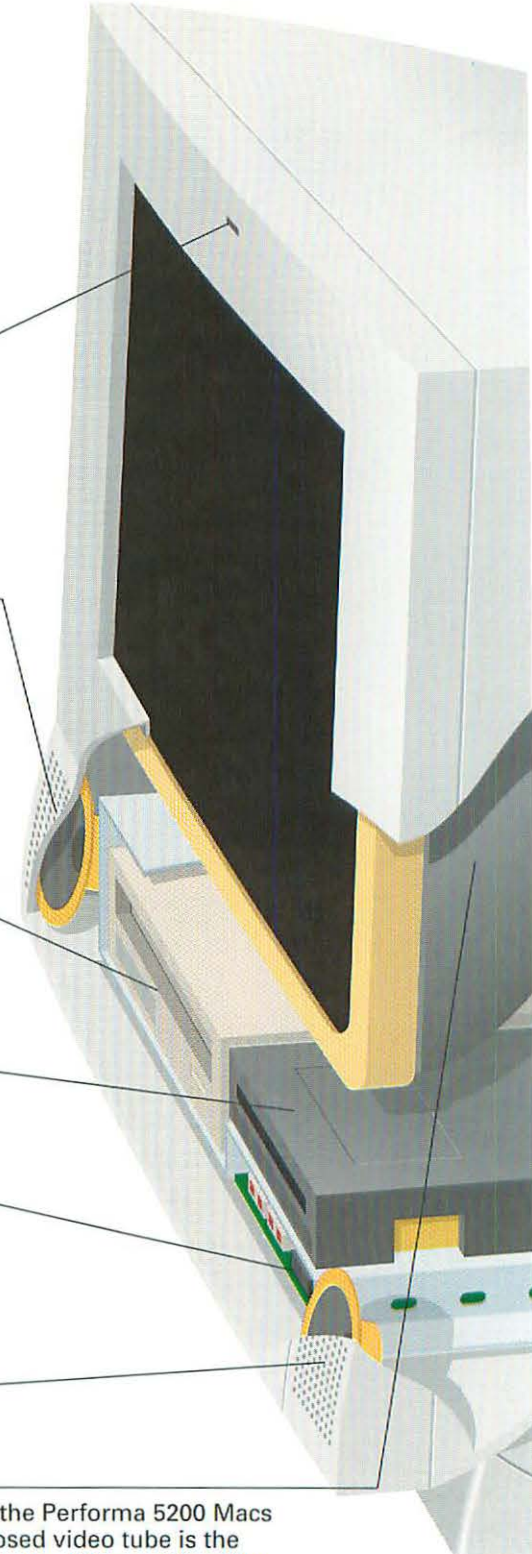
Though limitations such as the small (9-inch) screen finally relegated these original Macs to the history books, the all-in-one design concept is alive and well. Today's incarnation of the all-in-one Mac, the Performa 5200 series, retains the compact form and convenience of the classic Macs. The Performa 5200 sits on a tilt-and-swivel stand, and looks like a display monitor with a floppy-disk slot. However, there's more inside than a 15-inch color picture tube. This all-in-one Mac contains a high-speed PowerPC CPU and internal modem, and is expandable. The Performa 5200 is also a multimedia machine with built-in stereo speakers and microphone, a CD-ROM drive, and expansion slots for video input and a TV tuner.

Despite these high-tech features, the Performa 5200 has a lot in common with the old Classic-style Macs. They both have small footprints, conserving desk space. Instead of arriving on your desk in the some-assembly-required state of most PCs, the all-in-one Mac comes with everything you need already assembled in one box. You can unwrap the Mac from its shipping box, plug it in, and compute.

Part of Apple's original idea for the Mac is a computer that could be set up and used by someone who was not a computer expert. The Performa 5200 comes closer to this idea than did the original Mac, which did not contain everything you needed. It had no internal hard disk, no expansion slots, no video port for a second monitor, and no more than 128 kilobytes of memory. Today's all-in-one Macs have these things and more.

You won't be able to see everything that is inside a Performa 5200. To protect you from the high voltage of a naked cathode ray tube, Apple has made parts of the interior inaccessible. You can, however, access the hard drive and expansion slots from the back of the machine. The logic board also slides out of the computer from the back, enabling you to add your own RAM. Removing and replacing the logic board does not require you to disconnect or reconnect any cables. All connections with the other parts of the Mac are made automatically when you slide the logic board back in. In many ways, the all-in-one Mac continues to be the ultimate plug-and-play computer.

The Performa 5200



Built-in microphone
There is also a sound input port in back.

MAC FACT Early Macs had the signatures of the key people on the Mac design team embossed on the inside of the casing.

Built-in stereo speakers
Since the first Mac, all Macs have had at least one small speaker that could play sound. These speakers are designed to produce quality sound. There are also headphone jacks.

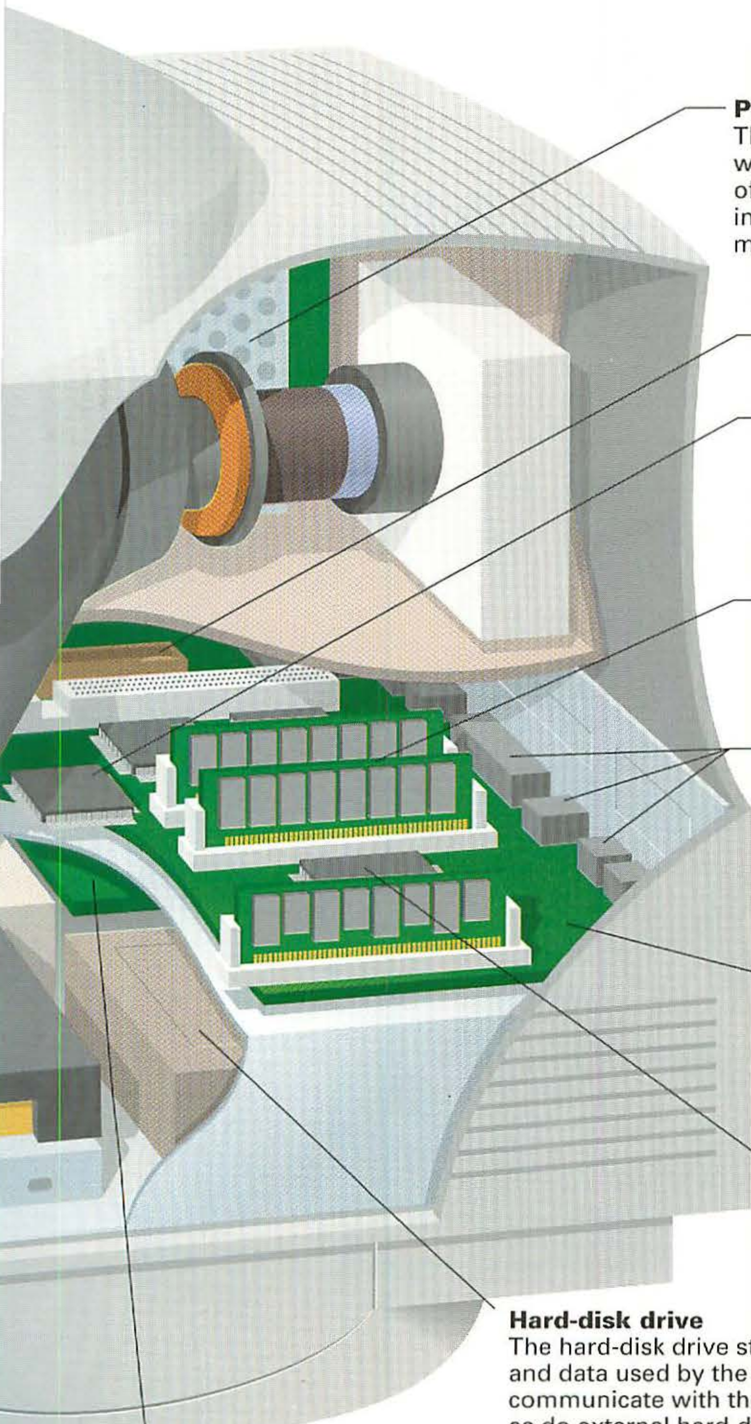
CD-ROM drive
Like the hard-disk drive, the CD-ROM drive communicates with the RAM and CPU through the SCSI bus.

Floppy-disk drive
A ribbon cable connects the floppy-disk drive to the logic board.

Remote control sensor
The optional TV tuner card comes with a hand-held remote control, which you can use to turn on the Mac as well as channel surf.

Built-in stereo speaker

Video display tube
Instead of a separate enclosed video monitor, the Performa 5200 Macs utilize an internal 15-inch video tube. The exposed video tube is the main reason Apple doesn't want users opening an all-in-one Mac: The tube is dangerous to you (it can carry exposed high voltages even when unplugged), and you are dangerous to it (it is fragile).



Power supply
The power supply converts AC power from a wall receptacle and delivers AC and DC power of varying voltages to all the electronic parts inside the Mac, as well as to the keyboard and mouse on the outside.

PDS slot

ROM
The read-only memory contains permanent code used by software applications.

RAM
You can install up to 64MB of memory in the form of standard SIMMs (single inline memory module).

Input/output ports
A standard set of ports connects the logic board to the world outside, including the keyboard and mouse (ADB port), external hard disks (SCSI port), networks and printers (LocalTalk port), and modems (modem port).

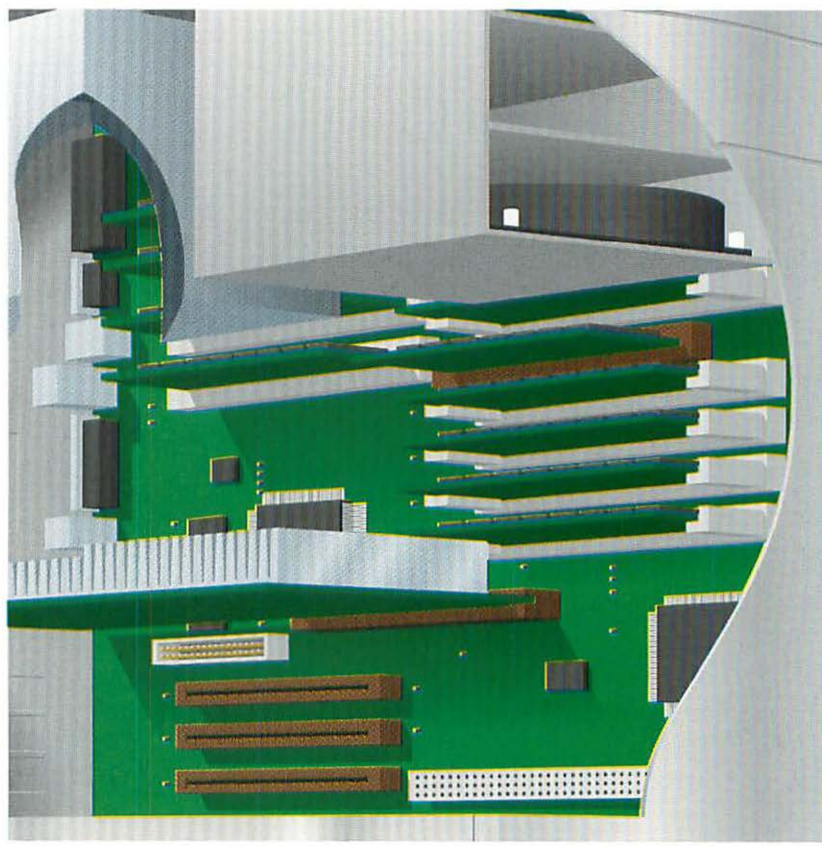
Logic board
The logic board contains most of the Performa's thinking circuitry, including the CPU, ROM, and RAM, as well as the circuitry for the communications and networking ports.

CPU
The brains of the Mac, the CPU does all the calculations that make your software work. In this case, the CPU is a PowerPC 603 microprocessor.

Hard-disk drive
The hard-disk drive stores software applications and data used by the CPU. Internal hard-disk drives communicate with the logic board via the SCSI bus, as do external hard-disk drives.

TV-tuner card
This optional card can connect to cable TV or an antenna to view television in a window on your screen. An optional video card can connect the Mac to a VCR.

CHAPTER 2 Inside a Desktop Mac



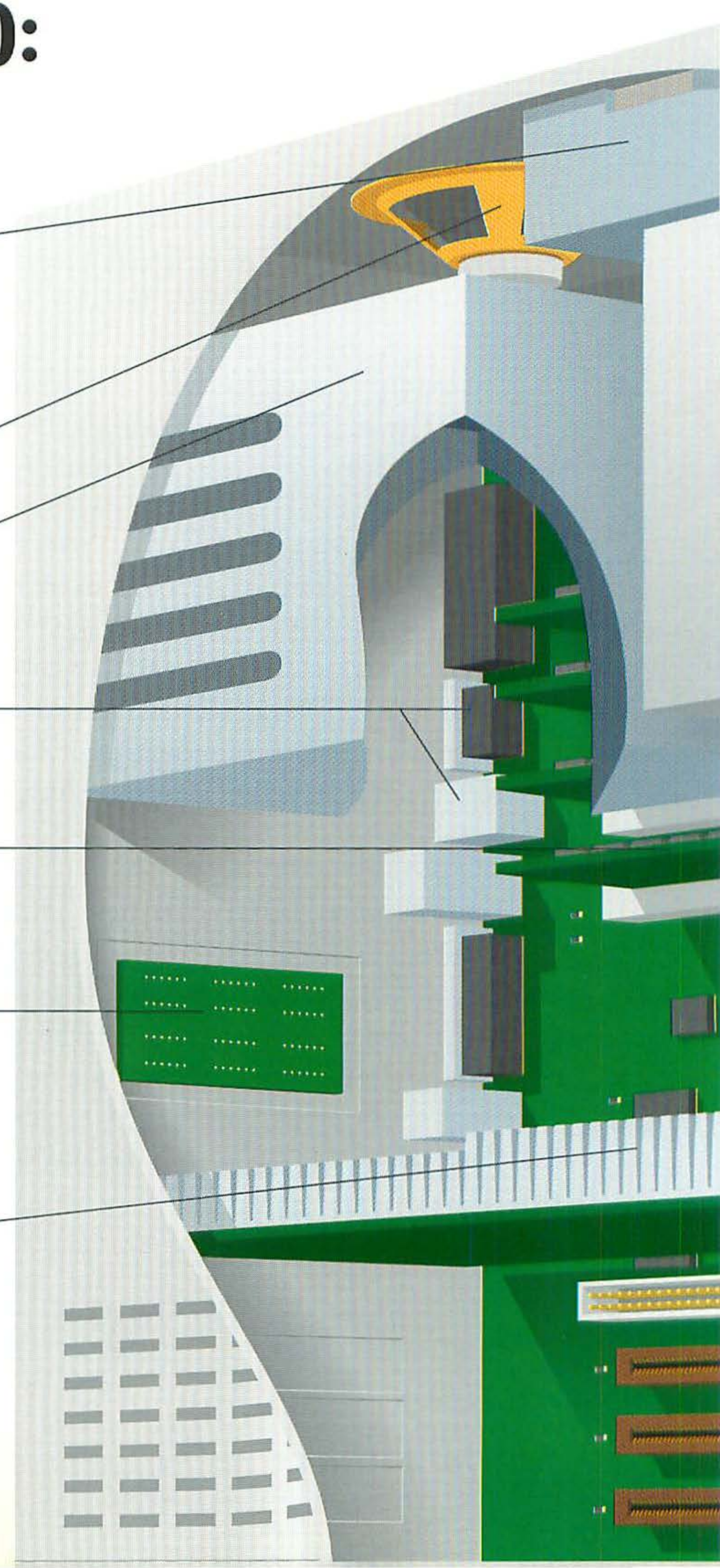
ALTHOUGH the Mac started out as an all-in-one computer, the majority of Macs have been of the standard desktop variety, with separate monitors. This is because standard desktop computers tend to be more customizable than other types. Although today's all-in-one and laptop Macs let you add your own RAM and expansion cards, the standard desktop models tend to have more expansion options. Standard desktop Macs usually have more expansion slots and can accept more RAM and more types of hard drives than other Macs. And, of course, with a standard desktop computer, you can choose your own display monitor.

The first standard desktop Mac, the Macintosh II, appeared in 1987. It was the first model designed to be opened easily by users, and the first Mac with expansion slots. Since the Mac II, desktop Macs have taken many other forms besides the traditional form (about six inches high by a foot or so wide). The popular IIcx and IIci of the early nineties used compact cases no wider than a 13-inch display monitor. Apple used a flat "pizza box" case in some the lower-end Mac models, such as the old LC and the more recent Power Mac 6100.

We've also seen desktop Macs that don't fit under the monitor at all. The so-called *tower cases* stand in an upright position and fit nicely under a desk. These include the old Quadra 950 and the more recent Power Mac 9500, as well as several Apple Workgroup Servers. Tower Macs usually have everything the under-the-monitor varieties do, plus some top-of-the-line features. Towers have more room inside for extra storage devices and expansion slots and usually have the fastest processors.

In this chapter, we'll take you inside a configuration known as a mini-tower, a smaller version of the tower that usually sits on top of the desk. We choose to illustrate the Power Mac 8500 because it contains many of the features found in a variety of different desktop Macs. Some of the 8500's standard features are available as options for other Macs. Though it is one of the higher-end Macs, the Power Mac 8500 is not the fastest or the biggest Mac. It is, however, one of the most complete computers ever built for professionals. Besides a fast CPU, memory, and hard disk, its built-in sound and video in and out ports make the 8500 very popular in desktop publishing and graphic arts as well as in multimedia and video production.

Power Mac 8500: A Mini-tower



CD-ROM drive
The CD-ROM drive and the hard drive connect to an internal SCSI bus that is separate from the external SCSI connector. The internal SCSI bus is a Fast SCSI 2 bus, which runs at 10MB per second, twice as fast as the external SCSI connector.

Speaker

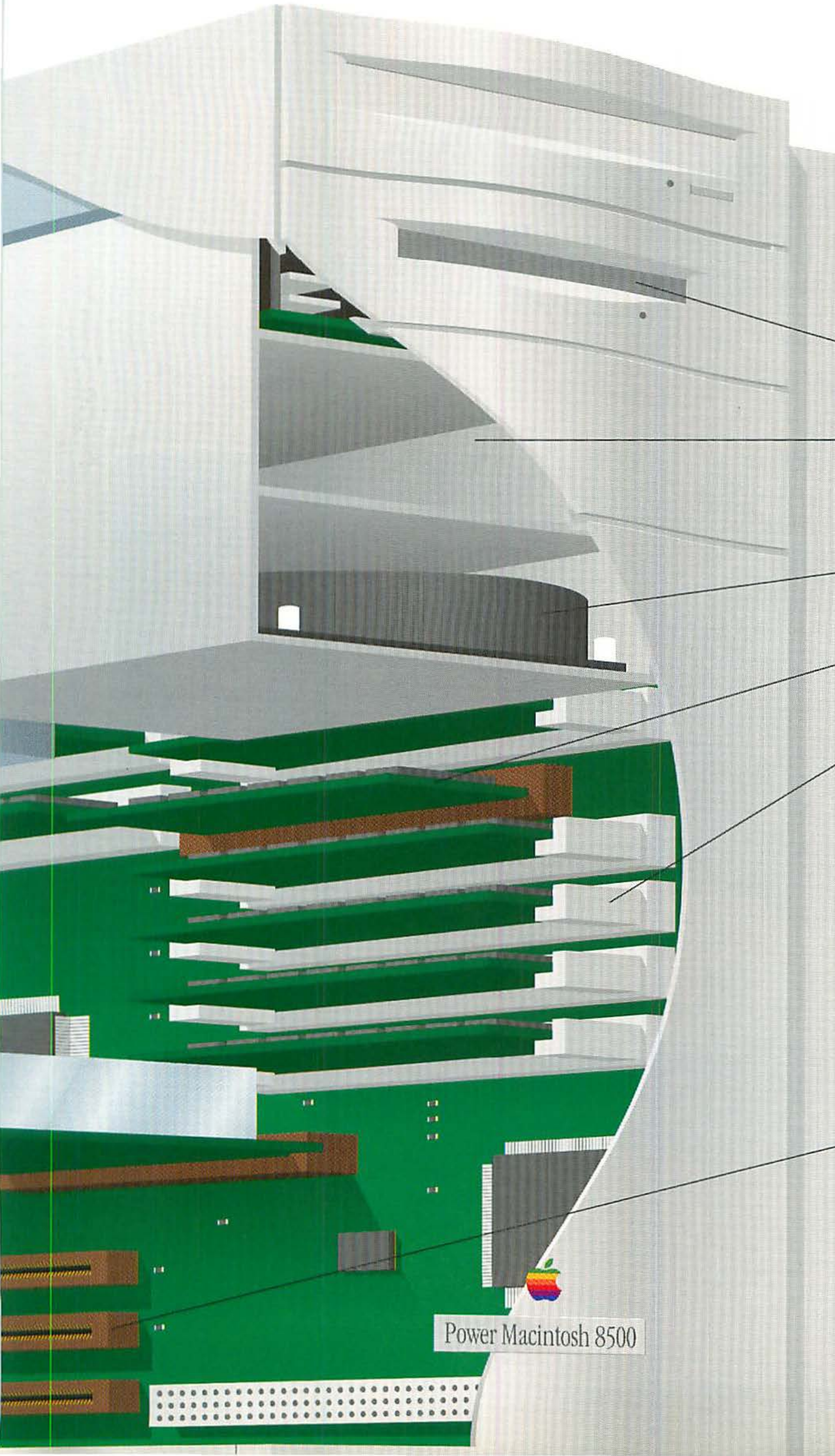
Power supply
The 225-watt power supply provides enough power to run storage devices in the storage bays, as well as expansion cards in the PCI.

Built-in Ethernet
The logic board contains both the LocalTalk and the faster Ethernet network interfaces.

Video display RAM
Up to 4MB of video RAM (VRAM) and a wide 64-bit VRAM bus enable the 8500 to rapidly display millions of colors on a 21-inch monitor.

Video in and out ports
Ports for composite and S-VHS video enable you to capture full motion video at 30 frames per second at a size of 320 × 240 pixels, or at 640 × 480 pixels with fewer frames. The video-out ports enable you to save your edited video to a VCR.

CPU
The CPU is a PowerPC 604 microprocessor, a faster processor than the first-generation 601. As in some of the other Power Macs, the processor sits on a removable daughter card, which you can swap for a faster 604 chip running at up to 150 MHz. (The fins on top of the CPU aid in cooling.)



MAC FACT Overall, a Power Mac 8500 with a 120 MHz CPU is about twice as fast as the first Power Mac, the 6100/60, and almost five times faster than the Quadra 630, which had a 68040 CPU. For some tasks, such as full motion video, the 8500 is much faster.

Floppy drive

Expansion bay
This space can hold an additional storage drive.

Hard drive

Cache

RAM
Like some of the other higher-end Power Macs, the 8500 makes use of DIMMs (dual in-line memory modules), which are faster than the SIMMs of lower-end Mac models. The extra speed is supported by a memory bus that moves 128 bits at a time. Using DIMMs that contain 64 megabytes each, the Power Mac 8500 can hold up to 512 megabytes of memory, over four thousand times the amount of RAM in the original Mac.

PCI
PCI expansion slots are faster than the NuBus slots used in older Macs. The PCI slots can run some of the same expansion cards that run in other PCI-based computers, such as PCs that run Windows. With display, video, and Ethernet already on the logic board, you have a full three slots free for other purposes.

CHAPTER 3 Inside the PowerBook

THE first PowerBooks were a radical rethinking of the laptop computer. These laptops were the first with a keyboard that sits back toward the display screen, providing room to maneuver your arms in tight places as well as a comfortable place to rest your wrists while typing. And at a time when trackballs were a clip-on afterthought on PC laptop computers, the PowerBook integrated a trackball in an ergonomic central position. Also radical for a laptop computer was the ability to instantly connect external devices such as mice, keyboards, hard disks, and in some models, monitors.

In today's PowerBooks, the trackball is gone, replaced by a cheaper and somewhat less ergonomic cursor-control device called a trackpad. Modern PowerBooks, however, include advances such as PowerPC microprocessors for Power Mac performance, bigger and brighter display screens, and infrared beaming windows for instant network communication. Another big improvement is a modular design that lets you install your own memory, or add a second hard-disk drive, a modem, wireless communications modules, and other devices.

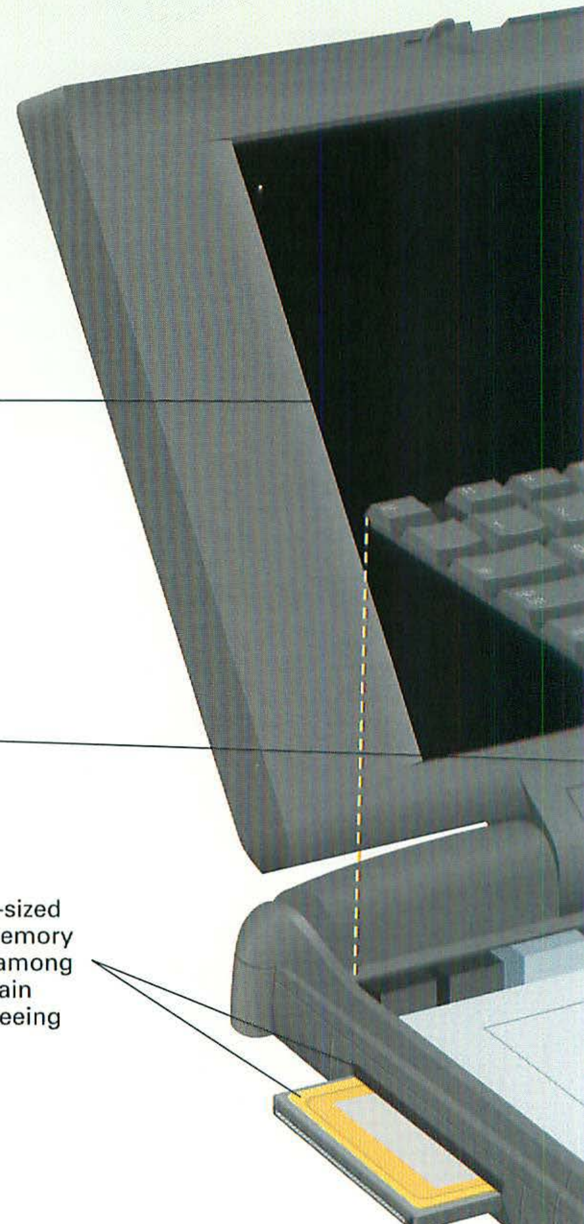
There are still two types of PowerBooks: the standard line and the lighter-weight PowerBook Duos. The standard PowerBook has the same external ports as a desktop Mac, including a SCSI port, a printer port, an Apple Desktop Bus (ADB) port for an external keyboard and mouse, and in some cases, a port for standard color video monitors. PowerBooks also have some ports desktop Macs don't have, including external slots for credit-card-sized expansion cards, call *PC cards.*

The Duo is both a laptop and desktop Mac, splitting the Mac functions into two parts. The CPU, RAM, hard disk, and a serial port are in the laptop, while a desktop docking station called the *Duo Dock* contains room for another hard drive, a floppy drive, most of the input/output circuitry, and ports, including SCSI and video output. This lightens the weight of the laptop component of the Duo system to under 5 pounds (compared to over 6 pounds for the standard PowerBook), with a thickness of less than 1½ inches.

When docked, a Duo has everything that a desktop Mac has, and more than a standard PowerBook, including two NuBus expansion slots, which are built into the Duo Dock. There are also lighter, more compact docks available from Apple and third-party manufacturers; they take the form of light-weight bars that you plug into the back of the Duo. These docking bars can provide features such as Ethernet ports or support for large monitors, and let you take them on the road with you.

All PowerBooks have low-power, flat-panel displays. PowerBook displays come in monochrome, grayscale, and color versions, but color displays draw more power and shorten battery life. All PowerBooks come with power-saving software, which turns off the display and hard-disk drive after a set period of inactivity.

PowerBook 5300



Display All PowerBooks (including the Duos) come with one of two types of liquid crystal display: passive matrix or active matrix. The latter provides sharper images, but is more expensive. Both are lit from behind with low-power lamps. A third option, dual scan, is actually an improved passive matrix design. The display screen in the 5300 measures 9½ inches, bigger than the screen in the first desktop Macs.

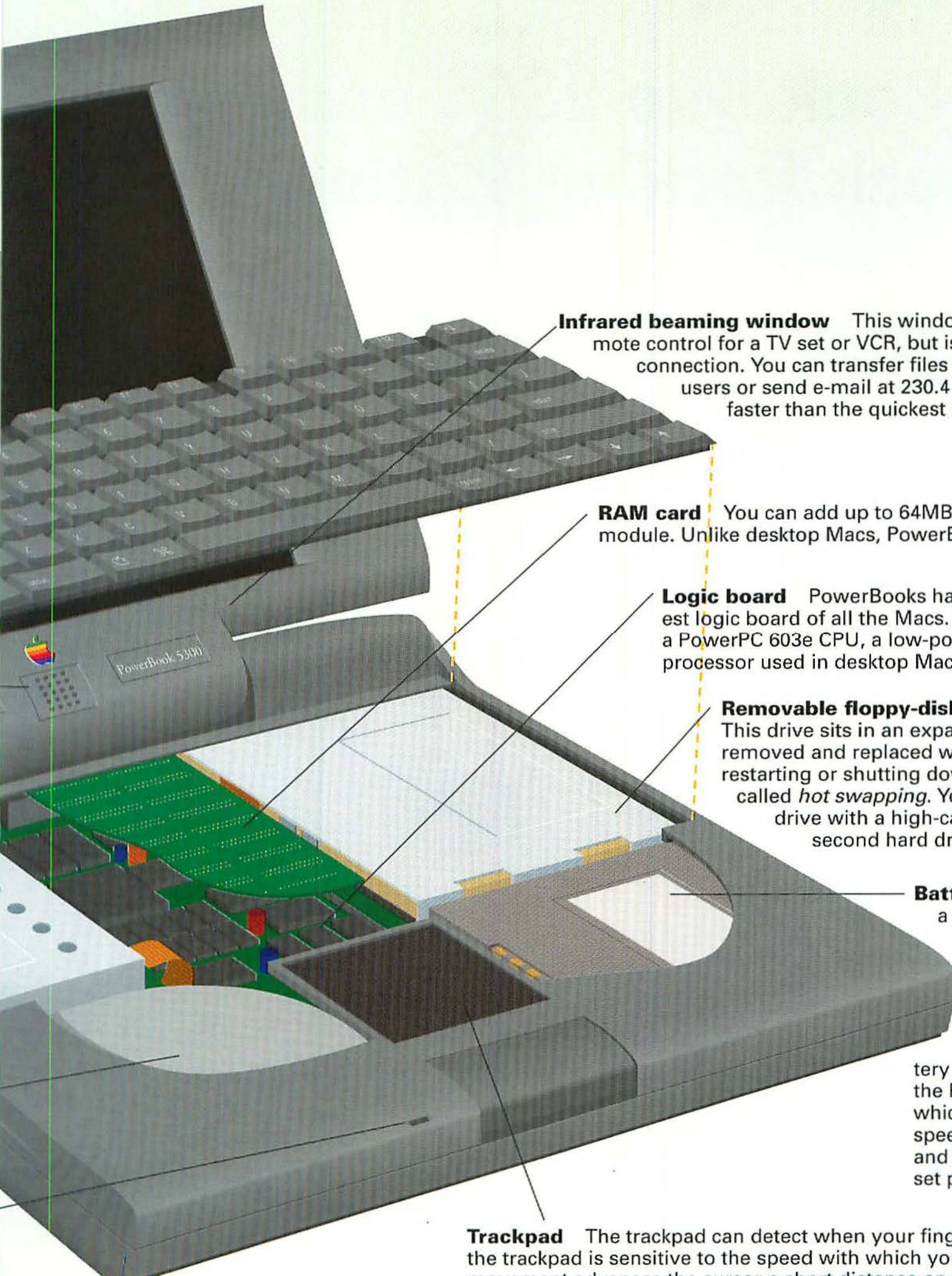
Built-in microphone

PC card slots These two slots can accept credit-card-sized PC cards (also called PCMCIA cards). PC cards can be memory devices, modems, or wireless communication devices, among other things. There are several cards available that contain both a modem and an Ethernet interface on one card, freeing the second PC card slot for another expansion card.

MAC FACT Apple's first attempt at a battery-powered Mac, the Macintosh Portable, had a battery life of almost 12 hours, but weighed in at 17 pounds. It wasn't what users wanted, and Apple sold less than 100,000 units. The 7-pound PowerBook, with a 3-hour battery life, was an instant success and became the best-selling laptop computer ever: Apple sold $1 billion worth during the product's first year alone.

Hard-disk drive Hard drives in PowerBooks usually weigh 6 ounces or less and contain disks that range from .3 inches to 2.6 inches in diameter.

Automatic sleep switch When you close the lid while the power is still on, the hook in the lid touches a metal contact in the receptacle, completing a circuit and activating the sleep mode.



Infrared beaming window This window is similar to the remote control for a TV set or VCR, but is a wireless network connection. You can transfer files to other PowerBook users or send e-mail at 230.4 kbps, eight times faster than the quickest modems.

RAM card You can add up to 64MB of RAM via a special RAM module. Unlike desktop Macs, PowerBooks do not use RAM SIMMs.

Logic board PowerBooks have the smallest and lightest logic board of all the Macs. This logic board contains a PowerPC 603e CPU, a low-power version of the microprocessor used in desktop Macs.

Removable floppy-disk drive/expansion bay This drive sits in an expansion bay, and can be removed and replaced with another drive without restarting or shutting down the PowerBook. This is called *hot swapping*. You can replace the floppy drive with a high-capacity optical drive, a second hard drive, or other devices.

Battery Most PowerBooks use a rechargeable nickel-metal-hydride battery, which can be recharged in 2 hours. The battery slides out the right side for easy replacement. System software can greatly increase battery life by occasionally putting the PowerBook in sleep mode, which slows the processor clock speed and turns off the hard disk and screen back-lighting after set periods of inactivity.

Trackpad The trackpad can detect when your finger is touching it. Like a mouse, the trackpad is sensitive to the speed with which you move your finger. A slow movement advances the cursor a short distance on the screen, and a rapid movement advances the cursor a greater distance. You can also click by tapping on the trackpad with some PowerBook models. Unfortunately, this is not true in the 5300, which requires that you click a separate button.

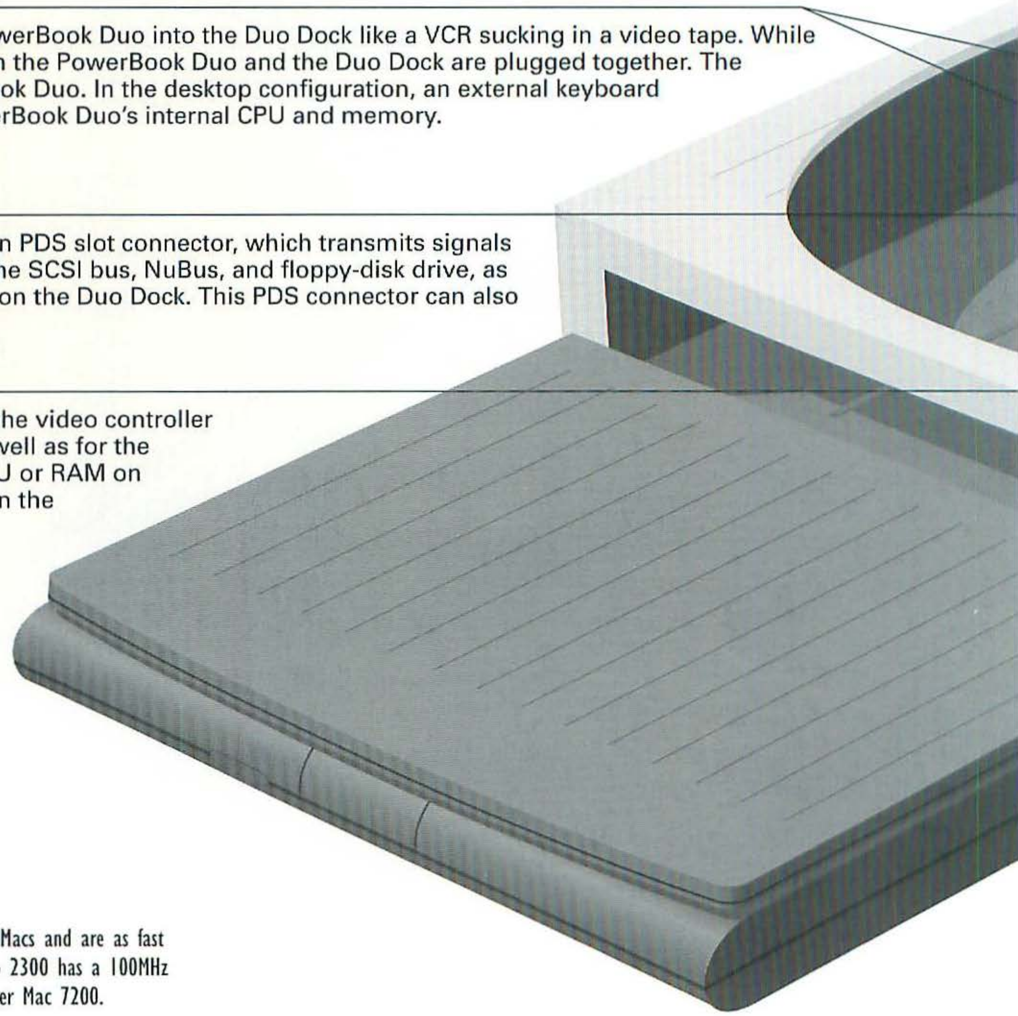
PowerBook Duo 2300 and Duo Dock Docking Station

NuBus slots Two NuBus slots in the Duo Dock can accept the standard expansion cards used in desktop Macs.

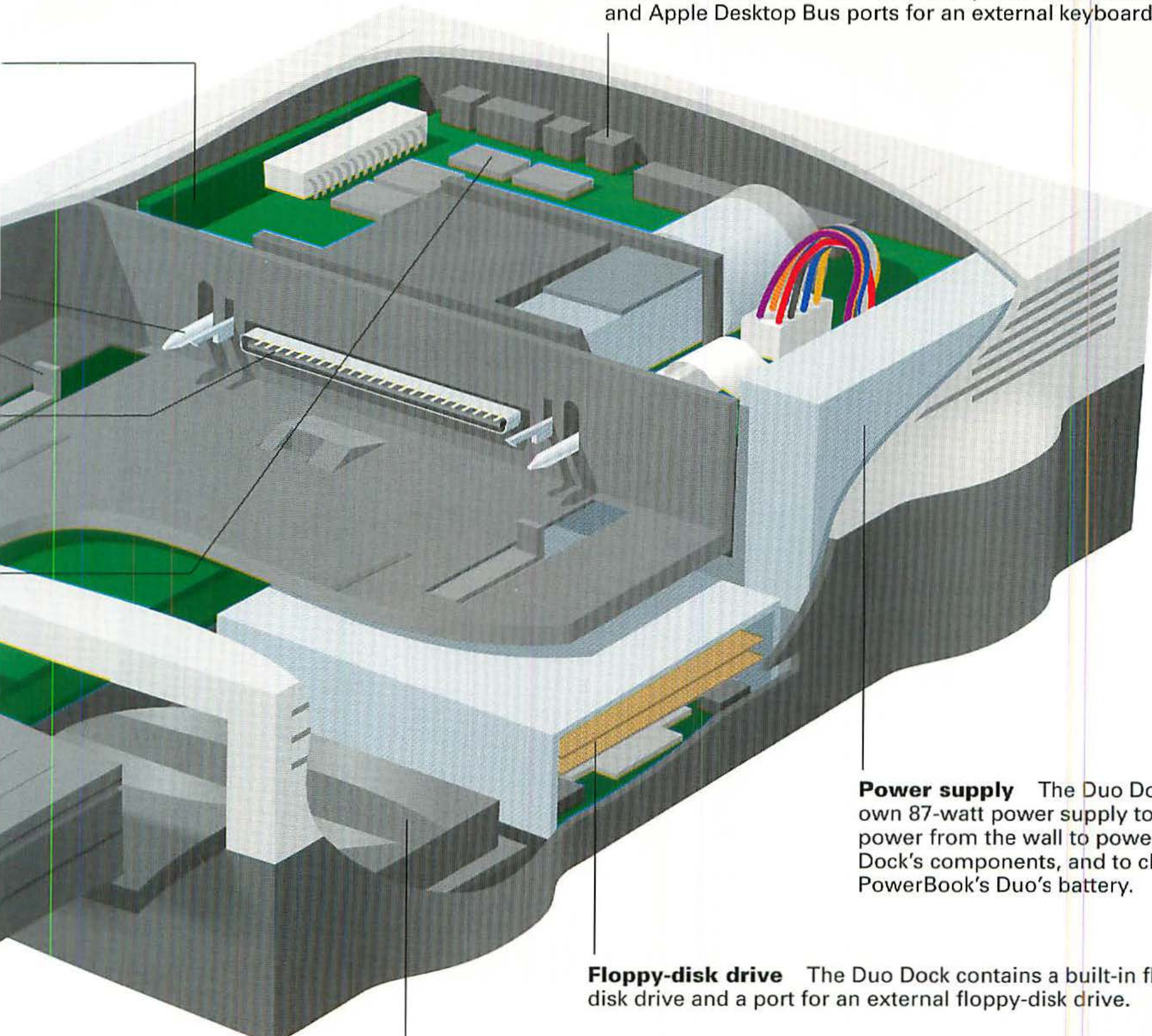
1. The power latch pulls the closed PowerBook Duo into the Duo Dock like a VCR sucking in a video tape. While this happens, the PDS connectors on the PowerBook Duo and the Duo Dock are plugged together. The power latch also ejects the PowerBook Duo. In the desktop configuration, an external keyboard and monitor are used with the PowerBook Duo's internal CPU and memory.

2. The expansion connector is a 152-pin PDS slot connector, which transmits signals from the PowerBook Duo's CPU to the SCSI bus, NuBus, and floppy-disk drive, as well as to the video and other ports on the Duo Dock. This PDS connector can also plug into smaller docking bars.

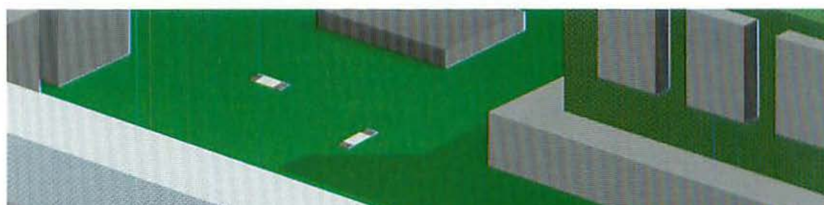
3. The Duo Dock logic board contains the video controller circuitry for an external monitor as well as for the input/output control. There is no CPU or RAM on this logic board; the CPU and RAM in the PowerBook Duo are used.



MAC FACT PowerBook duos can work nicely as desktop Macs and are as fast as midline desktop Macs. For instance, the PowerBook Duo 2300 has a 100MHz PowerPC 603 CPU, which makes it about as fast as a Power Mac 7200.


Ports The Duo Dock supplies the standard Mac ports: SCSI, standard video, Ethernet, two serial ports, sound in and out ports, and Apple Desktop Bus ports for an external keyboard and mouse.
Power supply The Duo Dock has its own 87-watt power supply to convert AC power from the wall to power for the Dock's components, and to charge the PowerBook's Duo's battery.
Floppy-disk drive The Duo Dock contains a built-in floppy-disk drive and a port for an external floppy-disk drive.
Optional dock hard-disk drive In addition to the hard-disk drive in the PowerBook Duo, the Duo Dock itself has a storage bay for an optional second hard-disk drive.

CHAPTER 4 How Mac Clones Work



THE word "Macintosh" described a unique union of system software and hardware for over a decade. Then in 1995, Apple created the Mac OS brand name and a separate identity for the software that drove the hardware. More than just a logo to stick on software packages, this new identity allows the Finder and all Mac applications to run on personal computers that don't have the rainbow apple on the front. The Macintosh-compatible machines—or Mac clones—have the same hardware and software plug-and-play compatibility that has always made the Mac the leader in ease-of-use.

There are two types of Mac clone design. The first is based on traditional Macintosh hardware, while the second, the PowerPC Platform (*PPCP*), is a radical rethinking of the personal computer, developed jointly by Apple, IBM, and Motorola. Both designs use the PowerPC microprocessor as their CPU.

The first generation of Mac clones is based on Apple's specifications for the logic boards of various Mac models. These clones, from companies such as Power Computing and DayStar Digital, use some key Apple components to ensure compatibility with the existing catalog of Mac software and add-on hardware. These key components include the Apple ROM and application-specific integrated circuits, or *ASICs* (pronounced *AY-six*). The Apple ASICs contain a lot of the controller circuitry for the various input and output systems. Apple also helps clone manufacturers to design their machines and certifies each new clone model for Macintosh compatibility. So far, testing proves that the clones are as Mac-like as Apple's Macs.

The clone manufacturers can also use a lot of standard, off-the-shelf parts, including internal metal frames and the CD-ROM and floppy drives that sit in them. Because these parts are the same parts that are used in millions of Intel-based PCs, Mac clones don't look as distinctive as Apple's Macs, but often can cost less than Apple's Macs, for the same features.

The PowerPC Platform (once referred to as the Common Hardware Reference Platform, or CHRP) goes even further in the direction of using standard PC parts. PPCP doesn't require the use of Apple ASICs at all. The ROMs in PPCP machines are much smaller than those in traditional Macs, and can be manufactured by companies other than Apple. This drives down costs and induces major PC manufacturers to build their own PPCP models.

The most radical aspect of PPCP is that it is designed to run multiple operating systems. In addition to Mac OS, the PPCP machines run Microsoft's Windows NT, IBM's OS/2, IBM's version of UNIX, AIX, Sun's UNIX front end, Solaris, and Novell's NetWare network operating system for servers. Supporting these models are standard Mac and PC ports.

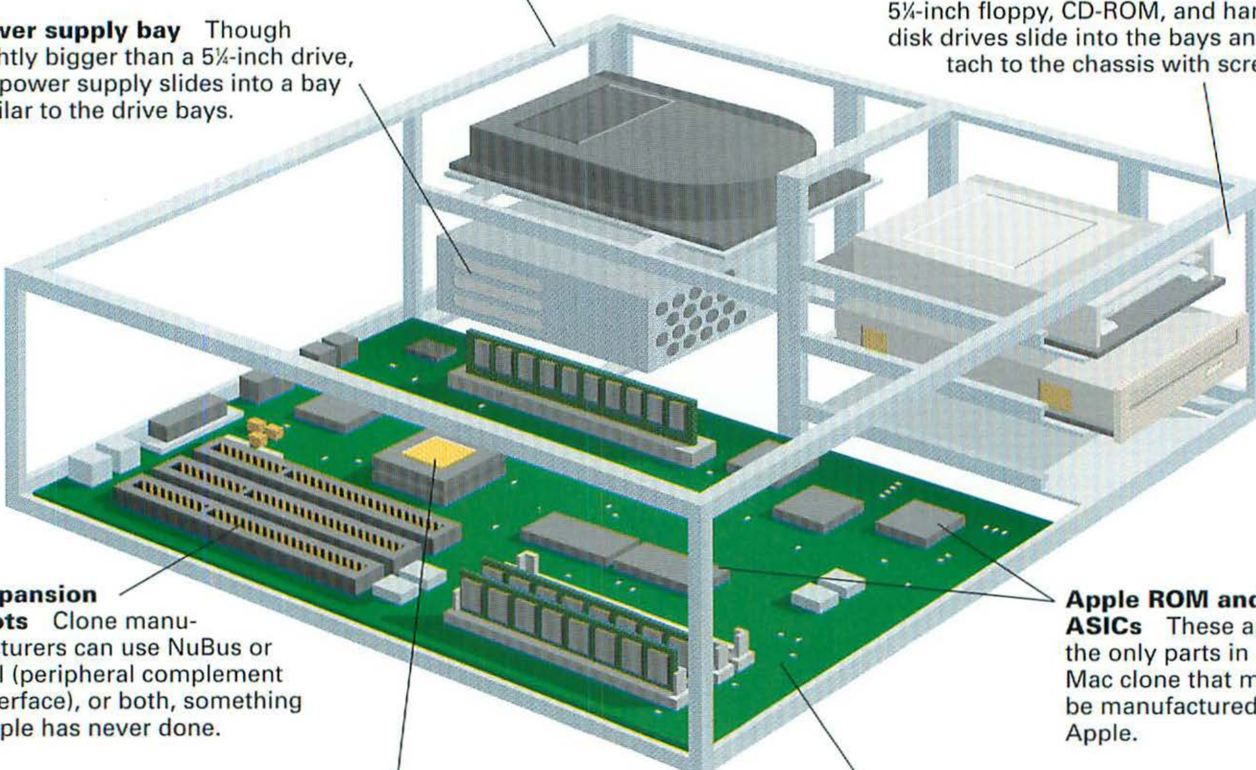
The first PPCP machines, to be released at the end of 1996, are expected to expand the Mac clone market and the use of Mac OS. One reason is that for the first time, IBM will be building a machine that runs Mac OS. The old saying that nobody ever got fired for buying IBM still has a lot of truth to it.

Mac Clone

Standard PC chassis Clone manufacturers can use standard PC metal frames that hold the logic board, drives, and power supply. A metal case usually slips over the top.

Drive bays Standard 3½-inch and 5¼-inch floppy, CD-ROM, and hard-disk drives slide into the bays and attach to the chassis with screws.

Power supply bay Though slightly bigger than a 5¼-inch drive, the power supply slides into a bay similar to the drive bays.



Expansion slots Clone manufacturers can use NuBus or PCI (peripheral complement interface), or both, something Apple has never done.

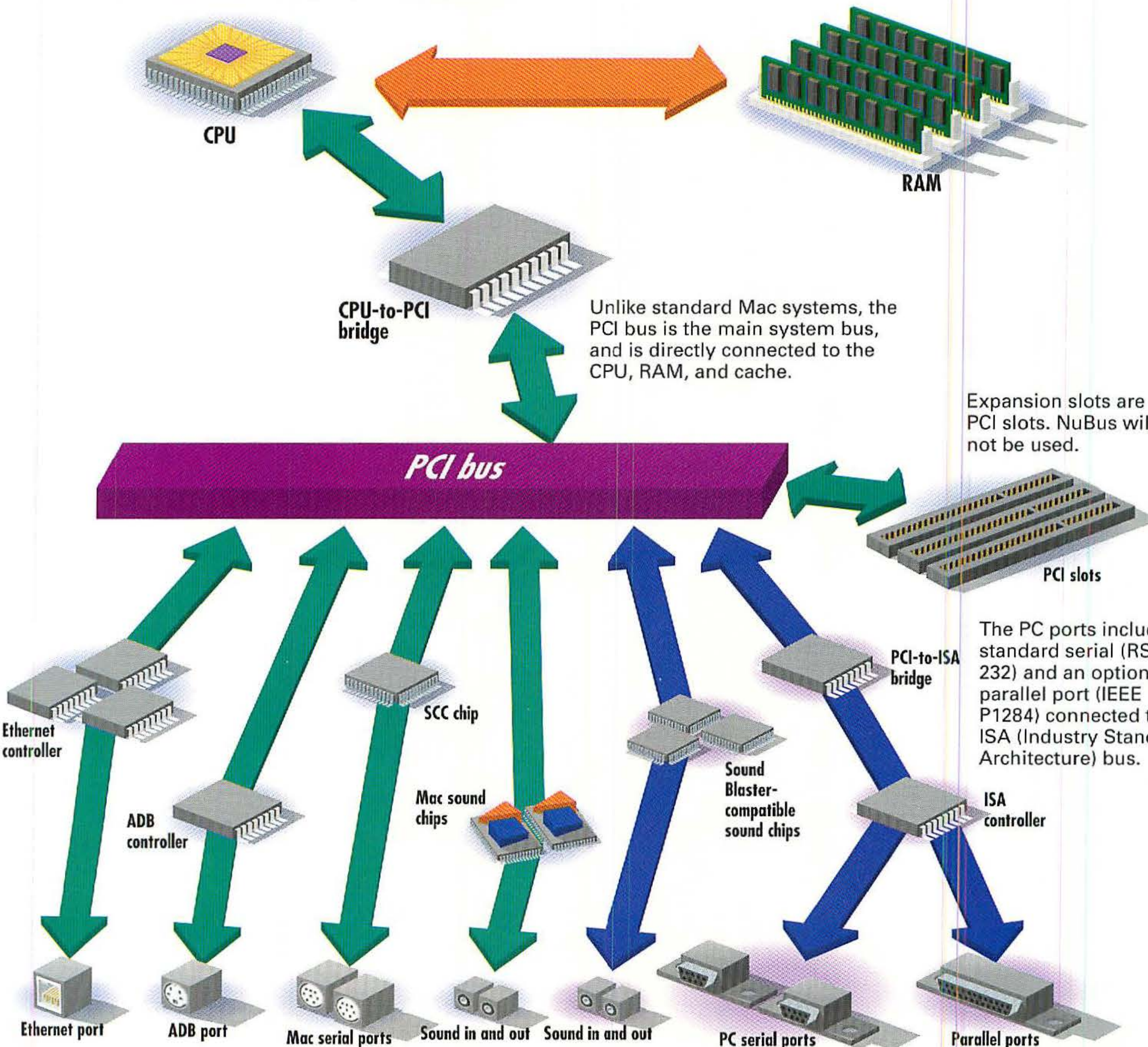
Apple ROM and ASICs These are the only parts in a Mac clone that must be manufactured by Apple.

PowerPC CPU This can be mounted on the logic board or on a removable daughter card. Manufacturers have the option of putting other things on this daughter card, such as an expansion slot.

Logic board The manufacturers can use Apple's logic board or build their own based on Apple specifications.

PowerPC Platform (PPCP)

The PPCP unit can start up with with either Mac OS or another operating system, or in a mixed mode that runs slower than a single operating system mode.



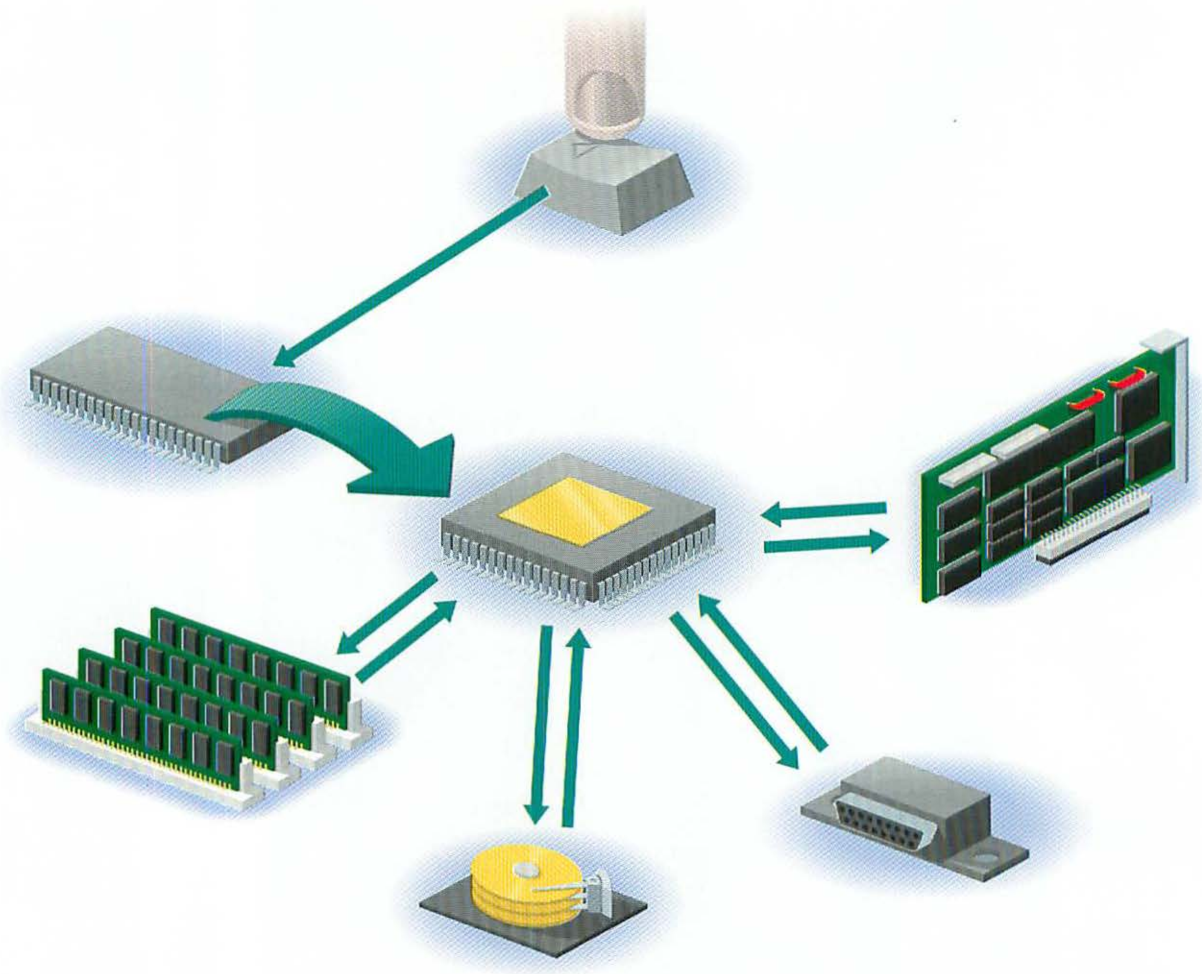
Unlike standard Mac systems, the PCI bus is the main system bus, and is directly connected to the CPU, RAM, and cache.

Expansion slots are PCI slots. NuBus will not be used.

The PC ports include standard serial (RS-232) and an optional parallel port (IEEE P1284) connected to an ISA (Industry Standard Architecture) bus.

The Mac ports include an Apple desktop bus (ADB) for keyboard and mouse, and printer and modem ports enabled for LocalTalk and GeoPort.

PPCP specifies sound-in and sound-out ports for both Mac and Sound Blaster-compatible stereo sound. This enables both Mac and Windows multimedia software to work.



PART 2

MAC OS

IN 1994, ten years after the first Macintosh, Apple created the product name Mac OS. Previously, the software that made the Mac a Mac had no proper name, and had gone by various incomplete names: the Mac operating system, the Mac system software, System 7, the System folder, and the Apple ROM (read-only memory). Mac OS encompasses all of these things.

Mac OS comes preinstalled on your Apple Macintosh or the Macintosh-compatible computers from other hardware vendors. It's what makes the Mac run. It starts up the Mac, displays graphic images on the screen, and produces sound. It opens and saves files to your hard disk, sends and receives data over a modem, and communicates with other computers on networks. Mac OS also creates the graphical interface that your word processors, spreadsheets, and other programs use.

Mac OS is not a single piece of software, but a collection of software routines stored permanently in ROM and in the System folder on the hard disk. The Apple ROM contains several different collections of routines. The ROM code at the lowest functional level is the Mac operating system, a technical term that hearkens back to the early days of programming. The ROM's operating system is not a complete operating system environment as we think of Mac OS or Microsoft Windows, but instead is the part of Mac OS responsible for moving bits between the various pieces of hardware in the Mac.

The ROM largely consists of a collection of procedures and functions called the Macintosh Toolbox, which every application uses. Applications use Toolbox code to create windows, menus, and dialog boxes, as well as some of their respective features. The Toolbox is one of the reasons all Mac programs look and work the same way. One of the jobs of the Toolbox is to manage Mac OS resources in the System folder. Resource files contain the standard elements used by all applications, such as fonts, icons, and sounds. The System file is one such resource file. Application files also contain their own resources, such as menu titles and menu items.

A companion to the System file is the Finder, the application that creates the familiar Mac desktop, the most visible part of Mac OS. Through the Finder, you launch applications and control what files go onto your hard disk. Mac OS also has Control Panels, where you record settings and turn features on and off, and extensions, which add new features to Mac OS.

One of the more innovative new features enabled by extensions is Open Doc, which focuses on user tasks rather than on applications. Open Doc is a method of using *component software*—miniapplications that you can use together to create your own mix of software features. For instance, Open Doc can enable you to use a single document to edit text, perform spreadsheet functions, access the Internet, and create graphics, all without having to have four applications open. Open Doc is an industry standard that runs on Windows and UNIX as well as Mac OS.

Although Open Doc is an add-on to System 7.5, the next version of Mac OS, Mac OS 8, will incorporate Open Doc and other recent add-on features as core Mac OS features. However, Mac OS 8 goes well beyond the capabilities of System 7.5 and represents the first time that Apple has rewritten Mac OS from the ground up. Mac OS 8 will offer improved performance through a more efficient structure, based on a new operating system core called the Microkernel. Other improvements include a new memory model and code that is completely PowerPC native.

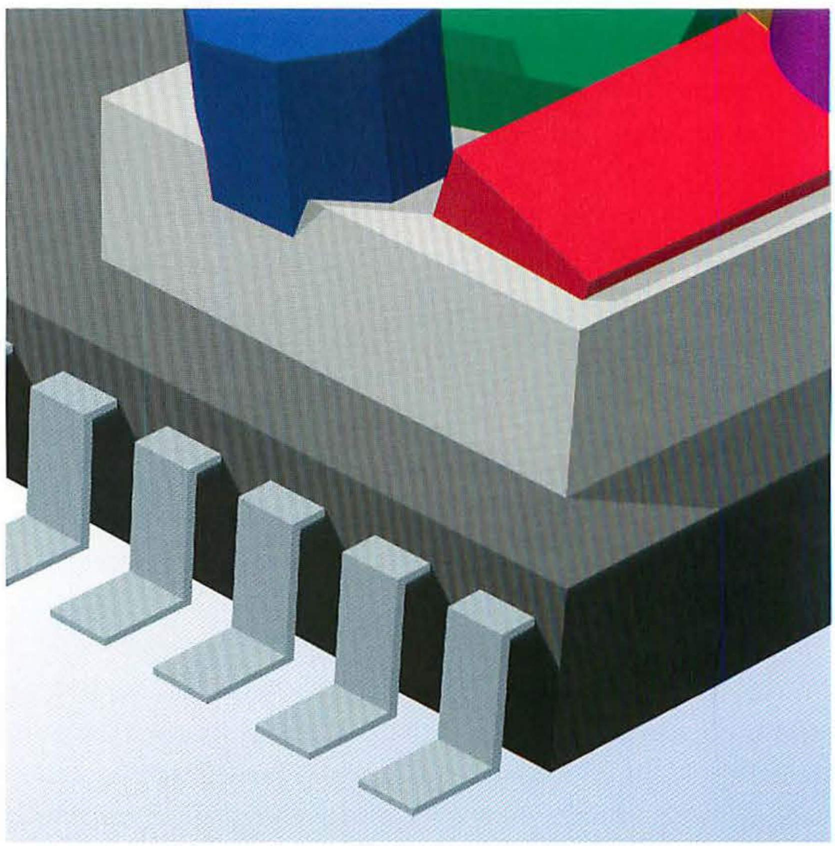
Another radical departure from past versions of Mac OS is the removal of System software dependence on the Apple ROM. Mac OS 8 will use Toolbox routines contained in the System folder, which Apple will rename the Mac OS folder. The PowerPC Reference Platform—the specification for Macintosh-compatible machines that can also run other operating systems—(see Chapter 3) calls for only a small ROM that contains startup code. Mac OS 8 also adds *hardware abstraction*, a programming technique that lessens the dependence of Mac OS on specific hardware.

Before the PowerPC Reference Platform existed, you couldn't build a Mac-compatible PC without licensing the Apple ROM. By moving the Toolbox code out of ROM and providing hardware abstraction in Mac OS 8, Apple is encouraging any company to produce Mac-compatible personal computers without getting Apple's specific approval. You may someday be able to build your own Mac-compatible machine with off-the-shelf parts, just as you can now do with PCs.

Despite radical differences between Mac OS 8 and System 7.5, the new version keeps the basic philosophy that every version of Mac OS has ever had. For one, you never have to tell your Mac when you add new hardware, such as a new hard disk, expansion card, or monitor—Mac OS recognizes it and knows how to communicate with it. You also never have to interact directly with the basic operating system code, such as the routines that deal with input/output and file management, as one does with other operating systems. The Mac has no equivalent to the DOS mode of Windows. There are no config.sys, autoexec.bat, or win.ini files to edit when you install new software, as there are in Windows. Most of Mac OS is invisible. What you don't see not only won't hurt you, but can help a great deal.

CHAPTER 5

How the Mac ROM Works



SINCE the first Macintosh became available back in 1984, the Macintosh ROM (read-only memory) has held the system code that defines the Mac look and feel. The Mac ROM is a set of chips containing 2–4MB of short routines grouped into three categories: the Macintosh Toolbox, the Macintosh operating system, and in Power Macs, the 68LC040 emulator. The software in the System folder serves to support, extend, and update the ROM code.

The Mac Toolbox comprises most of the code in the Mac ROM. Toolbox routines are grouped together into functional sets called *managers*. Software applications use the Toolbox managers to create windows, menus, dialog boxes, sound, and other aspects of the Mac interface. For example, QuickDraw, the graphics manager, draws the text and graphics you see on screen. Because every application uses the same set of managers to create the interface, most Mac applications work the same way.

When Apple programmers create new managers, they usually add the managers to the System file first, often moving them into ROM with later Mac models. To the applications, however, the location of the managers doesn't matter. This means the same application can work on different models of Macs that use different versions of ROM.

Applications also make calls to the operating system, which is the most basic functional level of Mac OS. The operating system performs such tasks as allocating RAM (random-access memory) to an application, controlling the movement of data to and from hard disks and communications ports, and enabling applications to open, quit, and save files.

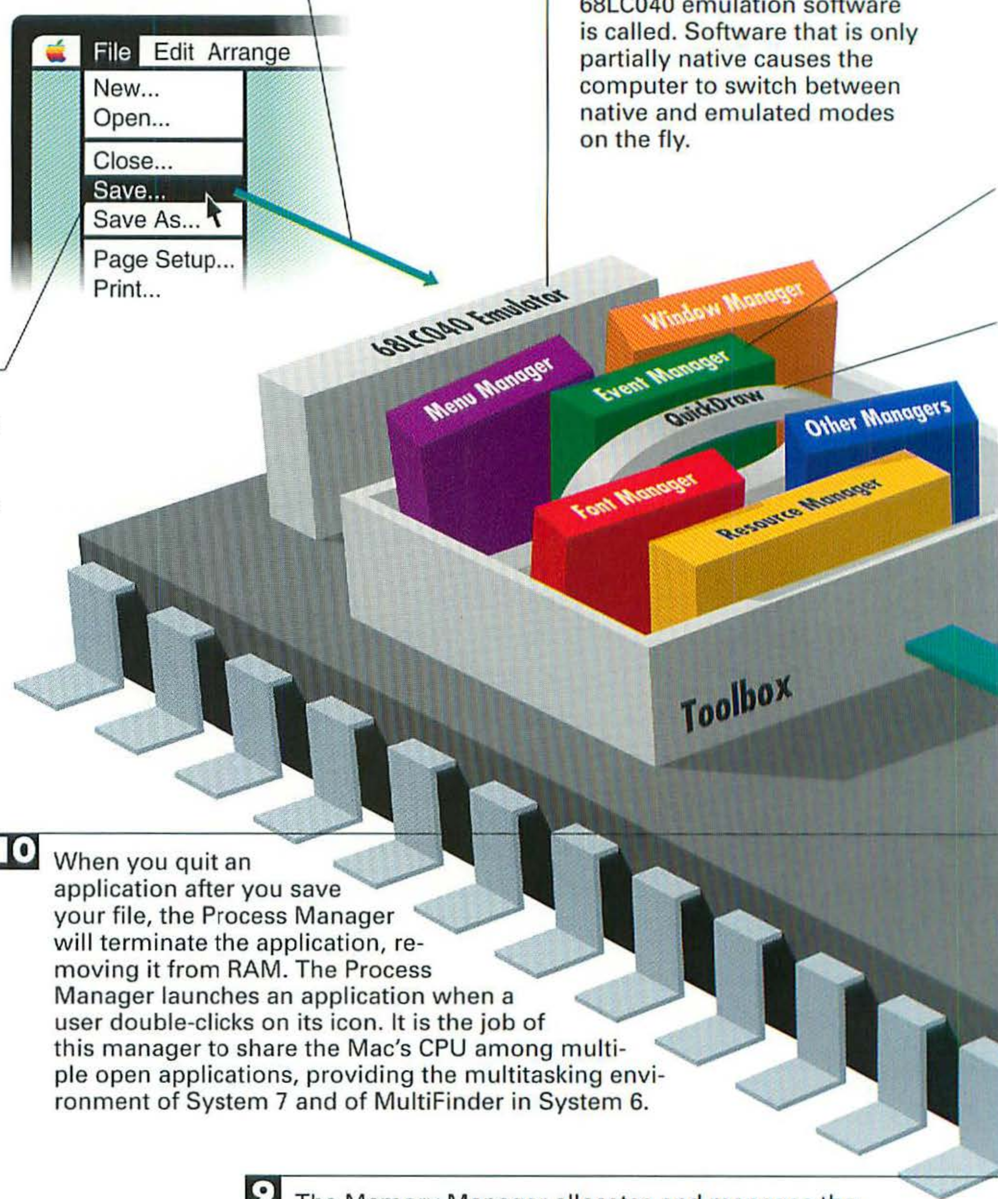
The other major part of ROM, the emulator, creates a virtual 68LC040 CPU chip in software to enable applications that were written for 680x0-based Macs to run on Macs with PowerPC processors. Power Macs have two sets of Toolboxes: one written for use by 680x0 applications and another with certain key managers written for the PowerPC processor.

The ROM chips used to be what made the Mac the Mac. They were Apple's crown jewels, as only Apple produced them. While the Mac ROM is still the heart of Macintosh and Mac-compatible computers running System 7.*x* or older, the ROM required for Mac OS 8 machines will contain only the code required to start up the Mac. Mac OS 8 will place the Toolbox and emulator code in the Mac OS folder (the Mac OS 8 name for the System folder). This will make new Macs relatively less expensive, since the Mac OS 8 ROM chips will be smaller. In addition, Apple will no longer be the sole supplier and will permit third-party vendors to produce the chips.

The Mac ROM

2 The application doesn't have to know exactly where in ROM the procedures are located: The application calls a table, which then directs it to the ROM routines. This way, the ROM in each new model of Macintosh can be different without affecting software compatibility.

3 If the Mac uses a PowerPC processor and the application is not Power Mac native, the 68LC040 emulation software is called. Software that is only partially native causes the computer to switch between native and emulated modes on the fly.



1 When the user enacts a command in an application, such as pulling down a menu and selecting the Save command, the application makes calls to the Macintosh Toolbox in ROM, which in turn triggers a chain of procedures.

MAC FACT The Mac ROM often contains extra space, which Apple sometimes fills with nonessential data that can be accessed by programmers in the know. Sometimes this material is fun stuff, and other times it is experimental data that Apple is considering for future models. The ROM of the first Mac Classics contained an entire System folder that could be used to start up the Mac without the use of a hard or floppy disk. The key combination that invoked this hidden System folder at startup, ⌘-option-x-o, was disabled after a few months of production.

10 When you quit an application after you save your file, the Process Manager will terminate the application, removing it from RAM. The Process Manager launches an application when a user double-clicks on its icon. It is the job of this manager to share the Mac's CPU among multiple open applications, providing the multitasking environment of System 7 and of MultiFinder in System 6.

9 The Memory Manager allocates and manages the portion of memory used by an application. When an open file is saved, the Memory Manager lets the other managers know where in RAM the file can be found. The memory allocation is constantly changing, depending on what the user is doing within an application.

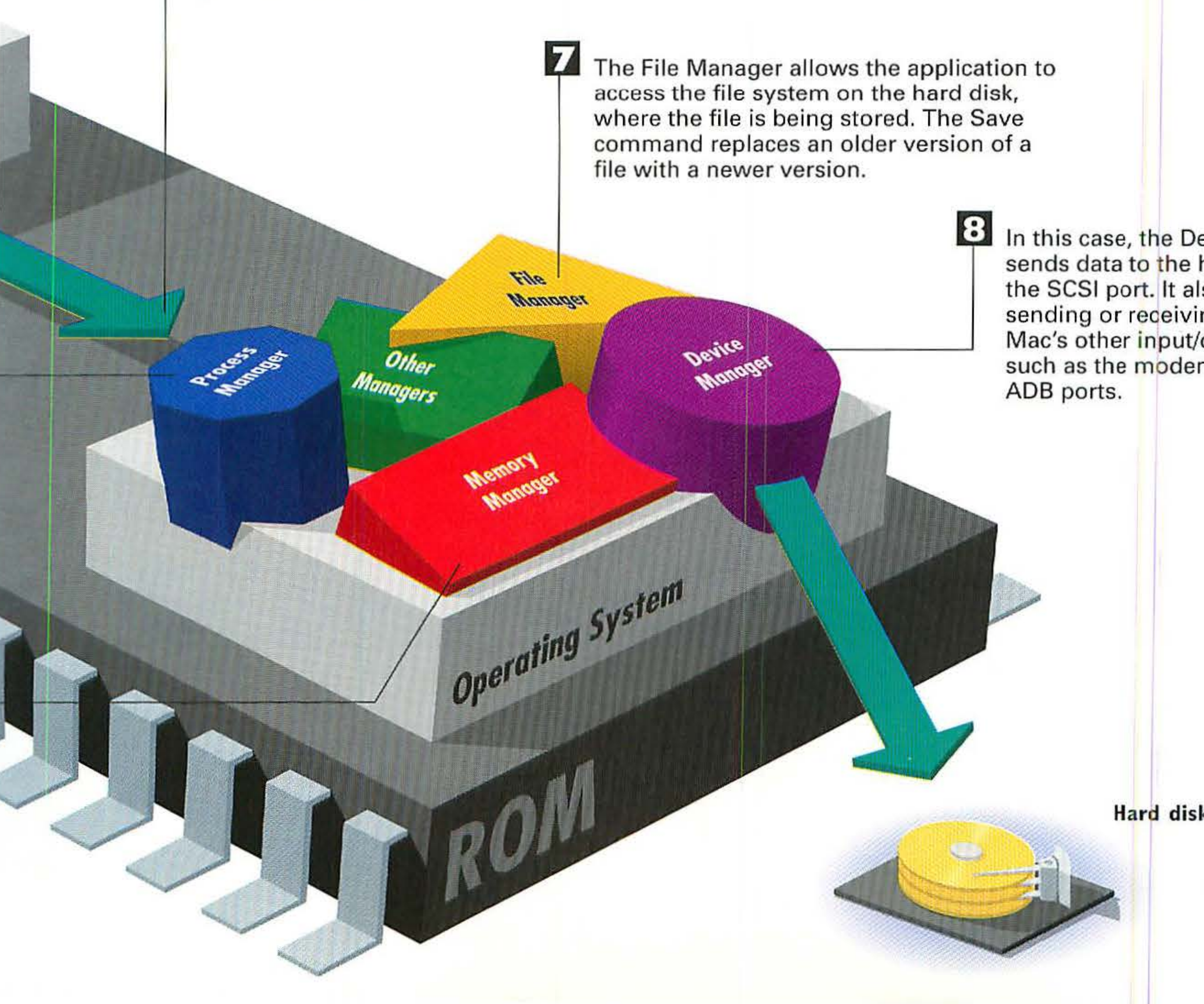
4 A series of Toolbox routines called managers are enacted. This drawing depicts some of the main managers, but the Toolbox contains many others. The Menu Manager handles how a menu works when the user pulls it down and selects a command. The Window Manager keeps track of multiple windows open on the desktop. The Resource Manager allows the application to read and write system resources, such as the fonts, which reside on the startup disk.

5 QuickDraw, the Mac's graphics controller, displays the cursor and draws the menu on the screen while erasing the part of the screen behind the menu. QuickDraw is also responsible for drawing the text, graphics, windows, and everything else on the screen.

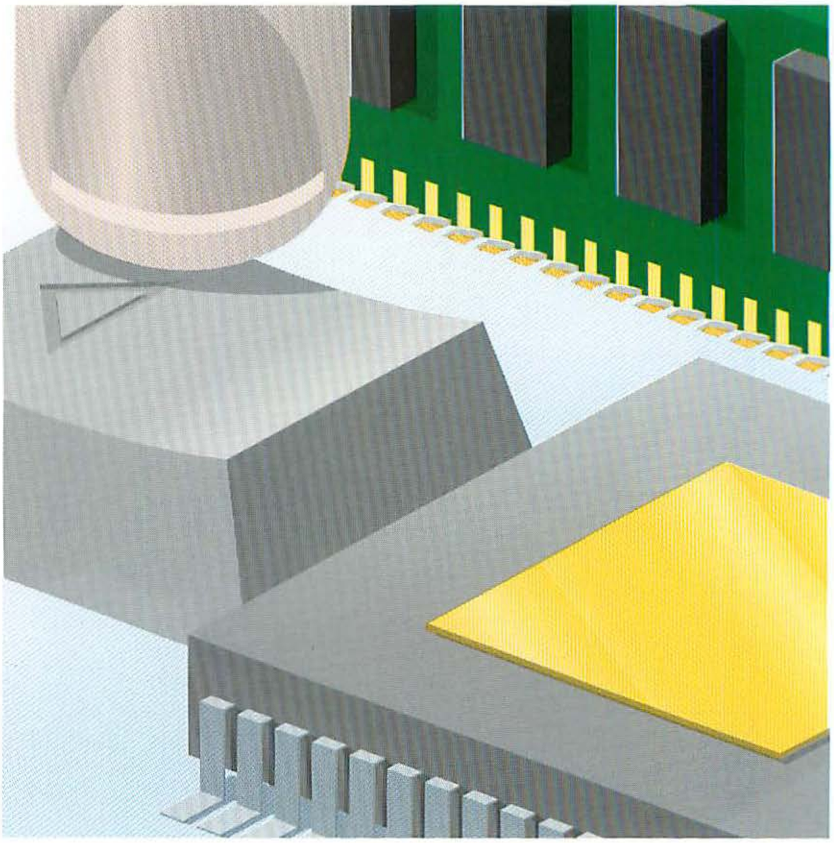
6 Toolbox routines make calls to the operating system, telling it to interact with the Mac hardware. In this example, the Toolbox tells the operating system to save a copy of the file onto the hard disk.

7 The File Manager allows the application to access the file system on the hard disk, where the file is being stored. The Save command replaces an older version of a file with a newer version.

8 In this case, the Device Manager sends data to the hard disk through the SCSI port. It also handles the sending or receiving of data to the Mac's other input/output ports, such as the modem, printer, and ADB ports.



CHAPTER 6 How Startup Works



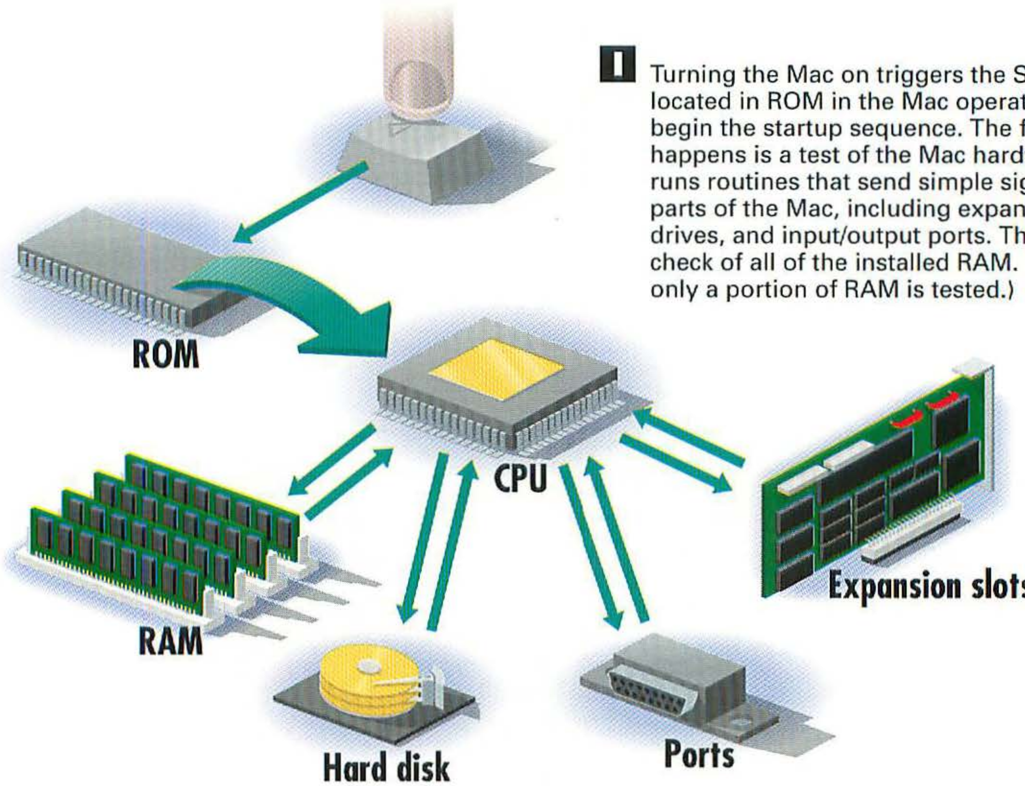
ONE of the busiest times for the Mac is during the first few moments after you turn it on, when there is a flurry of activity that involves most of the Mac's components. It is during startup that the Mac creates the software structures and procedures of its functioning universe.

The creation of the Mac's universe during startup begins with a sort of big bang—a flash of energy that triggers an expanding chain of events. And like the big bang studied by physicists, a lot of the key events that determine the rules by which the system will operate occur in the first few moments. By the time you start to see something happening on screen, the Mac has already checked itself out and established its operating environment.

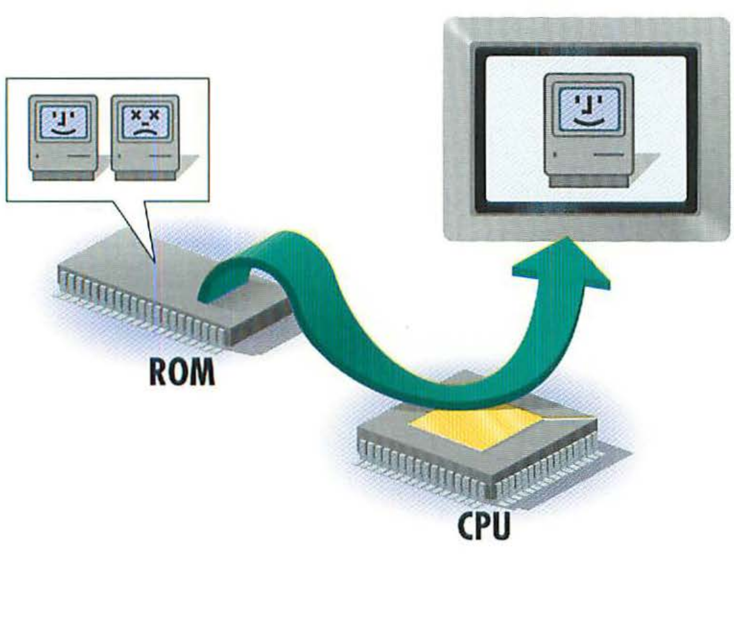
Because everything in RAM is erased when you turn off a Mac, much of the startup procedure involves loading routines and information into RAM, so that the CPU can access them when needed. The Macintosh operating system is loaded from the Mac ROM. The system extensions that add functionality to the operating system are loaded from the System folder contained on a floppy disk, hard disk, or other type of disk storage device. But before anything can be loaded into RAM, the Mac must find and test its own hardware configuration.

The startup procedure described in this chapter is based on Systems 7.0 and later. Startup for Macs running System 6 or earlier is similar, except that *system extensions* (called inits in pre-System 7 terminology) and *control panels* (formerly known as cdevs) are at the root of the System folder instead of in their own subfolders within the System folder.

Mac Startup Procedure

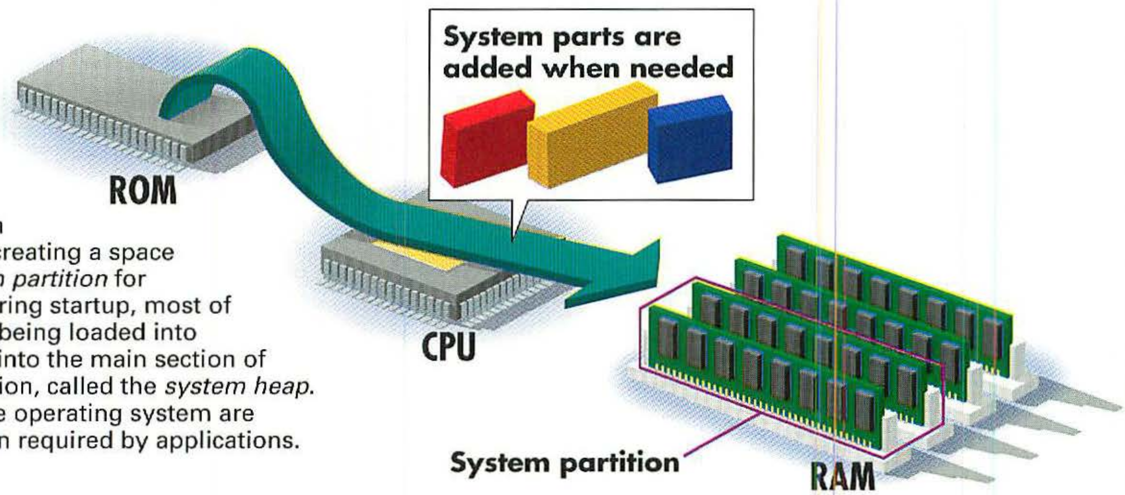


1 Turning the Mac on triggers the Start Manager—located in ROM in the Mac operating system—to begin the startup sequence. The first thing that happens is a test of the Mac hardware. The CPU runs routines that send simple signals to various parts of the Mac, including expansion slots, disk drives, and input/output ports. The last test is a check of all of the installed RAM. (During a restart, only a portion of RAM is tested.)

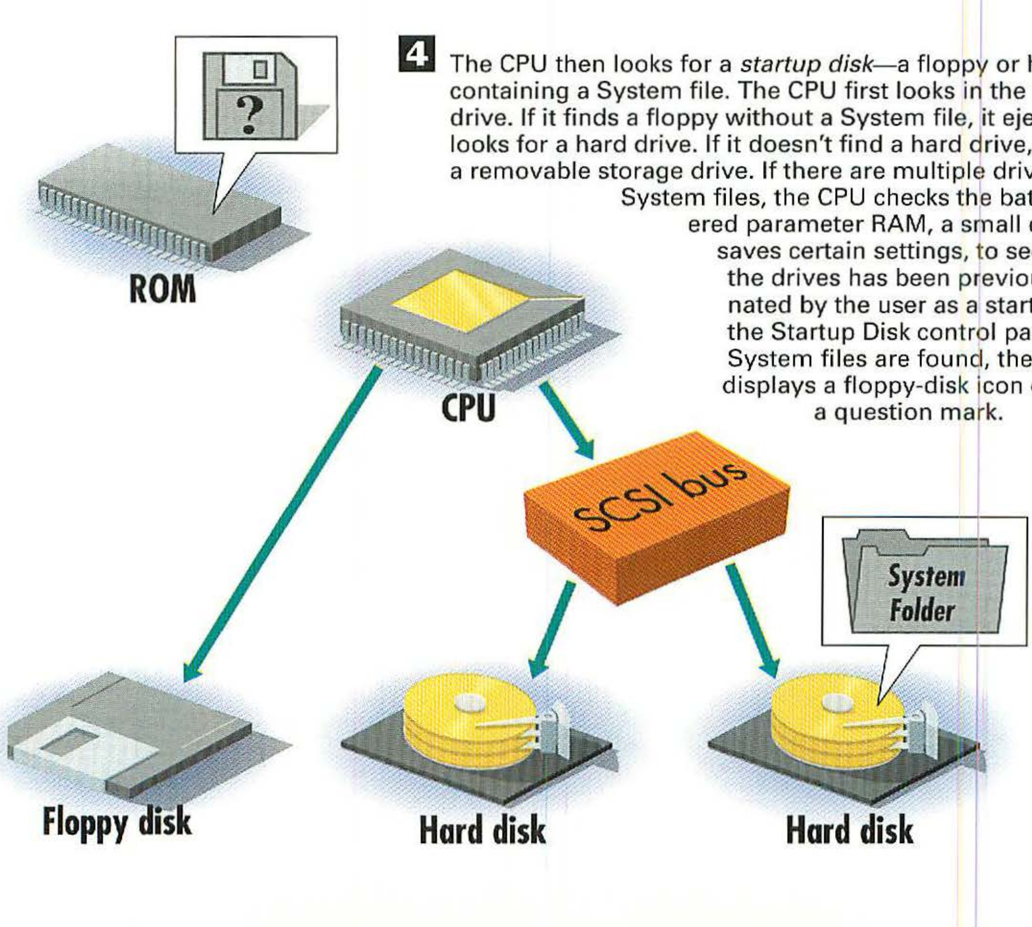


2 If a hardware problem is found, such as a bad RAM chip, the Mac displays a sad Mac icon and plays the failed startup sound, a musical arpeggio. If no hardware problems are found, the Mac displays a happy Mac icon and plays the successful startup sound (a beep in older Macs, a chord in PowerBooks and later Macs). The icons and sounds are stored in ROM.

3 The Mac operating system starts up in full, creating a space called the *system partition* for itself in RAM. During startup, most of the components being loaded into RAM are loaded into the main section of the system partition, called the *system heap.* Other parts of the operating system are loaded later when required by applications.

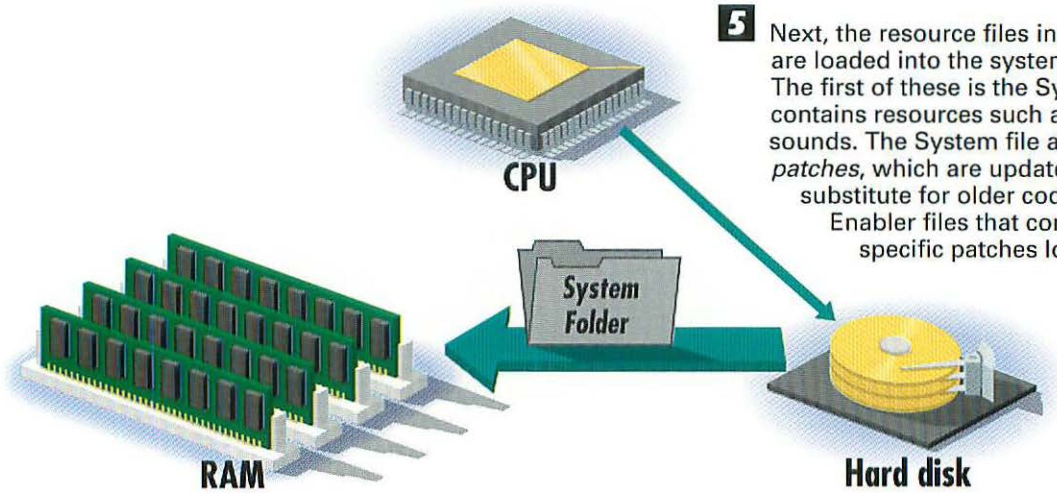


4 The CPU then looks for a *startup disk*—a floppy or hard disk containing a System file. The CPU first looks in the floppy-disk drive. If it finds a floppy without a System file, it ejects it and looks for a hard drive. If it doesn't find a hard drive, it looks for a removable storage drive. If there are multiple drives with System files, the CPU checks the battery-powered parameter RAM, a small chip that saves certain settings, to see if one of the drives has been previously designated by the user as a startup disk via the Startup Disk control panel. If no System files are found, the monitor displays a floppy-disk icon containing a question mark.

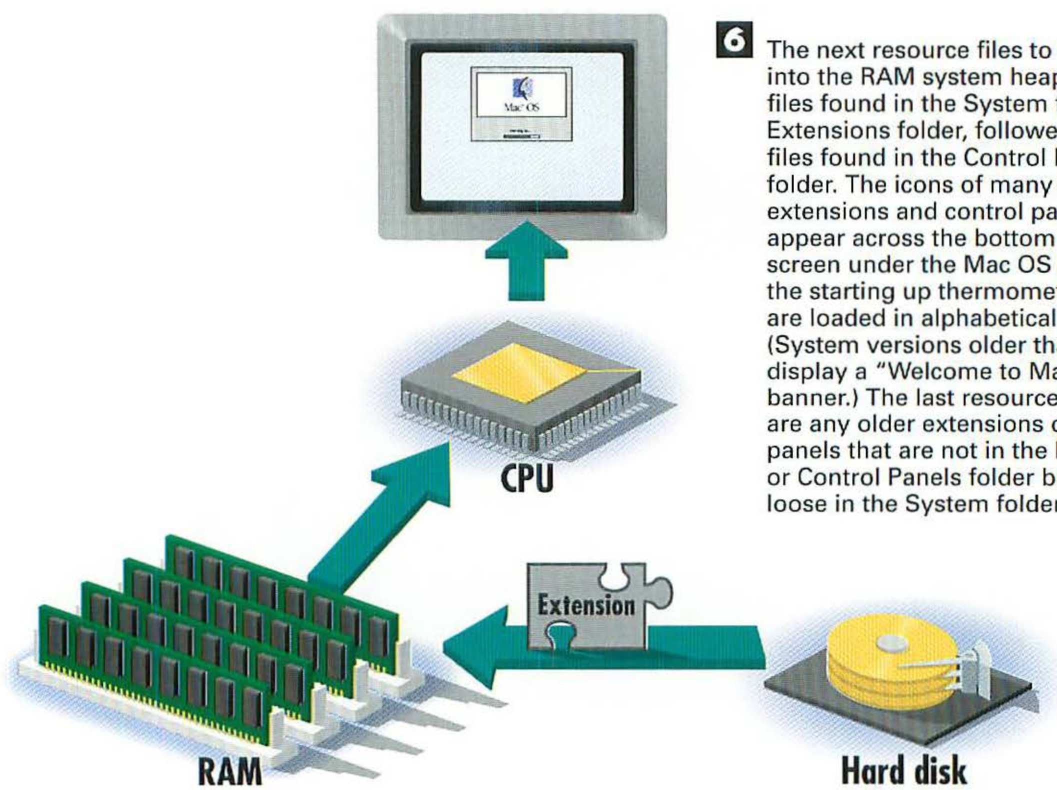


Mac Startup Procedure

(Continued)

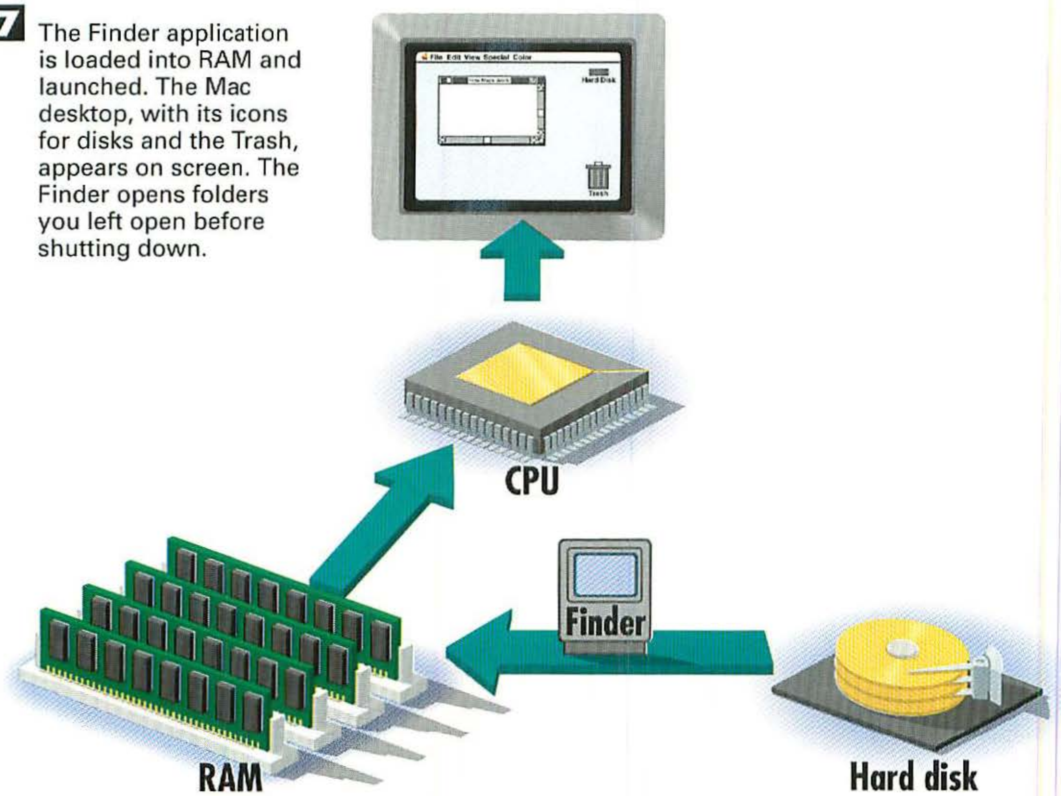


5 Next, the resource files in the System folder are loaded into the system heap in RAM. The first of these is the System file, which contains resources such as basic icons and sounds. The System file also contains *ROM patches*, which are updated routines that substitute for older code written in ROM. Enabler files that contain model-specific patches load at this time.

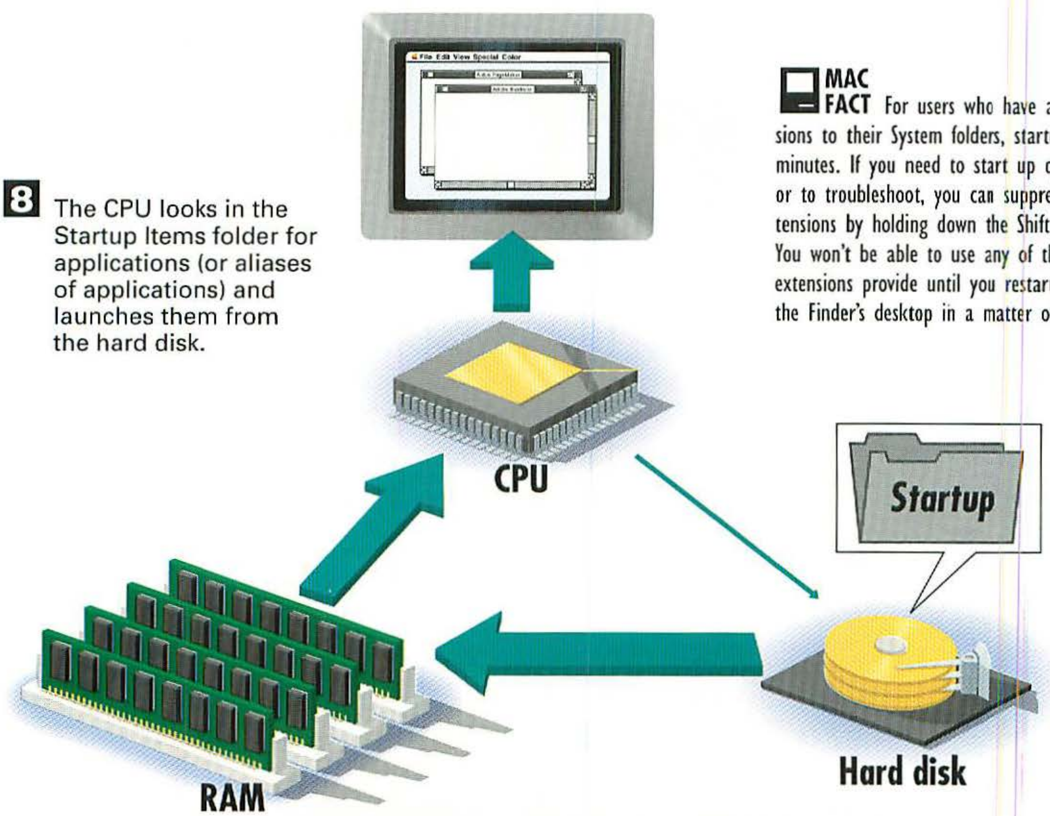


6 The next resource files to be loaded into the RAM system heap are files found in the System folder's Extensions folder, followed by the files found in the Control Panels folder. The icons of many system extensions and control panels appear across the bottom of the screen under the Mac OS icon and the starting up thermometer as they are loaded in alphabetical order. (System versions older than 7.5.1 display a "Welcome to Macintosh" banner.) The last resources loaded are any older extensions or control panels that are not in the Extensions or Control Panels folder but are loose in the System folder.

7 The Finder application is loaded into RAM and launched. The Mac desktop, with its icons for disks and the Trash, appears on screen. The Finder opens folders you left open before shutting down.



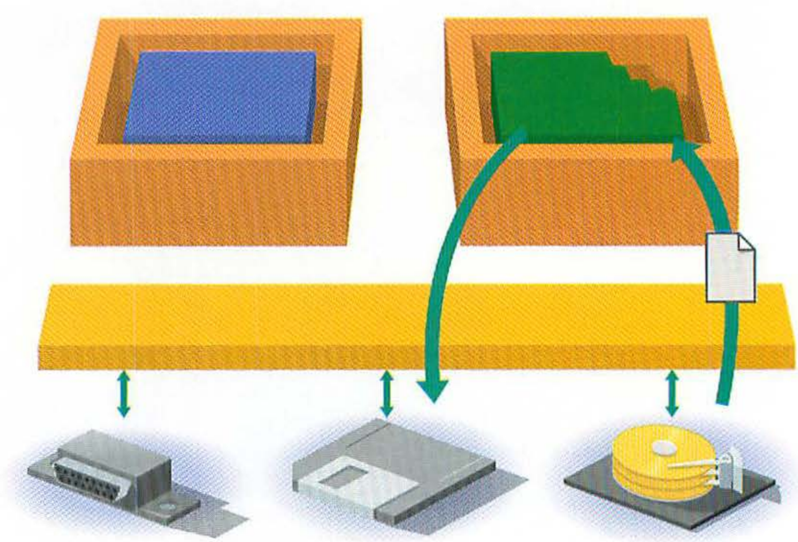
8 The CPU looks in the Startup Items folder for applications (or aliases of applications) and launches them from the hard disk.



MAC FACT For users who have added a lot of extensions to their System folders, startup can go on for many minutes. If you need to start up quickly to grab a file or to troubleshoot, you can suppress the loading of extensions by holding down the Shift key during startup. You won't be able to use any of the features the system extensions provide until you restart, but you will get to the Finder's desktop in a matter of seconds.

CHAPTER 7

How the System Works



ALTHOUGH the Mac's ROM supplies the know-how to run software and create the Mac user interface on screen, the System file supplies many of the interface's attributes. These attributes are called *system resources.* System resources—which include sounds, menus, dialog boxes, scroll bars, icons, and cursors—are shared by all applications, including the Finder. There are also specific resources in applications, such as menu titles and menu items used by all documents of that application, and in document files, which contain information such as the location of the document window on screen. System, application, and document resources, which are stored on disk, are all managed by the Resource Manager in the Mac Toolbox in ROM.

You can customize your collection of sound resources and foreign language keyboard layouts by dropping new ones into the System file. Old resources can be removed by dragging them to the Trash. However, sounds and keyboards are just the tip of the resource iceberg; there are dozens of other types of resources that can be customized, but only with a program such as Apple's ResEdit. You replace many of your Mac's resources every time you upgrade your system or application software.

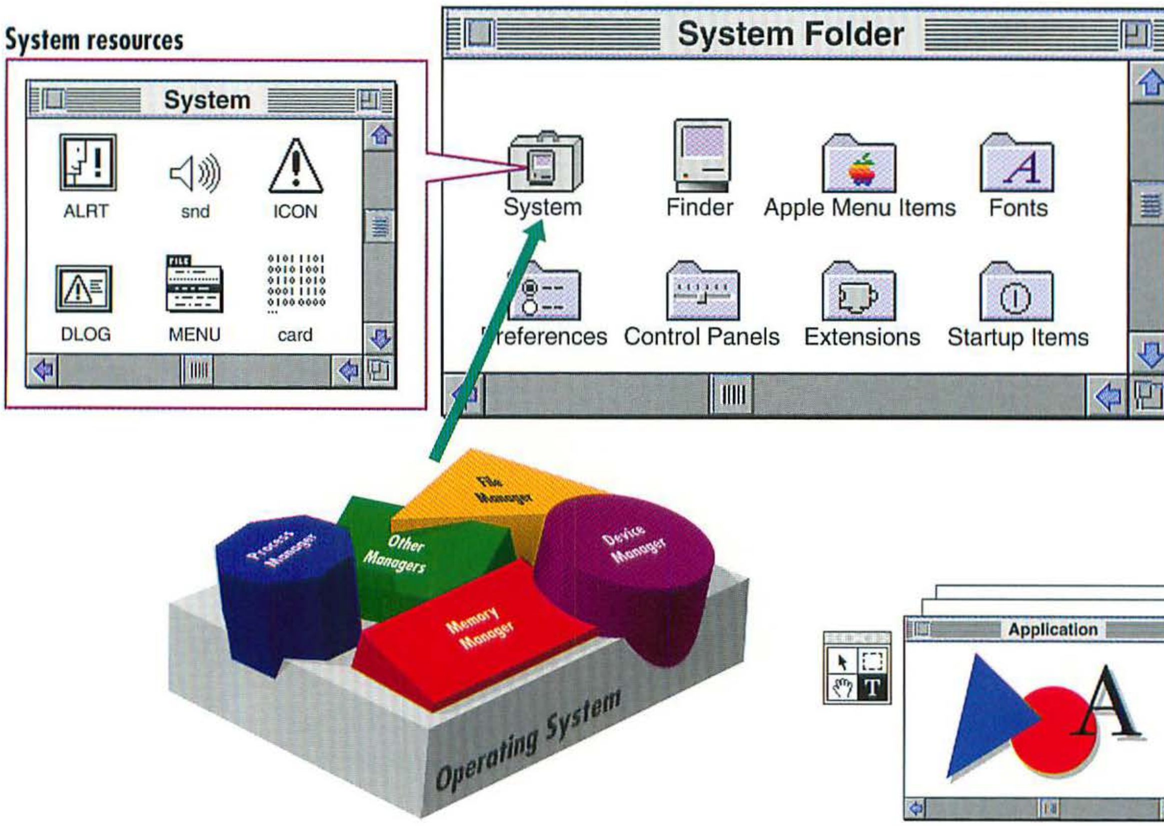
Although most of the System file's disk and RAM space is taken up by resources, the system also contains routines that augment the code in the Mac Toolbox. Starting with System 7, Apple added Toolbox-like managers that can add features to all applications, but aren't in the ROM Toolbox. Examples include the Program-to-Program Communications Toolbox and the Apple Event Manager, which allows applications to exchange data and to control each other.

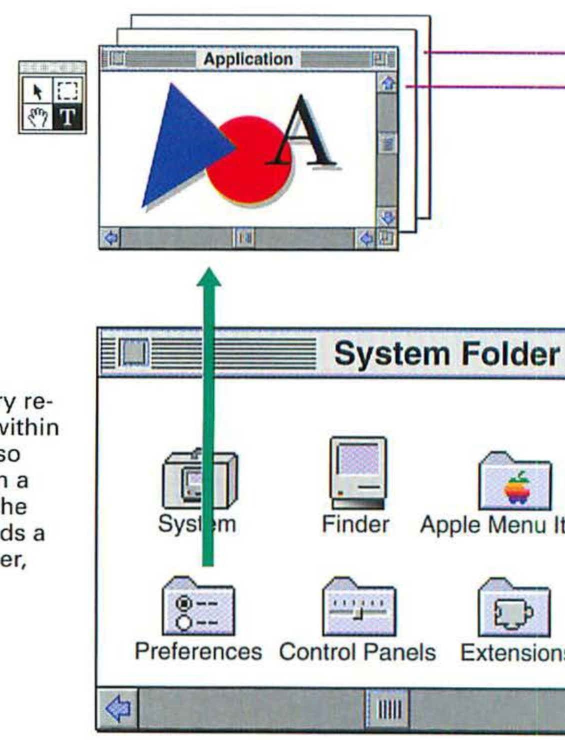
The system software is designed to allow users to add significant new capabilities by dropping files called *system extensions* into the Extensions folder, which is inside the System folder. Apple's system extensions offer new classes of functions to all applications without completely rewriting the System file or creating a new ROM chip. Examples of system extensions include QuickTime for the display of video and PlainTalk for understanding your voice. Application extensions written by third-party software manufacturers add features to a single program; system extensions are external additions to the Mac Toolbox and enable *any* application to make calls to the new managers. The open design of Mac OS allows for the future addition of system extensions not even conceived of today.

Mac OS 8 will use extensions and resources faster and with fewer system crashes by adding a central controller called the Microkernel. Unlike the ROM's operating system code, which is a decentralized collection of routines, the Microkernel is a commander-in-chief that schedules what software has access to the CPU and RAM. The Microkernel also manages a feature called *preemptive multitasking*, which allows several tasks to occur at the same time. Preemptive multitasking will let you simultaneously copy a file to a floppy disk, send another file over a network, render a complex graphics file, and type in a word processing program without interruption.

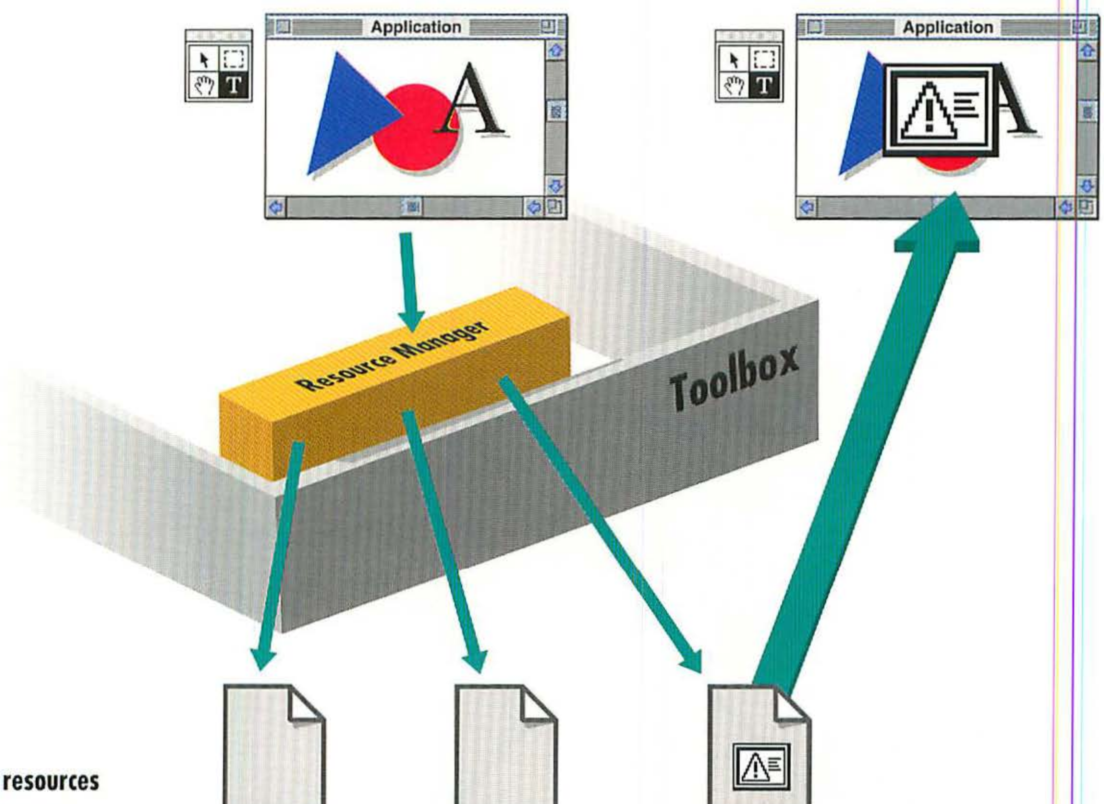
The System File and Resources

1 At startup, the operating system opens the System file, making the system resources available to applications. This process is completely invisible to the user.



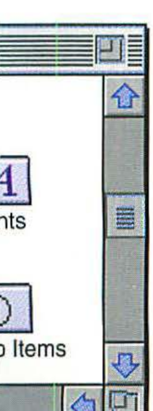


2 Applications can have their own proprietary resources, which are shared by documents within the application. Individual data files can also have resources, such as preferences. When a document opens, the application looks in the System folder's Preferences folder and reads a preference file for settings saved by the user, such as default fonts and margins.



Document resources
Application resources

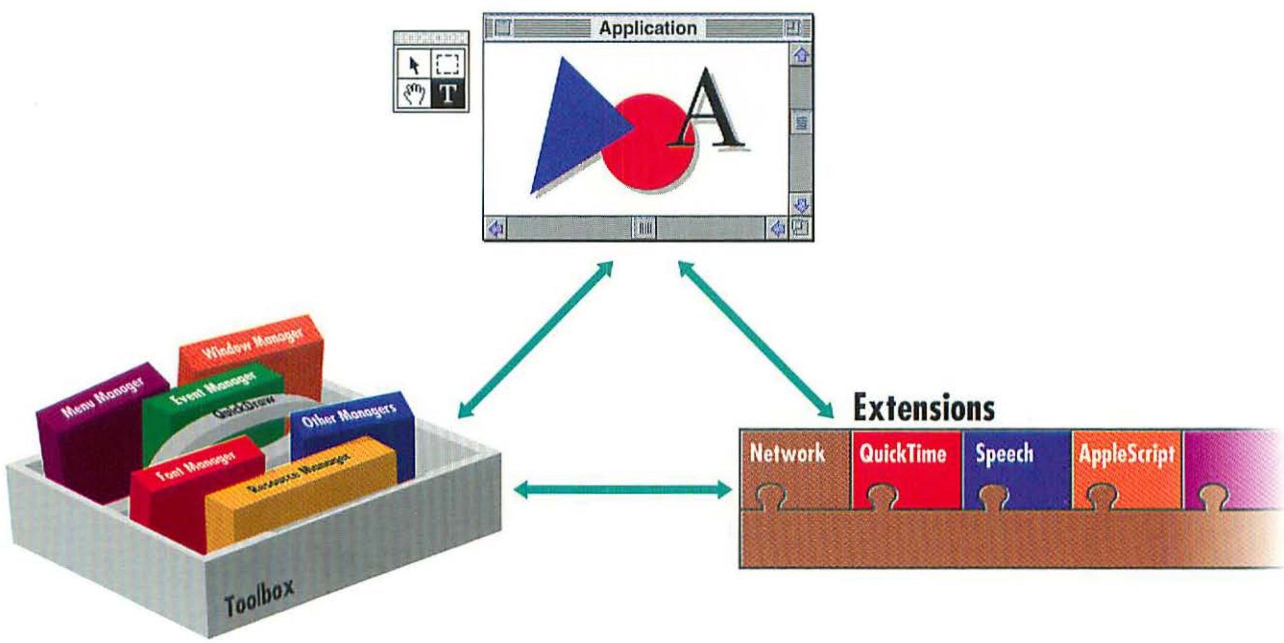
3 When an application needs a resource, such as a particular dialog box, the Resource Manager in the Toolbox typically searches through various resource files. Starting with the most recently opened resource files, it looks first in document files, then in application files, and finally in the System file, stopping any time it finds the requested resource. Applications can also bypass the Resource Manager and specify particular resources for certain uses.



MAC FACT In addition to over a hundred different types of resources contained in the System file, there are lines of code that fix errors (or bugs) in the ROM. These are called *ROM patches,* and they are loaded into RAM at startup. ROM patches intercept calls from applications and the operating system and run the new code instead. Thus, the ROM in old Macs can be upgraded by running new system software.

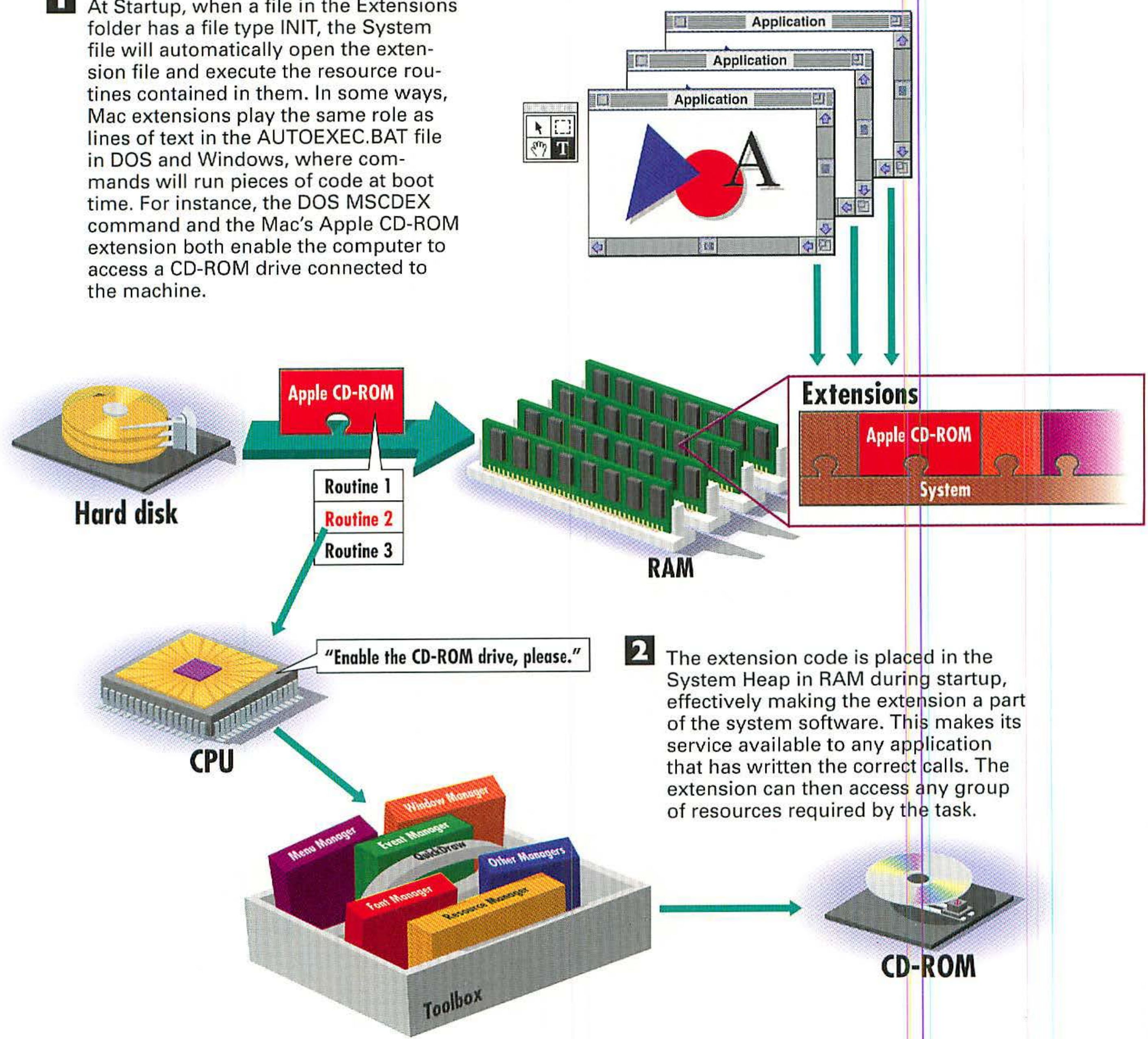
System 7 Extensions

System extensions—which are the equivalent of new Toolbox routines—can be added to the Mac to provide applications with new classes of functionality. Application programmers can use the new features by adding a few lines of code to their programs to call the new routines. Users add the new functionality by dropping one or more system extension files into the Extensions folder inside the System folder.



MAC FACT System extensions sometimes follow a migratory path, starting as an add-on extension, then later being included as a built-in part of the system software, and finally becoming an actual Toolbox routine in ROM. 32-bit color QuickDraw, which started out as an extension, was included in the Mac ROM starting with the Mac IIci.

1 At Startup, when a file in the Extensions folder has a file type INIT, the System file will automatically open the extension file and execute the resource routines contained in them. In some ways, Mac extensions play the same role as lines of text in the AUTOEXEC.BAT file in DOS and Windows, where commands will run pieces of code at boot time. For instance, the DOS MSCDEX command and the Mac's Apple CD-ROM extension both enable the computer to access a CD-ROM drive connected to the machine.

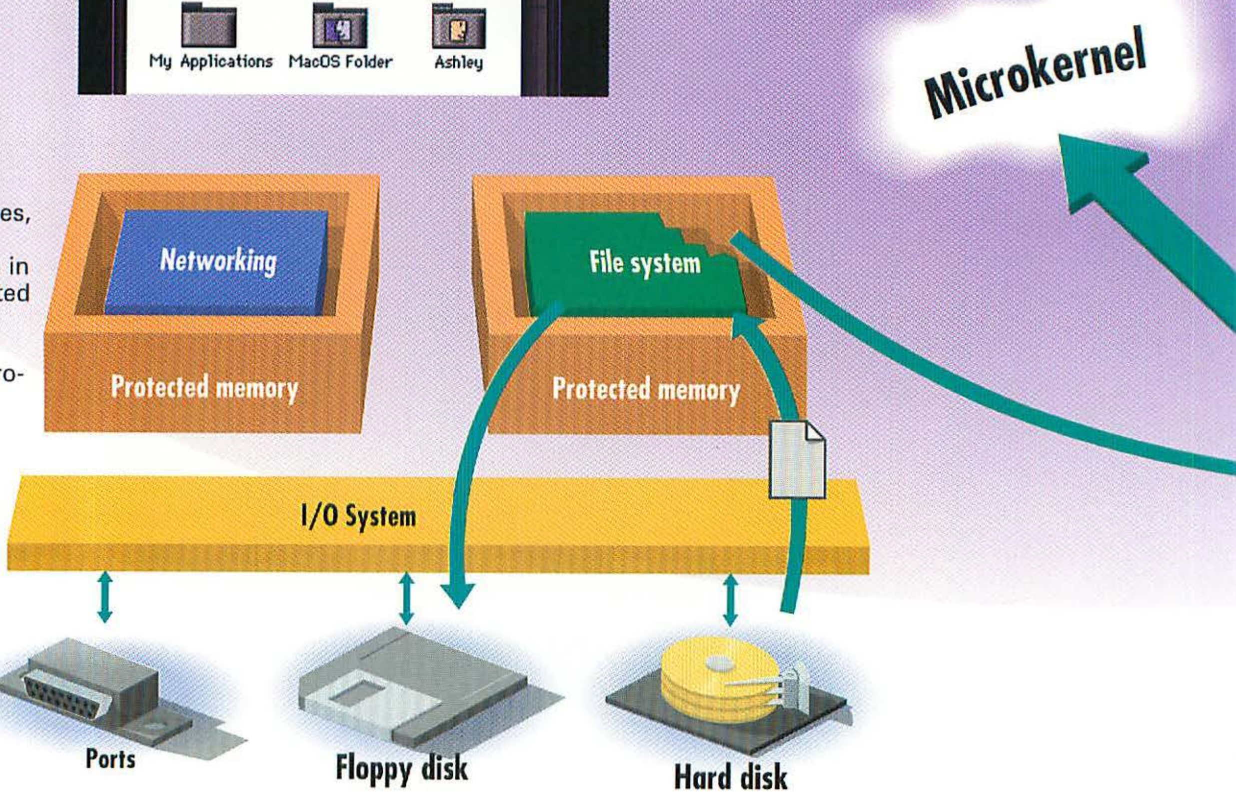


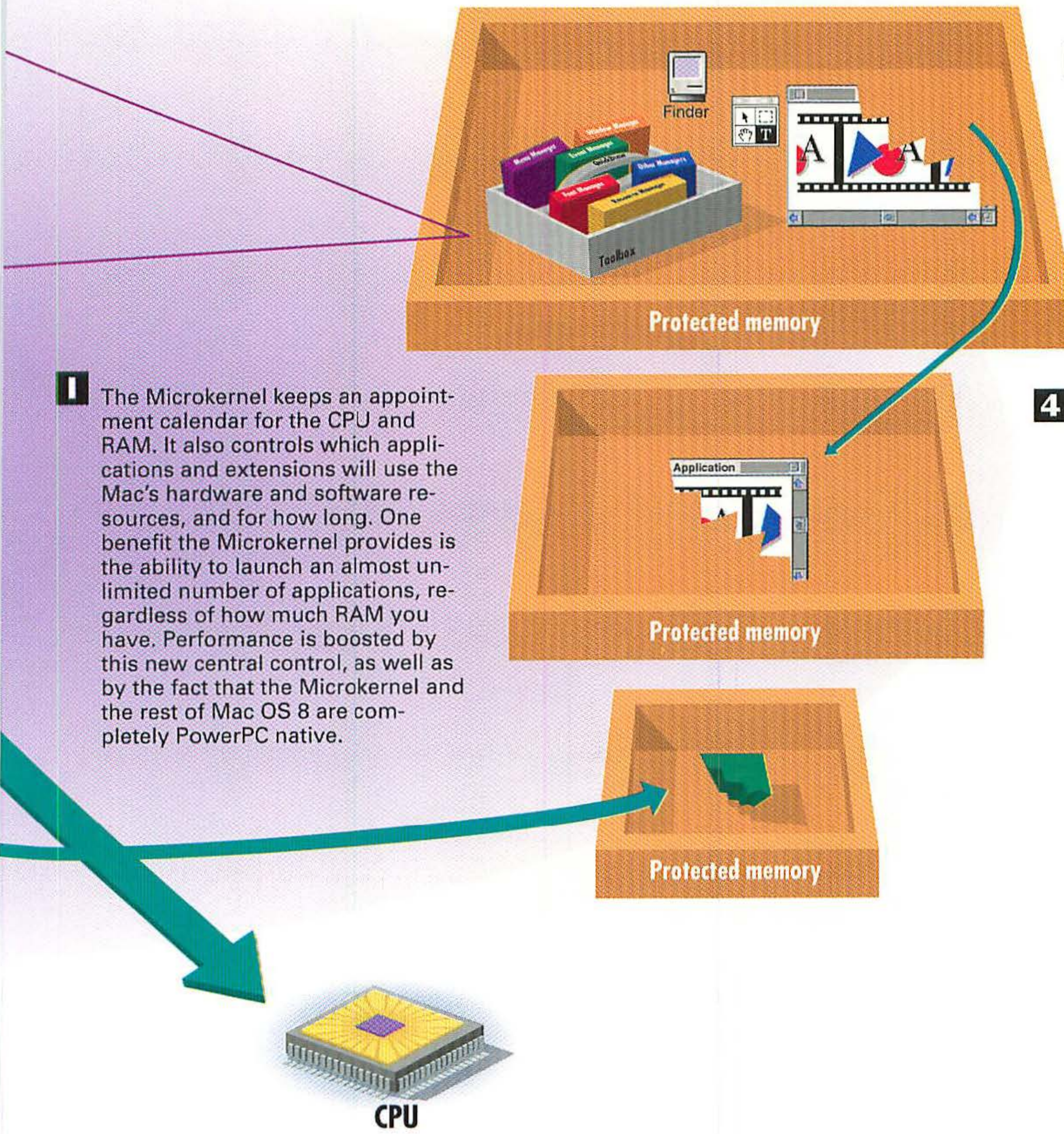
2 The extension code is placed in the System Heap in RAM during startup, effectively making the extension a part of the system software. This makes its service available to any application that has written the correct calls. The extension can then access any group of resources required by the task.

Mac OS 8, Mac OS Reinvented

6 Mac OS 8 will let you choose or create your own desktop styles, which not only look different from each other but can also sound and act differently. You can use a simplified desktop style for children to hide power-user features, turn files and folders into buttons, and add animated menus that play sounds. Business users can adopt a look and feel that displays the features they want most, perhaps keeping sounds to a minimum. In this way, each member of the family or business using the same machine can have his or her own Mac interface.

5 Network processes, files, and the I/O process each run in their own protected memory spaces, and can spin off tasks into new protected memory spaces.





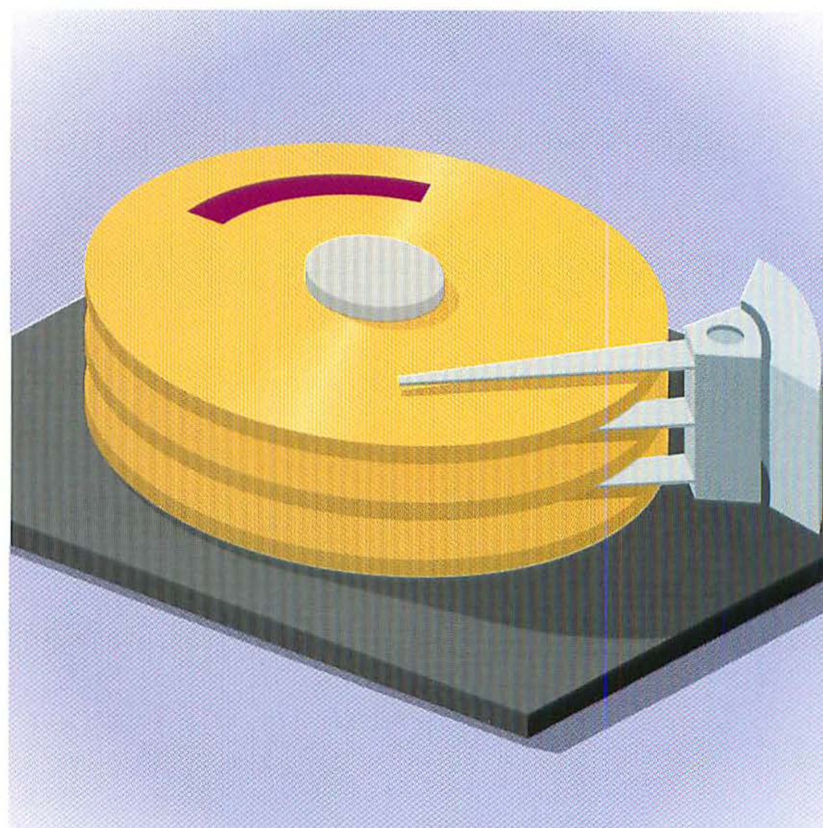
1 The Microkernel keeps an appointment calendar for the CPU and RAM. It also controls which applications and extensions will use the Mac's hardware and software resources, and for how long. One benefit the Microkernel provides is the ability to launch an almost unlimited number of applications, regardless of how much RAM you have. Performance is boosted by this new central control, as well as by the fact that the Microkernel and the rest of Mac OS 8 are completely PowerPC native.

2 Preemptive multitasking is a new feature that lets the Mac do several tasks at once. The code for each task runs in a completely separate, protected memory space. This will let you save a document to a floppy disk at the same time you type in a word processor. If the software in one protected memory space crashes, it won't affect the rest of the Mac, and you won't have to restart your computer.

3 Applications, the Finder, and the Toolbox managers all reside in one protected memory space. This ensures backward compatibility with older applications.

4 Before Mac OS 8, everything was in one memory space. Mac OS 8 applications can move a task, such as a computation for rendering a graphics image, into its own protected memory space.

CHAPTER 8 How the Finder Works



ALTHOUGH it resides in the System folder, the Finder is technically not a system resource like the System file. Rather, it is a very specialized application that launches automatically at startup and always runs—you cannot quit from the Finder. Mac users run the Finder in lieu of an operating-system mode, which operating systems such as DOS use when no applications are being run. The Finder provides users with the Mac desktop, a view into all the storage devices connected to the Mac. On the desktop, the Finder depicts application and data files as icons or as text lists in windows. It lets you create folders—the equivalent of directories in DOS—to store files or other folders. You also use the Finder to rename and copy files and to delete files by dragging them to the Trash.

The Finder was originally called so by Apple because it locates the files you need on your storage disks. Through the use of invisible databases called the Desktop files, the Finder keeps track of the dozens, hundreds, or thousands of files you may have. Each floppy disk, hard disk, or other storage device mounted on the desktop has its own Desktop files. The Finder consults these Desktop files when you search for a file with the Find command from the File menu.

It is also the Finder's job to *launch* applications—locating the application on a disk, loading it into RAM, and opening the application window on the desktop. To launch a program, you can double-click directly on an application icon or name, or you can double-click on a data file—the Finder can tell which application was used to create a file. The Finder that comes with System 7.0 or later also lets you launch an application by dragging and dropping a file on top of an application. This drag-and-drop method is handy if the file is in a format that can be opened by more than one application. Starting with System 7.5, you can drag text and graphics from certain applications and drop them on the Finder's desktop, creating files.

With the System 7 Finder you can use aliases to access files that are buried deep within several nested folders. *Aliases* are dummy files that are linked to a file. Double-clicking on an alias opens the file it is linked to without having to open any folders.

The Finder also lets you modify your system setup by adding fonts and sounds and changing settings in the control panels. You can even change your files' icons by pasting new ones into the Finder's Get Info box for each file.

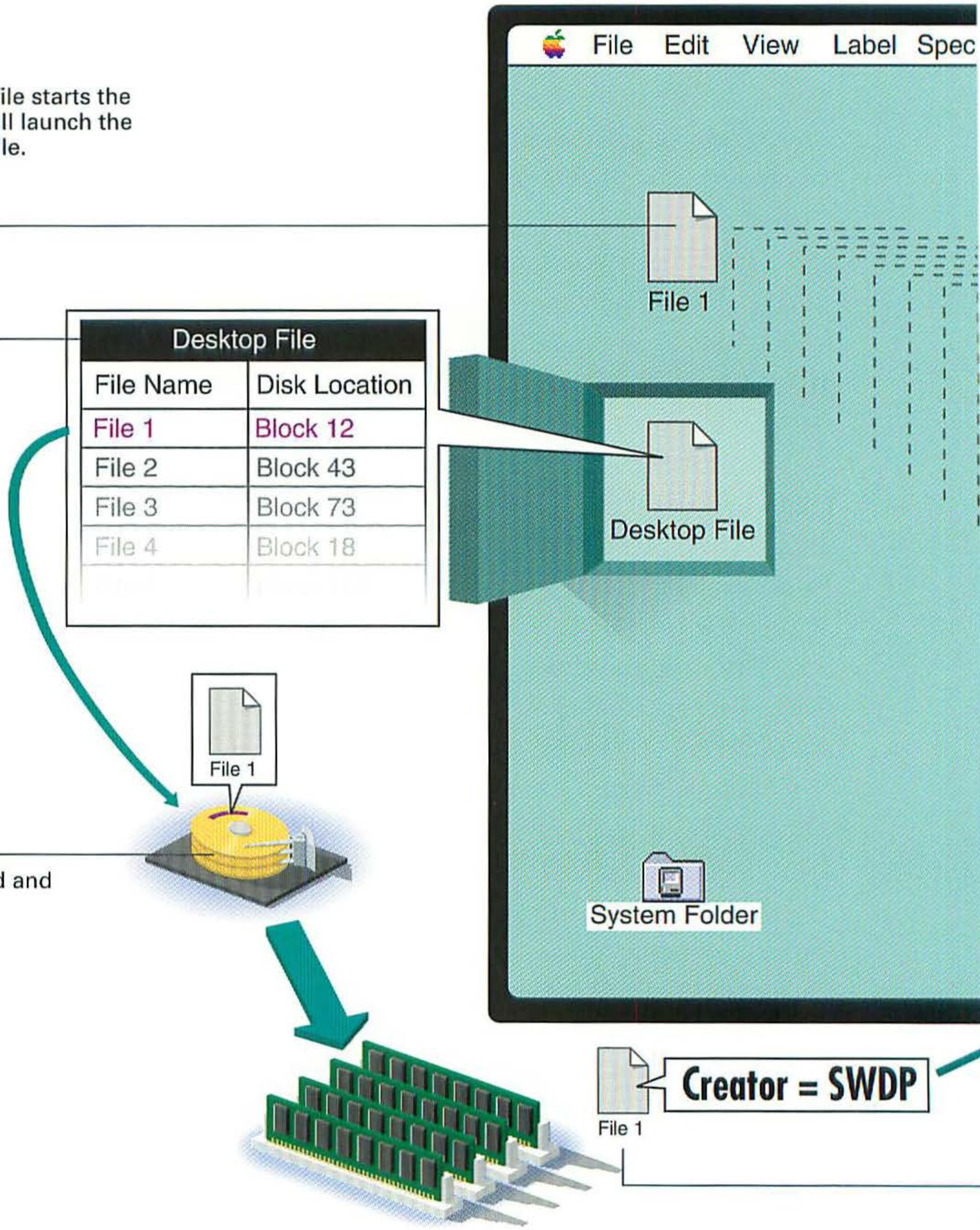
Technically, you don't need a Finder file to run a Macintosh. Many hard-disk utility packages come with an emergency floppy disk you can use to start up the Mac when your hard disk isn't working. This emergency floppy contains only a System file and the hard-disk utility program. Of course, without the Finder, there is no desktop, which is an important part of the Mac experience.

Opening a File in the Finder

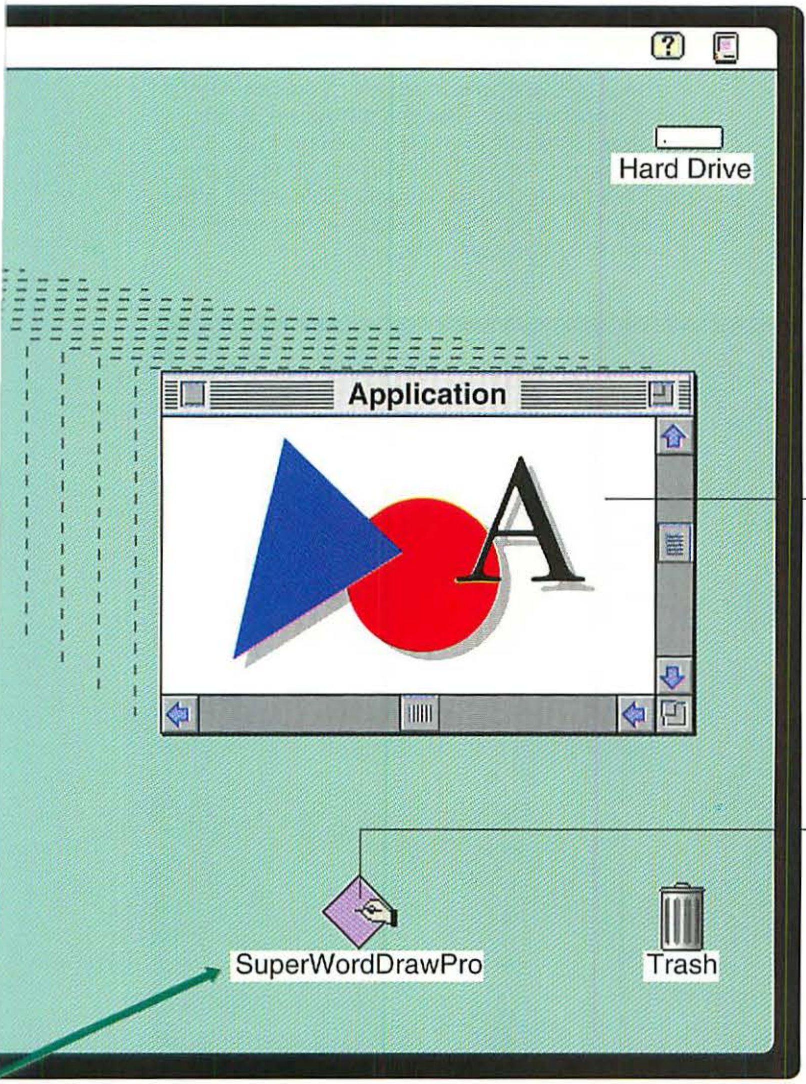
1 Double-clicking on a data file starts the sequence of events that will launch the application and open the file.

2 The Finder invokes invisible Desktop files, which are databases containing the exact location of all the files on a floppy or hard disk. When it finds the location of the file you double-clicked, the Finder makes a call to the disk.

3 The file on disk is accessed and loaded into RAM.



MAC FACT Since a disk's Desktop files change every time you add, delete, copy, or move a file, they can occasionally become cluttered with obsolete information, which slows down Finder performance. They can also become corrupted, which may cause files and folders to intermittently disappear. Fortunately, you can rebuild the Desktop files for each disk by holding down the ⌘ and option keys during startup.



6 The application opens the file that you double-clicked and displays it in a window.

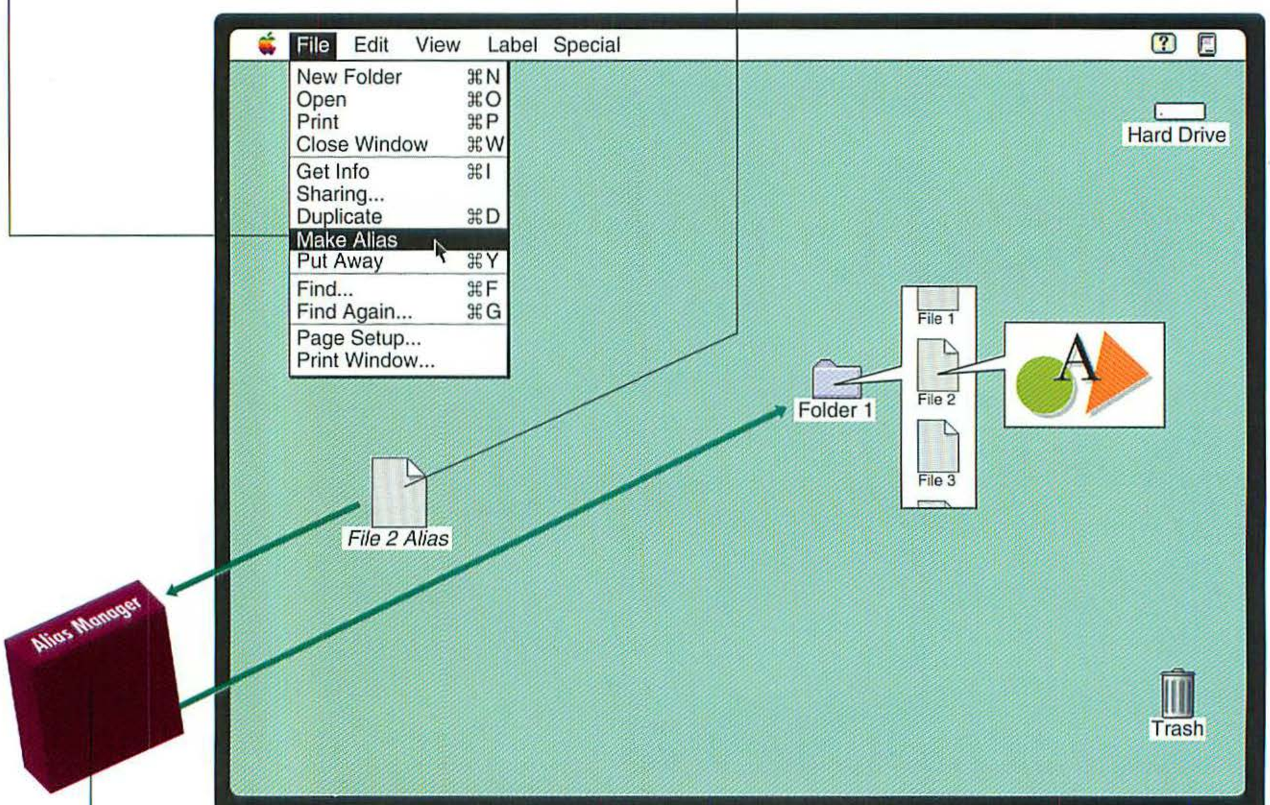
5 The Finder locates the appropriate application on the hard disk, loads it into RAM, and displays it on screen.

4 The Finder reads the file's four-character creator code to determine what application to launch. For instance, the creator code MSWD indicates that the file was created with Microsoft Word. An application called SuperWordDrawPro might have a creator code of SWDP. If there is no application on any disk that matches the creator code, the Finder displays a message saying that the application for this file can't be found. Or, if Apple Easy Open is enabled, the Finder may suggest an alternative program you can use to open the file.

Creating and Running an Alias

1 You create an alias by selecting a file, pulling down the Finder's File menu, and selecting Make Alias.

2 After the Finder creates an alias icon, you can move the alias into any folder, including the Apple Menu Items folder (making it appear in the Apple menu), or you can copy it to another disk. The default name of the alias is the original file's name followed by the word "alias," but you can rename the file to anything you want. You can tell a file is an alias because the characters in the name are always italicized.

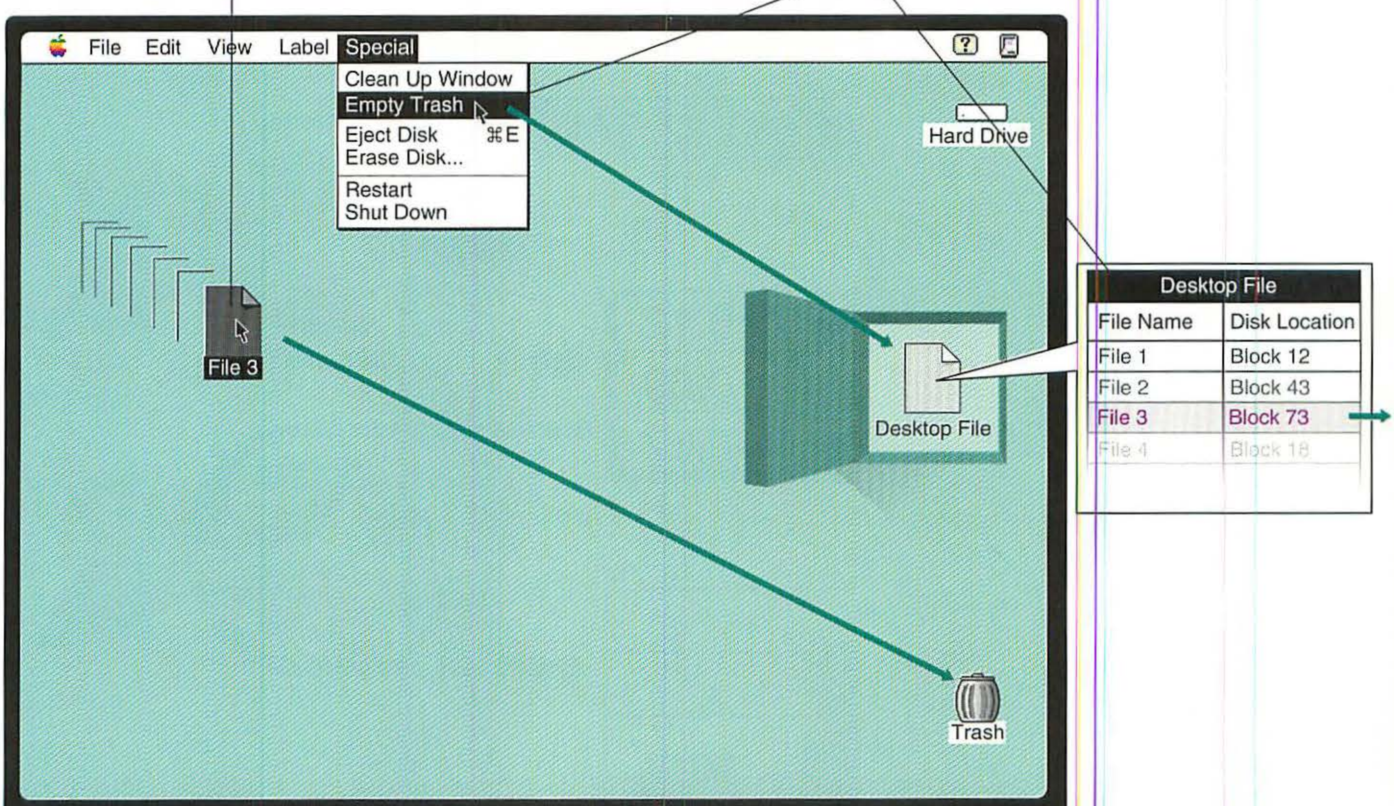


3 When you double-click on an alias, the Alias Manager in System 7 locates the original file and opens it using the normal Finder process (see "Opening a File in the Finder"). The original file will open even if it is buried deep within several folders. If the original file resides on a removable storage device, such as a floppy disk or CD-ROM, the Mac will ask for that device to be mounted (made available) so that the file may be opened.

Deleting a File

1 To delete any type of file, you drag the file to the Trash, which is actually a folder. At this point, the file is still completely intact, but it is designated to be discarded. If an application calls for a file that happens to be in the Trash, the Finder sends a screen message asking you to remove it from the Trash. You can drag a file out of the Trash at any time before you invoke the Empty Trash command.

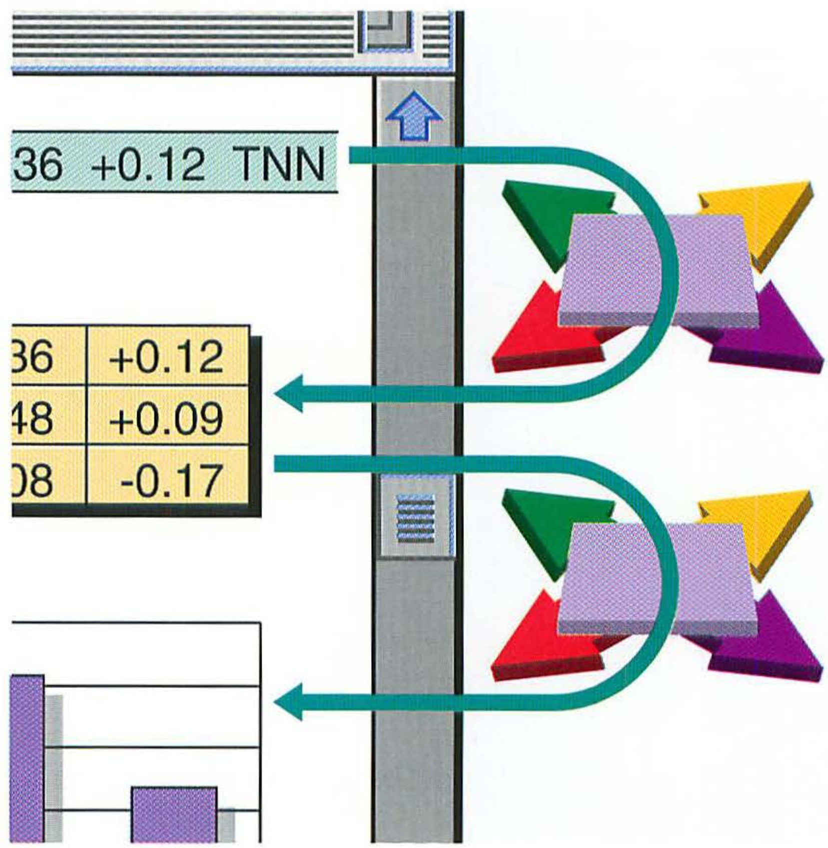
2 When you select Empty Trash from the Special menu, the Finder deletes the file's entry from the Desktop file, and the file's icon disappears from the Trash. Although the Finder can no longer locate the file, it still exists on the disk and can be recovered with file recovery software. However, the file is no longer protected from being overwritten, and it may be partially or totally erased from the disk the next time you save a file.



MAC FACT Although System 7 and MultiFinder in System 6 both allow multiple application windows to be open at the same time, this was not always the case. Prior to MultiFinder, you had to quit an application before you could go to the Finder or to another application. The first attempt at letting you open multiple applications was a program called the Switcher, but it could only display one application at a time. To switch from one application to another, you pressed an arrow, and the current application slid off the side of the screen while the new application slid in to replace it.

CHAPTER 9

How OpenDoc Works



In the early eighties, a typical software package consisted of a plastic sandwich bag containing a single-sided floppy disk and a photocopied instruction sheet. Today, an application can fill dozens of floppies or a CD-ROM and comes with an encyclopedic set of manuals. You now get hundreds of software features, but only a handful are useful to most people. The rest sit around taking up megabytes of RAM and hard-disk space. If you need to bring together the features in two or three applications, you need two or three times the RAM.

OpenDoc offers a task-centered alternative to the application-centric computing of large software packages. OpenDoc system software allows you to run component software made up of interchangeable, reusable parts. With OpenDoc, you can assemble a collection of features tailored to your needs, or buy a collection of specific features bundled together. For instance, you can create a document containing a text editor, a spreadsheet, a QuickTime movie viewer, an Internet browser, and a telephone dialing pad. You can remove features you don't use, keeping only the features that you need, not those that a software company thought would sell well.

There are two types of OpenDoc software: the container, which is the workspace, and the OpenDoc parts. You can add a feature to a workspace by simply dragging and dropping a part into a container document. Parts can share information with each other, and can create live links across networks and the Internet. You can have multiple active parts running at the same time in one document, gathering information from a mainframe or showing live television.

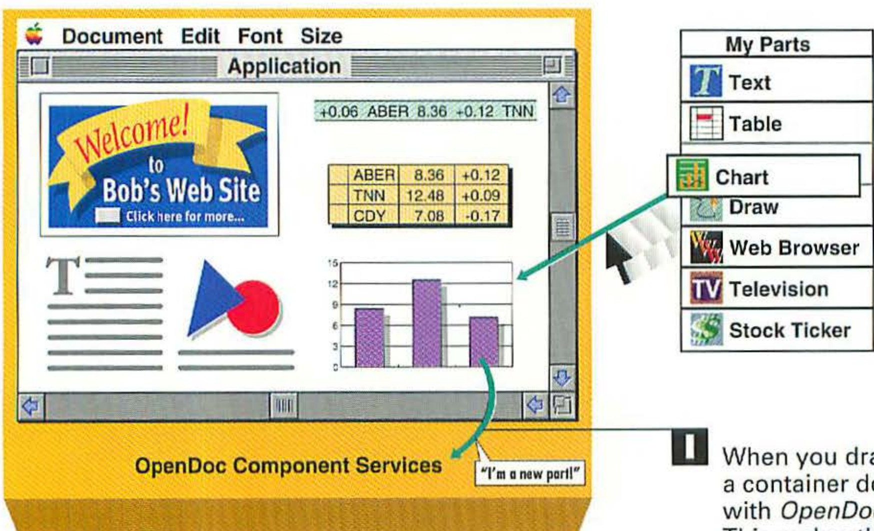
OpenDoc runs on Mac OS, Windows, OS/2, and AIX (IBM's version of UNIX). It is compatible with Microsoft's OLE. OpenDoc is an open standard backed by Apple, IBM, Adobe, and other companies, and is controlled by an industry consortium called Component Integration Labs. Industry-wide support enables OpenDoc parts from different manufacturers to work together. If you don't like Acme's text editor, you can replace it with ABC Company's text editor.

The components of OpenDoc system software come from several different companies, and are just as modular as the user software. For instance, OpenDoc includes IBM's System Object Model (SOM) for linking objects and Apple's Standard Interchange Format (known informally as "Bento") for storing information. However, should something better come along, a developer can replace Bento or SOM with another technology without "breaking" OpenDoc.

Just how many applications will become OpenDoc compatible is uncertain. If Apple's ambitious plans for OpenDoc succeed, component software will someday completely replace all-in-one applications. This would fundamentally change the way the computer industry creates and sells software. However, even partial success for OpenDoc will slow the ever-increasing size of the applications we use.

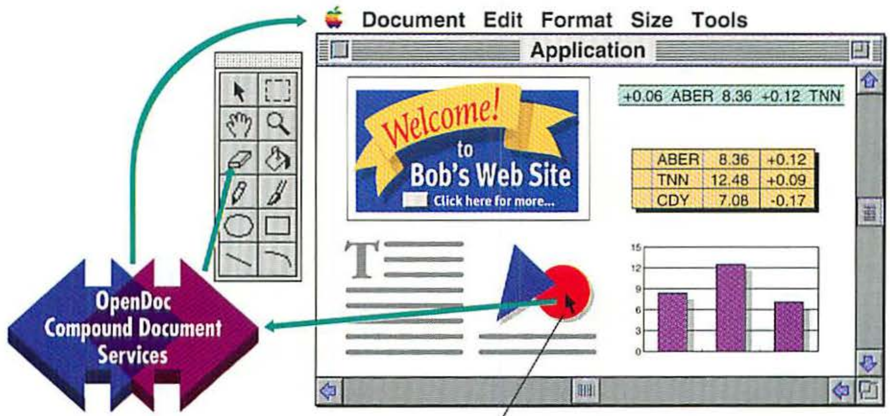
OpenDoc Containers and Parts

2 An OpenDoc part can be a one-function miniapplication, such as a text editor, spreadsheet, or a live link to a stock market database. You can also drag in parts that are attributes, such as a color pattern to drag over a graph, or an Internet address to drag over a World Wide Web browser.

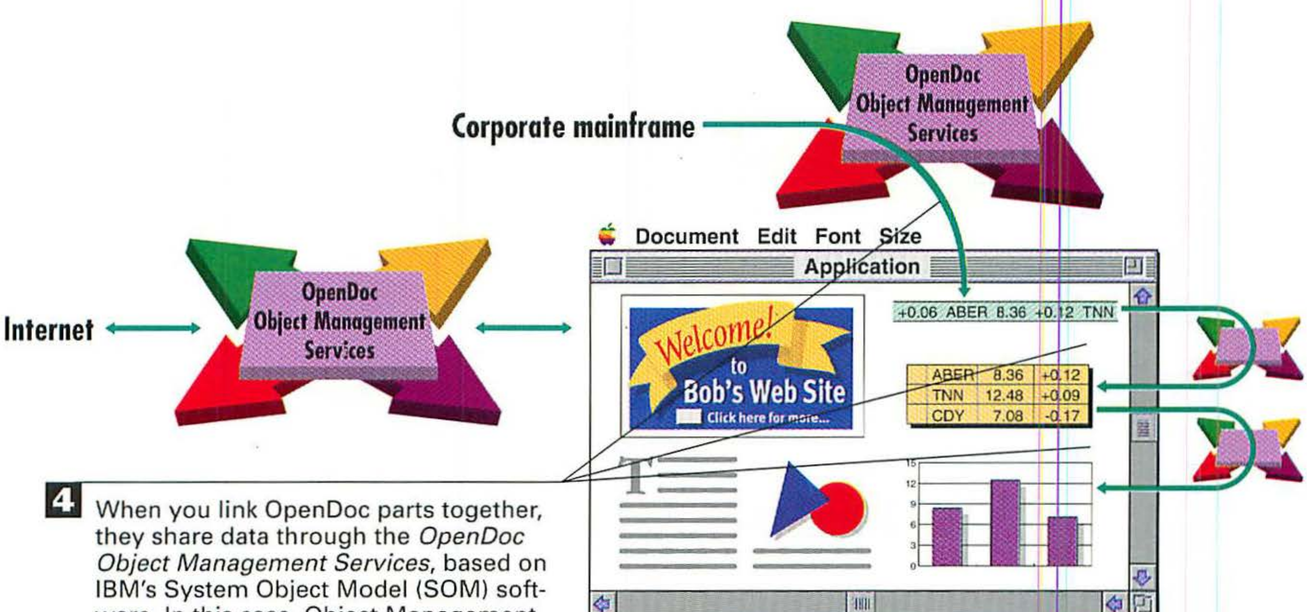


1 When you drag a new OpenDoc part into a container document, it registers itself with *OpenDoc Component Services*. This makes the other parts aware of its existence. Component Services allows parts to work together, sharing code and other resources.

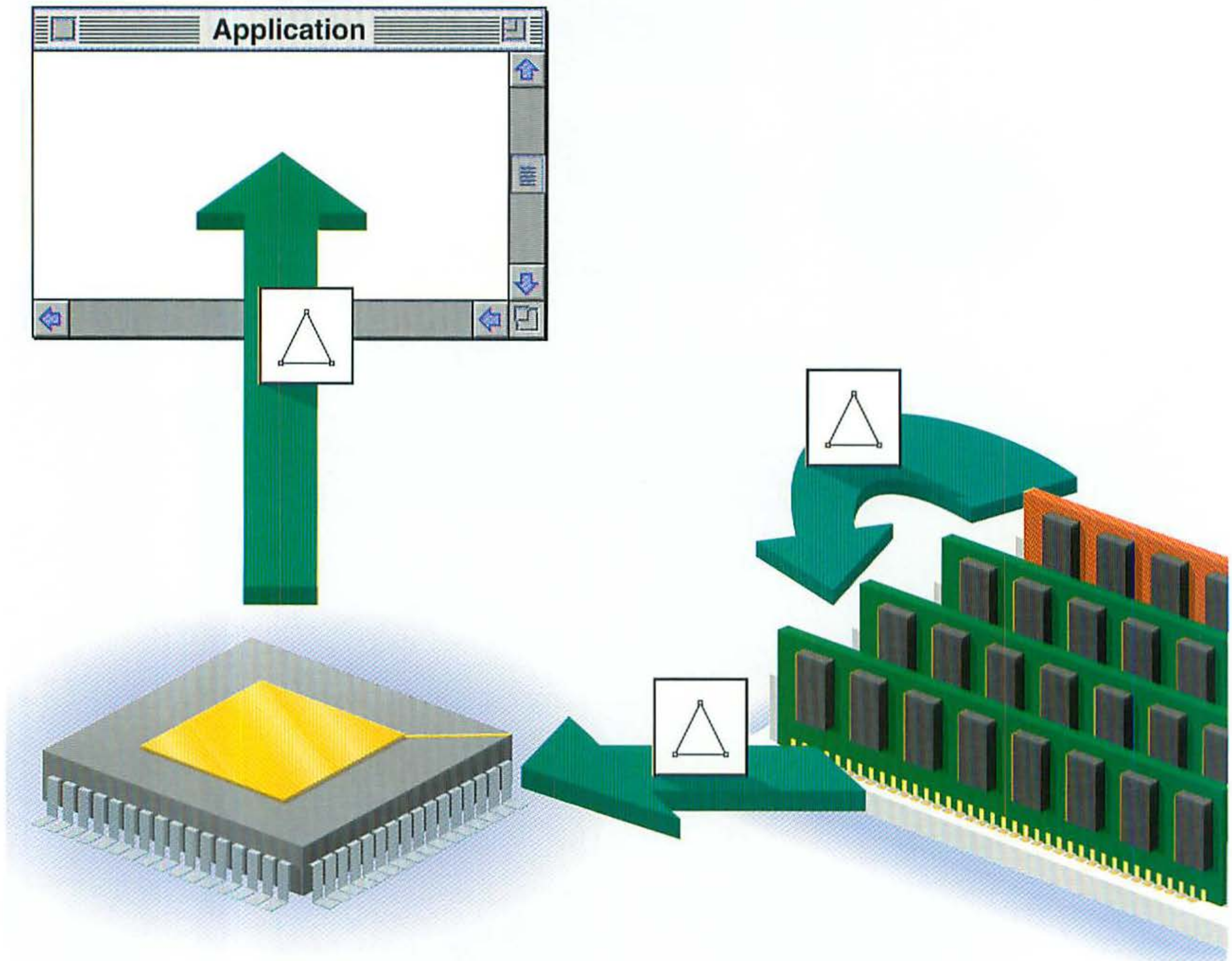
MAC FACT Much of Mac OS 8 itself will consist of OpenDoc containers and parts. This will make it easy for Apple to update pieces of Mac OS without having to rewrite the entire operating system.



3 When you move the cursor over a part, the *OpenDoc Compound Document Services* manages the display and user interface for the particular part. In this case, we've moved the cursor over a draw part, so the Compound Document Services changes the menu to a draw menu and brings up the drawing editor's tool palette.



4 When you link OpenDoc parts together, they share data through the *OpenDoc Object Management Services*, based on IBM's System Object Model (SOM) software. In this case, Object Management Services links the live stock market ticker tape part to the spreadsheet, which links to the bar graphs. All three continuously change as new data comes in from the link to the company mainframe.


Application

PART

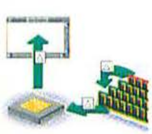
3

CPU AND MEMORY

FOR all the metal, plastic, and glass that make up a computer, just a square inch of silicon does the actual computing. This is the central processing unit (CPU), the computer's engine. Getting its instructions from software, the CPU juggles numbers by the millions to create complex graphics, calculate your spreadsheet results, and process the countless behind-the-scenes activities you can't see.

While CPUs of a few decades ago could fill the good part of a room, the CPU in today's personal computers resides on a single chip called a microprocessor. The microprocessor used in Power Mac models is one of the PowerPC chips jointly developed by IBM, Motorola, and Apple. Macs based on the original Mac CPU use one of the members of the Motorola MC68000 family.

But Macs don't run by CPUs alone. The CPU works as part of a team with the main system memory, which resides on a set of silicon chips called random access memory (RAM). The main system memory supplies the CPU with the software instructions and data it needs to do your work. The system memory receives the information from a permanent storage device, such as a hard disk, and temporarily holds it for the CPU. Computers use RAM because it can pass information to the CPU much faster than a hard disk could.



Functionally, the CPU and main memory are quite different. Software tells a microprocessor chip to perform tasks and to request information from memory, but RAM chips are "unintelligent" storage vessels for information. At a microscopic level, however, microprocessors and RAM chips include the same primary component, the transistor.

Transistors are tiny electronic switches typically measured in microns (millionths of a meter). These simple devices are certainly the single most important electronic device today. Cheap, reliable, and low powered, you can find them in everything from TVs to telephones, from cars to karaoke machines.

In microprocessors, RAM, and any other electronic component, transistors represent *binary numbers*, that is, numbers that use only the characters 0 and 1. A transistor represents 0 and 1 with two electronic states, on and off. A single transistor holding a 0 or a 1 is a *bit*, short for binary digit. A series of transistors (or bits) represents numbers larger than 1.

The RAM that makes up your Mac's main system memory is an array of millions of transistors holding 0's or 1's. The Mac ROM chips mentioned in Chapter 5, "How the Mac ROM Works," are similar to RAM chips. However, the values of 0 and 1 that the transistors hold in ROM are permanent.

A microprocessor chip is quite a bit more complicated than a RAM chip. Instead of merely storing bits, it executes millions of instructions every second based on information it gets from RAM. In a microprocessor chip, engineers combine the input and output signals of transistors in different ways to build tiny adding machines. Get enough of these minute motors together and put them inside a silicon chip and you have an integrated circuit. Microprocessors have many layers of thousands of integrated circuits.

Using integrated circuits, the CPU adds and subtracts binary numbers at dazzling speeds set by the clock rate, measured in megahertz (MHz). This means the transistors inside a 100 MHz microprocessor are opening and closing at a rate of 100 million times per second.

By operating at a speed that is far beyond the realm of everyday human experience, the CPU eventually translates simple math into actions like opening a window or displaying animation on the Mac monitor. Similarly, everything you do with a Mac—from calculating a spreadsheet to using the Finder to find a file—turns into a series of binary numbers. The same is true of all software, which delivers its commands to the CPU in the form of binary numbers. However, programmers don't write software as a string of 0's and 1's. We refer to software as *code* because programmers use high-level programming languages that in turn use the letters of the alphabet and English words to represent the 0's and 1's.

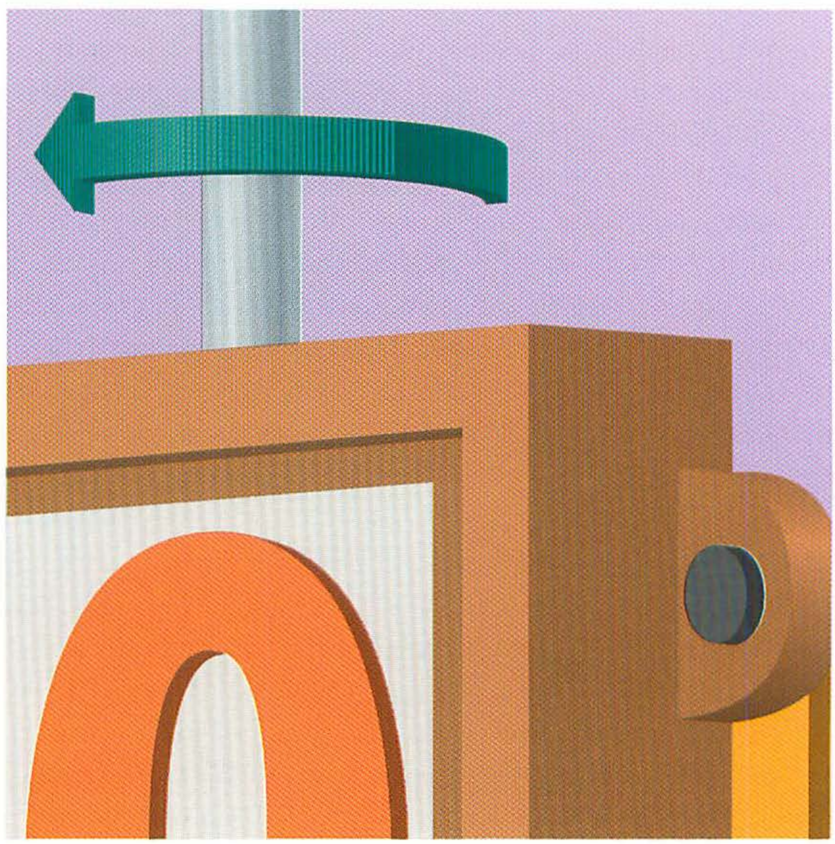
Of course, the CPU and main system memory aren't the only electronic components in a computer. Besides main system memory, personal computers use RAM chips for other purposes. For example, *video RAM* stores the information used to display images on your monitor. A *cache* is a special type of RAM that speeds computer performance by keeping the CPU from waiting for information. You'll find caches on the logic board, inside hard-disk and CD-ROM drives, and even inside the CPU itself. Mac OS can also use standard Mac hardware components to perform some special memory tricks. *Virtual memory* is the trick that uses part of a hard disk as an extension of the main system memory. The Mac can also do the opposite trick: create a RAM disk by using RAM chips as a virtual hard disk.

Mac OS determines how efficiently the Mac will use the CPU, main system memory, and the various other types of real and virtual memory in the Mac. Mac OS has had to keep up with the increasing speed of microprocessors and RAM chips through the years to make sure that the CPUs didn't sit idling while waiting for information. Mac OS 8 is a major step forward, and will radically overhaul the way the Mac uses system memory and virtual memory. However, the upgrading won't stop with Mac OS 8. Since the beginning of the computer age, hardware has traditionally been ahead of software's ability to take advantage of it. Hardware still has a long way to go before it hits a ceiling.

CHAPTER

10

Binary Numbers and Transistors



COMPUTERS use the binary number system to count. To understand why, let's look at some of the ways we can count. The most primitive method might be to assign a character—a vertical line (|), for instance—to each object we are counting. For example, if we had six sticks, we could represent them as

| | | | | |

This numbering system, however, becomes unmanageable when the numbers grow large. To prevent massing a large number of characters, we could assign a unique character to every number, but we'd soon run out of characters and would have an awful lot to memorize as well.

The decimal system is a compromise: It uses ten characters for the first ten numbers (0 through 9) and combinations of these characters to represent larger numbers. We have a 1's place, a 10's place, a 100's place, and so on. Each place represents ten times the place to its right; for example, the number 126 tells us we have six 1's, two 10's, and one 100.

In the binary system, we only have two characters—0 and 1—so each place represents twice as much as the place to its right. This gives us a 1's place, a 2's place, a 4's place, an 8's place, and so on. The binary number 1011, then, tells us we have one 1, one 2, no 4's, and one 8. In the decimal system, that's 1 + 2 + 0 + 8, or 11.

In computer terminology, each binary place is called a *bit*, so 1101 is a 4-bit number. Bits are arranged in groups of 8, called *bytes*. The biggest 8-bit binary number, 11111111, is 255 in decimal (1 + 2 + 4 + 8 + 16 + 32 + 64 + 128), which isn't very big. However, today's Macs can handle numbers that are 4 bytes long, or 32 bits. The biggest 32-bit binary number (32 1's) is the decimal equivalent of 4,294,967,295—enough to add just about any two numbers in one step.

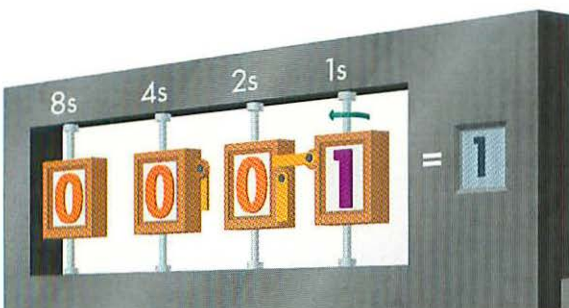
The reason binary notation is used in computers is that the electronics need only two types of electrical signals in different combinations to represent any number. The transistor is an ideal device to represent a single binary place, or bit. When a polarity is applied, we get a current flow, representing a 1. Reverse the polarity and the current stops—we have a 0.

Transistors are made from the element that gives Silicon Valley its name. Silicon, found naturally as silicon dioxide (aka silica) in quartz, agate, and sand, is an insulator—that is, it won't conduct an electric current. A process called *doping* turns silicon into a semiconductor—a material that is mildly conductive—by adding impurities. Silicon doped with phosphorus is called n-type and has a net negative charge. Silicon doped with boron, p-type, has a net positive charge.

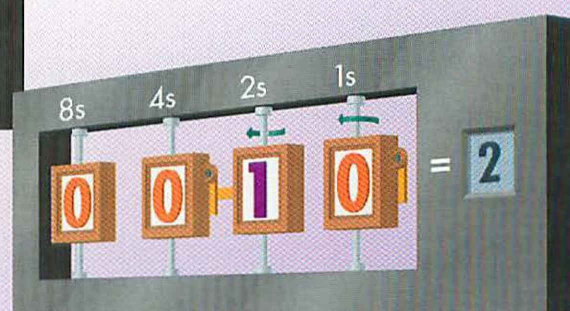
A transistor consists of three layers of p- and n-type silicon. A metal lead is connected to each of the three layers, now called the base, emitter, and collector. In this chapter we'll look at an npn transistor, the type used in most integrated circuits.

Binary Numbers and Transistors

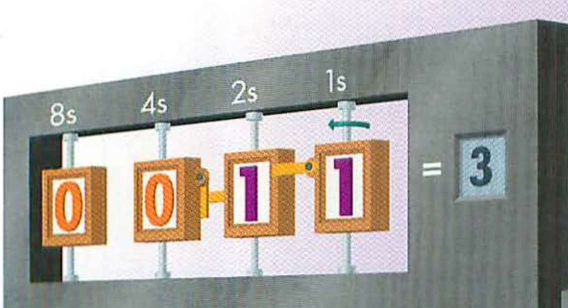
To demonstrate how binary addition works, we constructed an imaginary binary adding machine. Each place, or bit, consists of a spinning sign with a 0 painted on one side and a 1 painted on the other. Attached to the side of the signs are levers. When the 0 side faces forward, the lever is in the down position. When the 1 side faces forward, the lever sticks straight out to the left, hitting the next sign. The decimal readout on the right displays the decimal equivalent of the binary number.



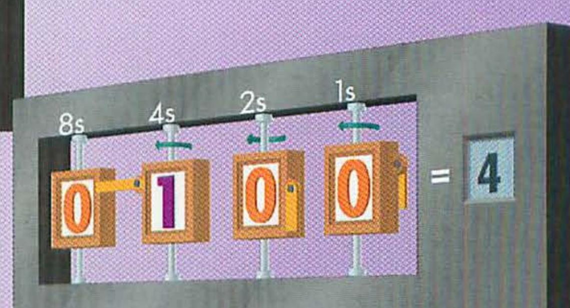
1 We start with 0000 and rotate the sign in the 1's place from 0 to 1. The lever hits the sign on the next column and stops.



2 To increase the number by 1, we spin the 1's place sign again, returning it to 0. This causes the lever to flip the sign in the 2's place from 0 to 1.



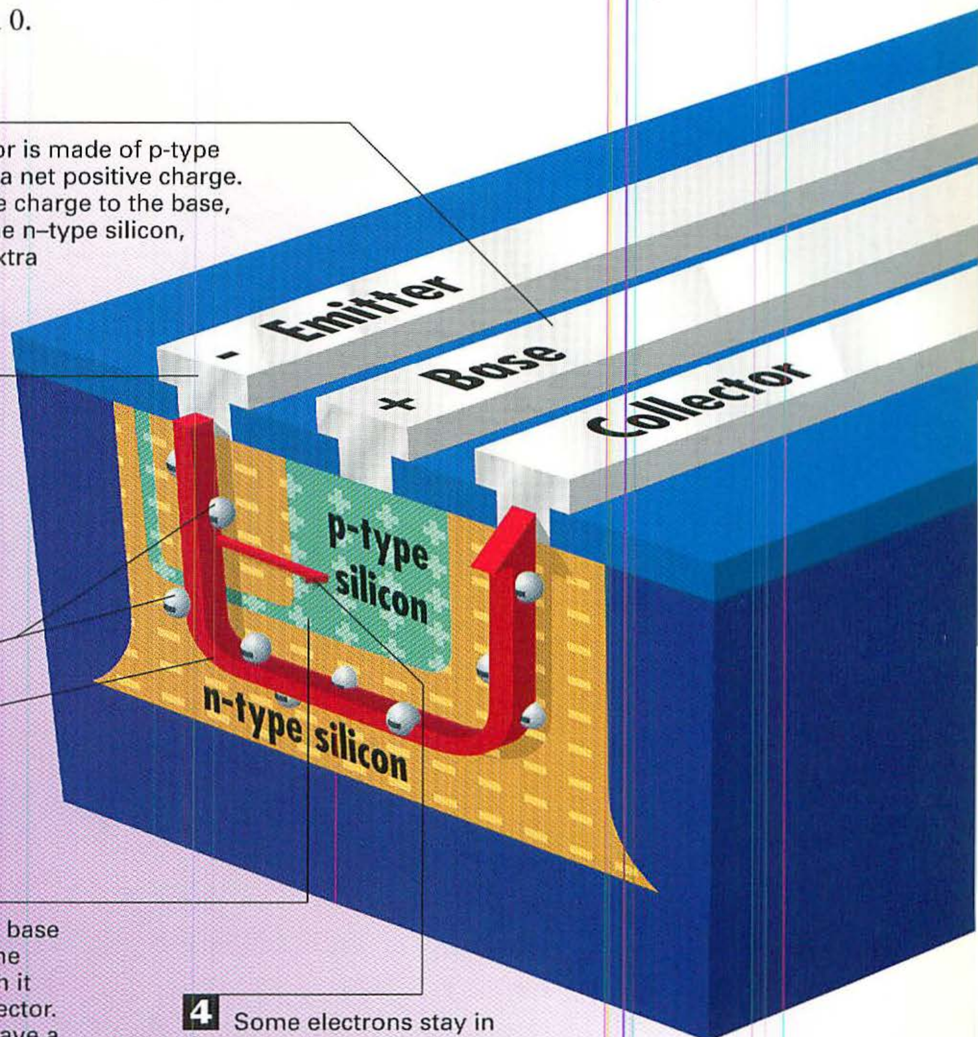
3 Increasing the number by 1 again causes the 1's place to advance from 0 to 1, but doesn't affect the 2's place.



4 Adding another 1 to the 1's place causes the signs in both the 2's and 4's places to turn, giving us the binary number 0100, or decimal 4.

A Transistor

A transistor makes use of the fundamental physics principles that opposite charges attract and that electrons, which are negatively charged, will move from an area of more electrons to an area of fewer electrons. A flow of electrons is a *current*. Applying a small positive charge to the base causes a current to flow through a transistor, putting it in active mode, which can represent the binary numeral 1. By reversing the charge at the base connector to negative, electrons from the emitter are repulsed and no current flows. This case represents a 0.



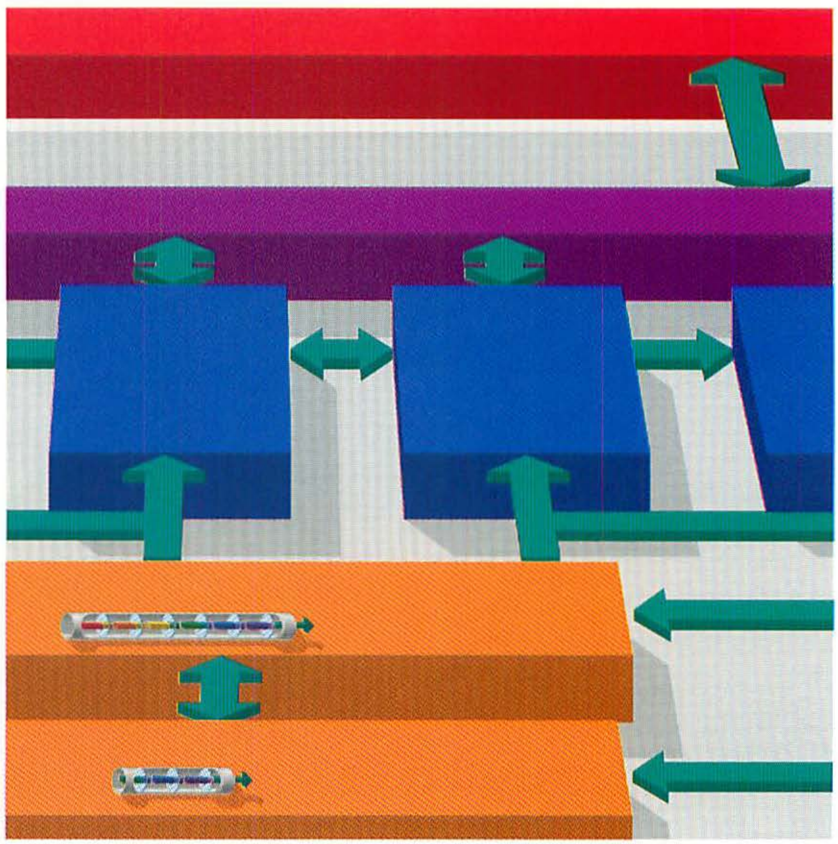
1 The base of an npn transistor is made of p-type silicon, which normally has a net positive charge. If we apply a further positive charge to the base, we'll draw electrons from the n–type silicon, which normally has a few extra electrons.

2 If we apply a negative charge to the emitter, it becomes even more negative than normal, tending to push electrons into the p-type base region.

3 Because the p-type silicon of the base is such a small region, most of the electrons are swept right through it into the n-type silicon of the collector. Because the collector does not have a negative charge applied to it, it is less negative than the emitter, which permits electrons to flow into it.

4 Some electrons stay in the p region. But because the p region is very thin (from .1 to 10 microns), the resulting current, called the *trickle current*, is much smaller than the current resulting from the electron flow from emitter to collector. In effect, the small trickle current controls the much larger current passing through the base from emitter to collector.

CHAPTER 11 How the CPU Works



CONNECT a few transistors together and you have an adding machine. Group a few adding machines together and you have circuits that perform more complex functions. Combine these functions into operational systems and eventually you'd wind up with millions of transistors switching on and off tens of millions of times per second—a microprocessor, a computer's central processing unit (CPU).

The CPU puts the computing in your computer. The rest of the Mac's components serve only to get information to the CPU, and the result of its efforts back to you.

The CPU functions reside in a microprocessor, a wafer of silicon measuring approximately one half inch on each side. This wafer sits inside a protective ceramic case. The microprocessor's thousands of circuits are microscopic in size, sometimes measuring only 0.35 microns (millionths of a meter) apart. While a computer may have more than one microprocessor, such as a coprocessor on an expansion card to help speed up the Mac, there is only one CPU.

Apple has used two types of CPUs in Macs: the older Motorola 680x0 line of microprocessors, and the next-generation PowerPC microprocessors, created by IBM, Motorola, and Apple. PowerPC processors are several times faster than the old 680x0 CPUs, and will continue to get faster for years to come. Most Macs sold today use PowerPC CPUs, including all Mac clones. Eventually, Apple will stop using 680x0 chips in Macs.

The CPU is the brains of a Mac, but it is not a free thinker; it does what it is told to do by software. The operating system and applications send the CPU two types of information: instructions and data. *Instructions* are commands from the software, and the *data* are the numbers used in the calculations. An instruction is a simple request, and a typical task usually consists of many instructions. One of the reasons the PowerPC line is so much faster than the 680x0 line is the way it handles instructions. PowerPCs use an advanced method called *RISC*, short for reduced instruction set computing. The 680x0 chips use a method called *CISC* (complex instruction set computing). RISC-based CPUs are faster than CISC microprocessors because they use small instructions that are all the same size. CISC CPUs, including the 680x0, are less efficient. They use more complex commands of differing sizes, and use a greater number of different instructions.

Microprocessors receive and send information through dozens of pins (the 68040 has 145; the PowerPC 604 has 304). These signals are synchronized by a clock, which provides timing signals, called cycles, at a constant rate. The speed of the CPU clock is measured in megahertz (MHz), millions of cycles per second. A common misconception is that the higher the MHz rating of a CPU, the faster the computer. While this is true for computers with the same CPU, comparing clock speed ratings of different CPUs to judge performance is like comparing the speed of cars by looking only at their engines' RPM ratings. For instance, a 100 MHz PowerPC 604 is about 1.5 times faster than a 100 MHz PowerPC 601. Comparing clock rates for RISC and CISC makes even less sense, as RISC often requires more clock cycles to finish certain tasks. However, because RISC computing is simpler and more efficient than CISC, manufactures can build RISC chips with much higher clock rates than CISC.

The other factors that influence a computer's speed include the amount of cache in the microprocessor, whether the CPU processes data in 32-bit or 16-bit chunks, and whether the computer communicates with the CPU in 16-bit chunks, 32-bit chunks, or with even wider data paths. The more bits of information that can be passed around, the faster the performance.

In addition, some CPUs get more done per clock cycle. For instance, the 68000 requires 4 cycles to process an instruction, whereas the 68040 only requires 1.3 cycles on average. The 68040 achieves this rate in part because its design includes instances of *parallel processing*, which allows certain tasks to be performed simultaneously. The PowerPC chips have an even greater capacity for parallel processing, and are able to execute more than one instruction per clock cycle. The PowerPC 601 can execute three instructions per clock cycle, while the 604 can process four instructions per clock cycle. The use of simple instructions in RISC chips makes it possible to use a *superscalar* design, which duplicates certain components inside the microprocessor called *execution units*, which are the calulating engines of the CPU. The 601 has four execution units, and the 604 has six.

The most important of these execution units is the *integer unit*, also called the fixed point unit. Integer math is the most common type of CPU calculation used by software. The 604 processor has three integer units.

Another type of execution unit does *floating-point* calculations, a type of math popular with graphical and mathematical software. RISC can execute floating point instructions faster than CISC, which helps make the Power Mac one of the fastest graphics machines around.

The advanced PowerPC design requires that it run *PowerPC-native* software that is specially written to take advantage of its RISC features. You wouldn't know this by using a Power Mac, which seems to run just about any Mac software, PowerPC native or not. This is because Power Macs use a piece of PowerPC-native code called a 68040 emulator. The emulator translates instructions from software written for the 680x0 Macs into PowerPC calls. The old software thinks it is running on a 680x0 Mac, while the PowerPC CPU thinks it is running PowerPC-native software. (This is the same principle behind Insignia Solution's SoftWindows, which lets you run DOS and Windows software on your Mac. SoftWindows emulates an Intel 486 CPU on a Mac.)

Running software in emulation is not as fast as running PowerPC-native software. In lower-end Power Macs, emulation mode is slower than running the software on a 68040 Mac. Since much of System 7.5 itself runs in emulation mode, Power Macs will get a big boost in speed when running a completely PowerPC-native operating system, such as Mac OS 8.

There is another limitation to the Power Mac's emulation software. The specific CPU it emulates is a 680LC40, the low-power version of the 68040 used in PowerBook laptop Macs. The problem is that the 68LC040 does not have a floating-point unit, so Power Macs (as well as 040 PowerBooks) cannot run 68040 software that uses floating-point math. Fortunately, most of these software packages are now available in PowerPC-native form.

PowerBooks with PowerPC CPUs can run the PowerPC native floating-point software because they use the PowerPC 603e. Unlike the 68LC040, the 603e does have a floating-point unit. The 603e, which is similar to the 601, gets its power savings (3.5 watts maximum compared to 6 watts in the 601) by using smaller caches. Several desktop Macs, including the Performa 5300 and 6200 series, also use the 603e, which runs a little slower than the PowerPC 601.

As important as floating-point math is in graphics today, the first Mac CPU, the 68000, had no floating-point capability. The Mac II in 1987 used a 68020, which worked alongside a separate floating-point coprocessor, the 68881. The 68030 CPU introduced in the SE/30 worked with an improved 68882 floating-point math coprocessor. The 68030 was the first Mac CPU to include a built-in memory management unit, which is required for the virtual memory feature of Mac OS. The first Mac CPU to include built-in floating point was the 68040, the last of the 680x0s used in Macs.

The 68040 was about as fast as Motorola could make the 680x0 line, but the future of the PowerPC line is almost open-ended. While the first PowerPC chips used in 1994 Power Macs ran at 60 MHz, today's versions are running at more than twice that clock rate. IBM and Motorola are working on a new PowerPC microprocessor that will soon exceed 200 MHz and run many times faster than today's high-end 604 CPUs.

While increasingly faster microprocessors are not an unexpected development, there is one interesting twist to the PowerPC line. IBM is working on the PowerPC 615, which will have hardware elements of both the PowerPC and the Intel Pentium lines of microprocessors. This is not software emulation, like SoftWindows or the 68LC040 emulation in Power Macs, but a new type of dual-personality CPU.

The 615 will be pin compatible with the Intel Pentium, which means you'll be able to use it in an unmodified PC. Apple plans to use it at some point in its version of the PowerPC Platform, a machine designed to run both Mac and Windows software. The 615 can start up as if it had either a PowerPC or a Pentium, and run as fast as either of those microprocessors. It can also boot up in a mixed mode, letting you run both types of software, but giving slower performance in return.

To date, the fastest PowerPC processors have been faster than the quickest Pentiums. The PowerPC's integer speed has been just slightly faster than that of the Pentium, but floating-point performance has been far ahead of Pentium. Motorola expects this trend to continue, keeping the Mac CPU in control of our favorite software.

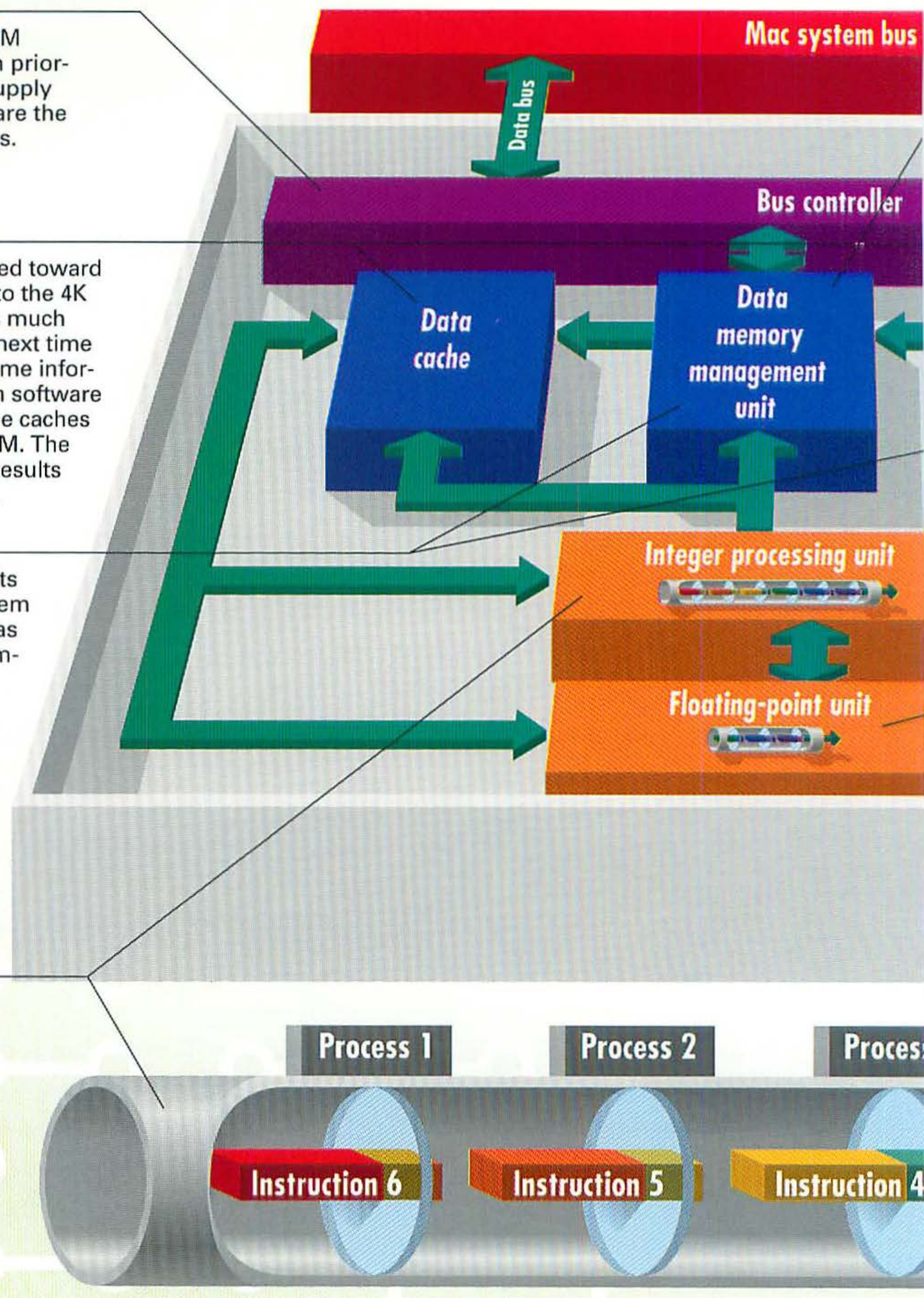
The 68040 Microprocessor

1 Instructions and data from RAM enter the bus controller, which prioritizes them to keep a steady supply to the execution units, which are the integer and floating-point units.

3 New data and instructions routed toward the execution units are copied to the 4K caches, which have 16 times as much cache space as the 68030. The next time the execution units need the same information (which happens often in software execution), they fetch it from the caches instead of going to external RAM. The caches also hold intermediate results that require further processing.

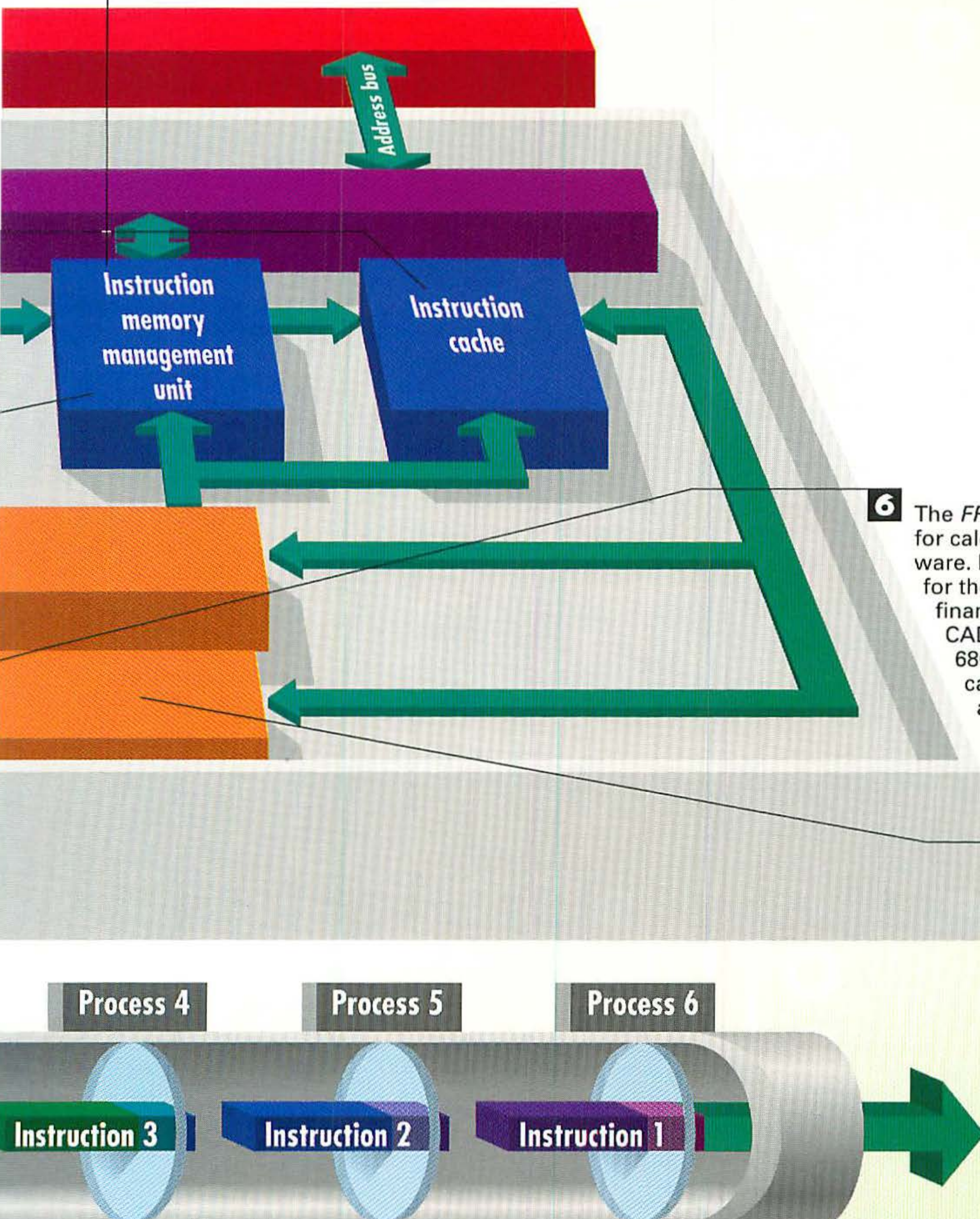
4 The memory management units control virtual memory, a System 7 feature that uses disk space as an extension to RAM. The memory management units keep tables of byte addresses to keep track of where the information is.

5 The *integer processing unit* processes most of the information during the operation of the Mac. The 68040's integer unit uses a parallel-processing technique called *pipelining* to process up to six instructions at a time. Instructions are sent through an assembly line consisting of six stages of execution—a different instruction can be run at each stage of the pipeline. (The 68030 has a three-stage pipeline.)



2 Data and instructions are separated and flow along two separate bus lines, enabling them to be processed simultaneously.

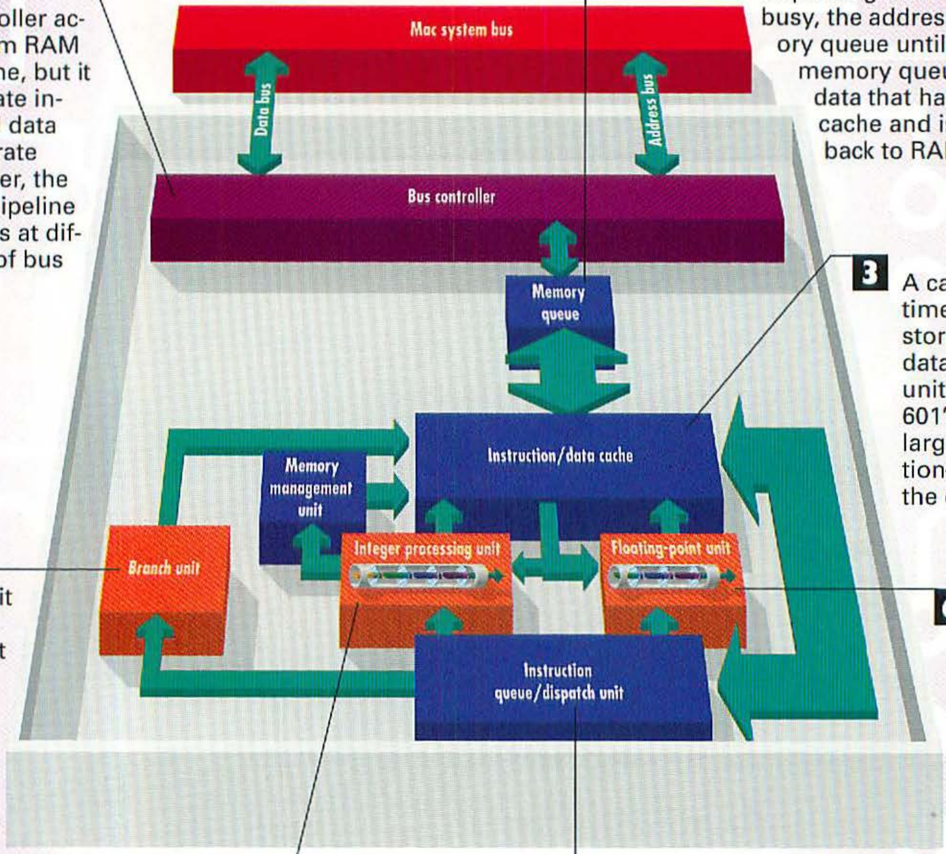
MAC FACT The Motorola MC68000, the first CPU used in Macs, was named after the 68,000 transistors it contained. By 1987, this amount more than quadrupled to 325,000 transistors in the MC68030. By 1990, Motorola was packing 1.2 million transistors into less than 1 square inch in the MC68040, which first appeared in Mac Quadras and Centrises. 1993's PowerPC 601 more than doubled this to 2.8 million transistors in a 1.7-square-inch area. The PowerPC 604 contains 3.6 million transistors, as many as in 53 68000 microprocessors.



6 The *FPU* (floating-point unit) can be used for calculations if specified by the software. Floating-point processing is faster for the types of calculations used in financial analysis and in drawing and CAD applications. The FPU in the 68040 has a three-stage pipeline, and can process three math instructions at the same time. Completed calculations are sent back to the bus unit for delivery to RAM.

7 The 68LC040, the low-power version used in the PowerBook 500 series and in some desktop Macs, does not have a floating-point unit.

The PowerPC 601 RISC Microprocessor



1 The bus controller accepts data from RAM 64 bits at a time, but it doesn't separate instructions and data into two separate buses. However, the bus unit can pipeline two operations at different stages of bus processing.

2 The memory queue holds the addresses of instructions that are not yet in cache but are requested by the execution units. When data isn't in the cache, requests go out to the bus. If the bus is busy, the addresses wait in the memory queue until the bus is free. The memory queue also stores new data that has been written into cache and is waiting to be written back to RAM.

3 A cache of 32K (four times that of the 68040) stores instructions and data until the execution units can use them. The 601's cache can supply a large flow of information—256 bits worth—to the execution units.

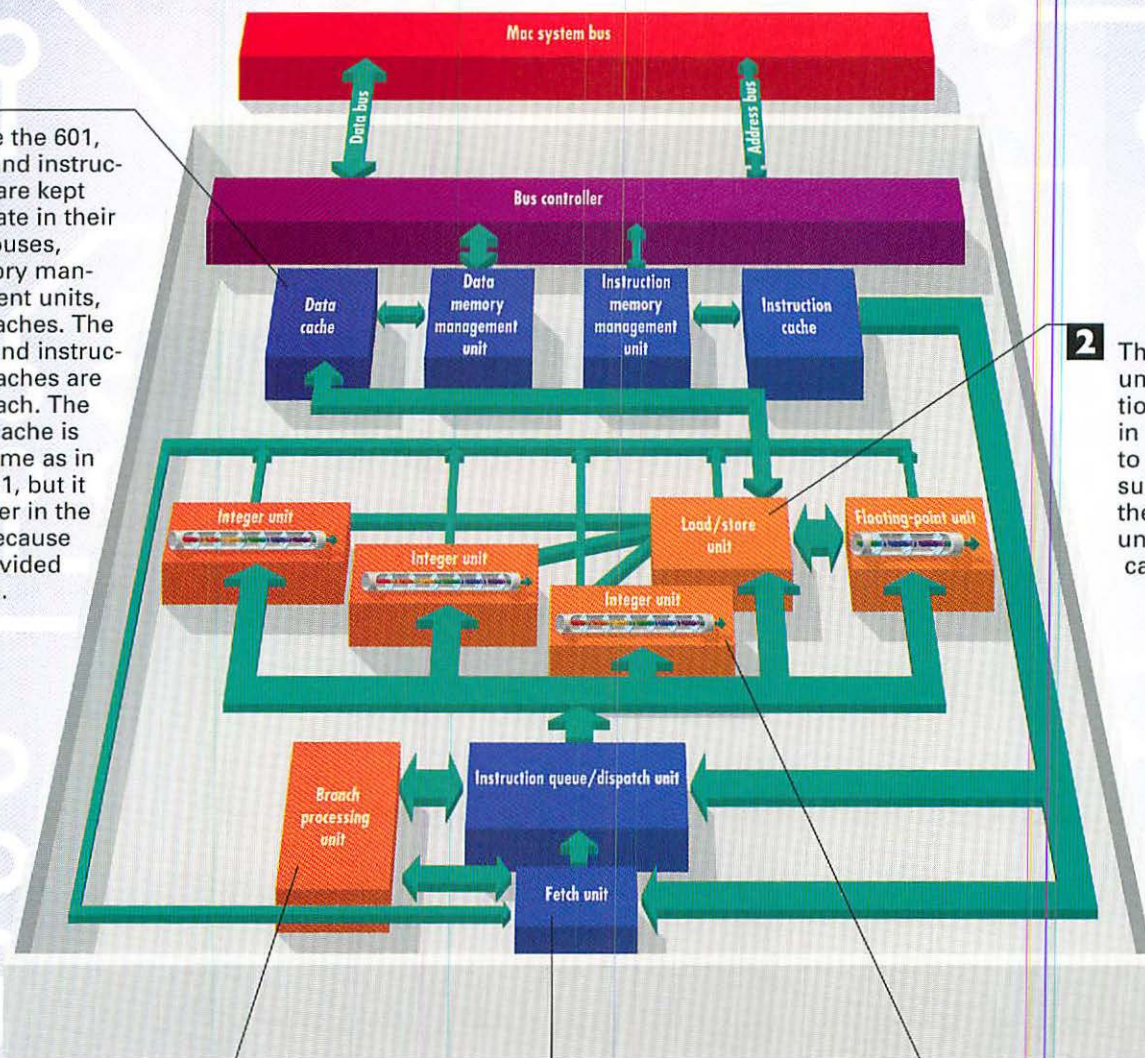
4 The instruction dispatch unit is filled with information from the cache in eight-instruction chunks. The instruction dispatch unit simultaneously feeds instructions to the three execution units as fast as they can take them. The three execution units run independently of each other and can process more than one instruction at a time, so the timing of dispatches can be quite complex.

5 The integer unit of the 601 uses a four-stage pipeline to process four instructions at the same time.

6 The floating-point unit does calculations used in graphics-heavy applications. The unit has a 64-bit data path to cache.

7 The branch unit is a third execution unit that is used when the computer must skip over to another part of a software program, changing the work flow. The branch unit keeps the branches in the software from interrupting the flow of instructions to the execution units. It does this by looking ahead for branches and fetching them from the instruction cache.

The PowerPC 604



1 Unlike the 601, data and instructions are kept separate in their own buses, memory management units, and caches. The data and instruction caches are 16K each. The total cache is the same as in the 601, but it is faster in the 604 because it is divided in two.

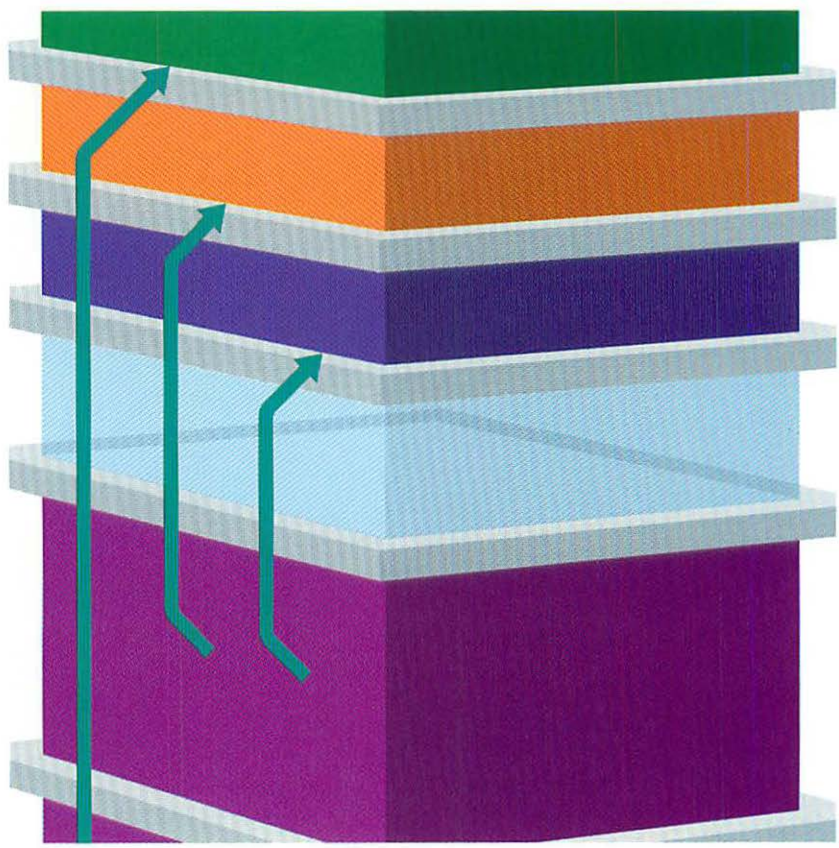
2 The load and store unit is a new execution unit, one of six in the 604. Its job is to transfer the results data between the other execution units and the data cache.

3 Parallel processing in integer math is increased over the 601 in two ways. First, unlike any Mac CPU before it, the 604 has three integer units running in parallel. Second, the pipeline within each integer unit has been increased to six stages.

4 The fetch unit supplies instructions to the eight-word instruction queue by accessing the on-chip instruction cache. The fetch unit can retrieve as many as four instructions at a time.

5 The branch processing unit performs the same function as it does in the 601, predicting branches of instructions in software and managing their flow.

CHAPTER 12 How RAM Works



THE main memory used in the Mac and most PCs is a type known as dynamic random-access memory (DRAM, or just RAM). It is called dynamic because the contents are constantly changing. *Random-access* means the CPU can directly access information stored anywhere inside the chips, just as you can access any part of a phonograph record by dropping the needle down wherever you like on the disk. (A cassette tape, by comparison, is not a random-access medium, because you have to wind through tape in order to get to a section in the middle.)

Most Macintosh memory comes in the form of small add-on cards called single inline memory modules, better known as *SIMMs*. Some of the faster Macs use dual inline memory modules, or *DIMMs*. DIMMs are faster than SIMMs because they can move more data at one time to the CPU. (PowerBooks use their own nonstandard memory cards.) Although SIMMs and DIMMs come in different storage sizes, most contain eight RAM chips, one chip for each bit in an 8-bit electronic word. (There's also a nine chip version used mostly by government agencies, called parity RAM, but it is not common.) Regardless of how many SIMMs or DIMMS your Mac has or how much RAM is soldered on the logic board, all the RAM in a Mac acts as a single pool of memory.

The CPU can access a piece of information stored in RAM by specifying an *address*, a number that identifies the location of each byte of information in RAM. Address 0 and the first several hundred address locations, referred to as *low memory*, are always taken by the system partition, which is created at startup. Applications are loaded into the highest available addresses, referred to as *high memory*.

The memory strategy used by Mac OS in versions 7.*x* and earlier is called *cooperative multitasking*, in which RAM is dynamically allocated according to the needs of the applications. This strategy enables multiple applications to share the available RAM by running in partitions. The operating system's Memory Manager determines which applications are allocated to which parts of RAM.

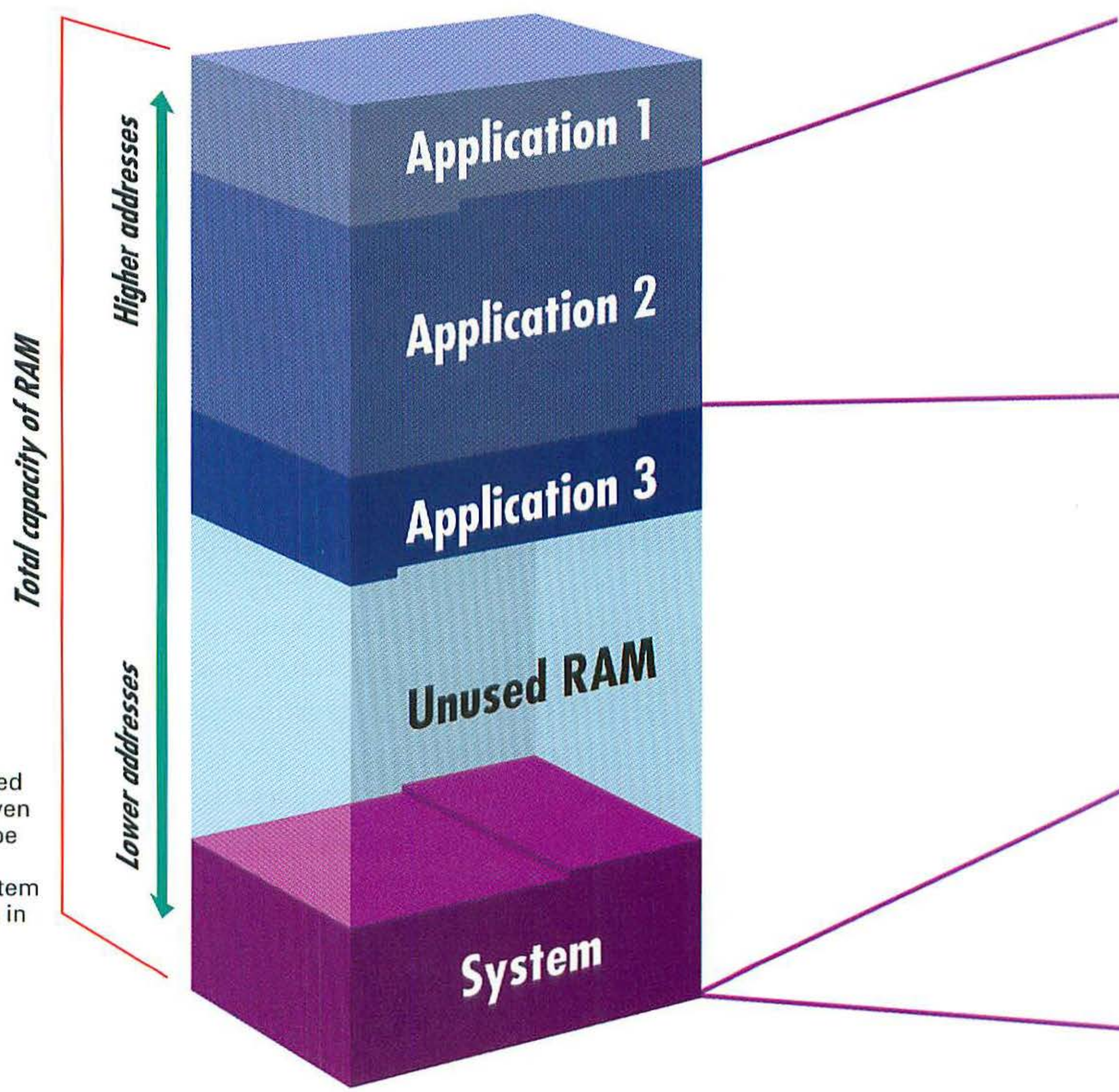
You can set the size of an application's memory partition by selecting the application's icon in the Finder and choosing Get Info from the Finder's File menu. The suggested size is set by the software manufacturer, but you may want to enter a bigger number if you are working with large files. Up to the point where the entire application is loaded into memory, the bigger the application partition, the fewer times the application will need to go to disk and the faster its performance.

This will all change with Mac OS 8, which will load only a small part of an application into RAM at one time, loading the rest as needed. You'll be able to open as many applications as you want. Adding RAM will increase performance.

Mac OS 8 will still use cooperative multitasking for running applications, but adds preemptive multitasking for low-level Mac OS functions and for application subtasks, such as rendering complex graphic images. Preemptive multitasking speeds performance and prevents software conflicts. Eventually, Mac OS will move to a completely preemptive multitasking system.

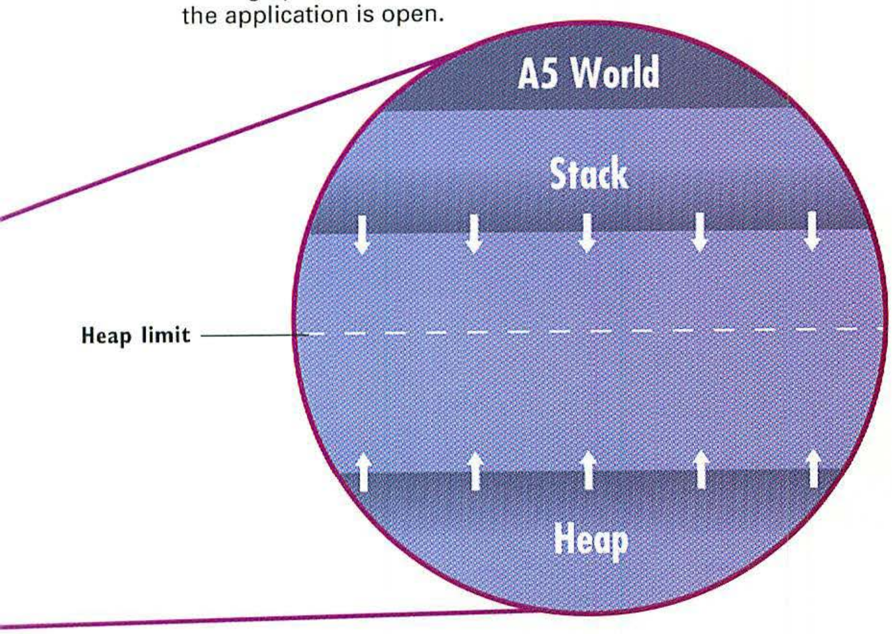
Dynamic Memory in System 7.*x*

2 When an application is launched, the operating system assigns it a partition in high memory. Usually, only a portion of the application and document are loaded into the partition. When additional applications are launched, new partitions are created starting at the highest address and working down. The total amount of memory taken up by an application partition is determined by the memory size setting in the application's Get Info box in the Finder.

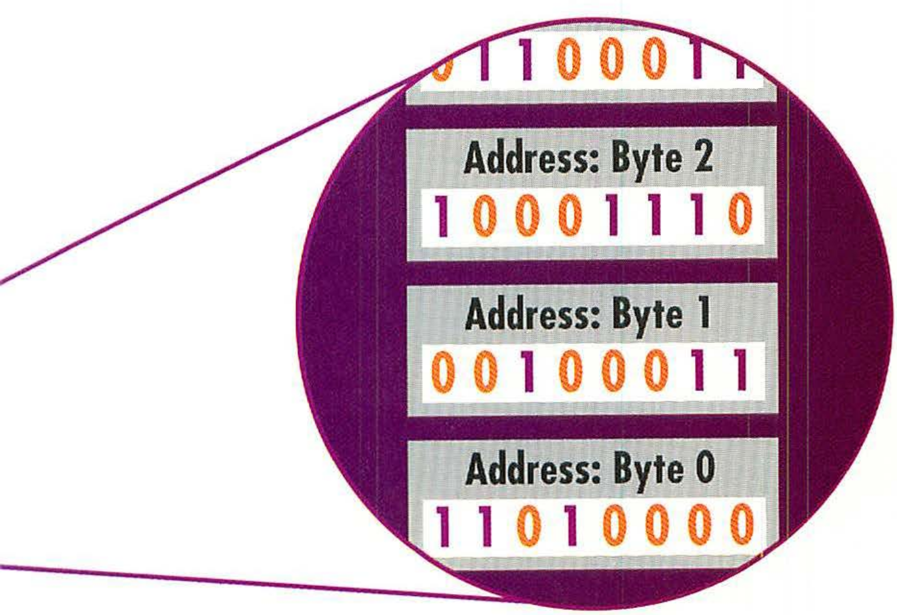


1 When information is loaded into RAM, each byte is given an address so that it can be located when needed. At startup, the operating system creates a system partition in low memory, starting at address 0.

3 Applications call their routines from the application *jump table*, which is a list of an application's routines and their memory addresses. The jump table is stored in the top part of an application partition, known as the A5 World, named after a part of the 68000 CPU called the A5 register. The A5 World also stores application global variables and parameters used by the Mac Toolbox and operating system. The A5 World is static, never changing size while the application is open.

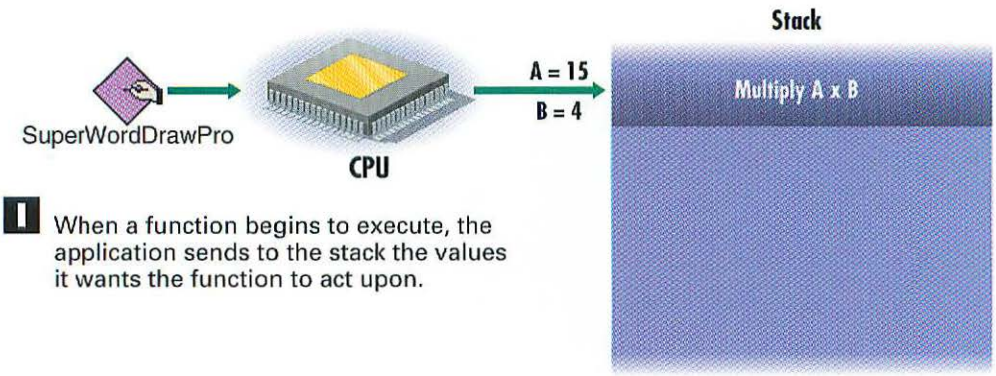


4 Most of the action occurs in the application *stack*, which holds temporary values, and in the application *heap*, which holds code segments, resources, and document data. The stack and the heap are dynamically allocated, constantly expanding into and retreating from an empty area of unallocated memory that sits between the two. The application stack can temporarily fill the entire unallocated space, but the heap can fill only a portion of it.

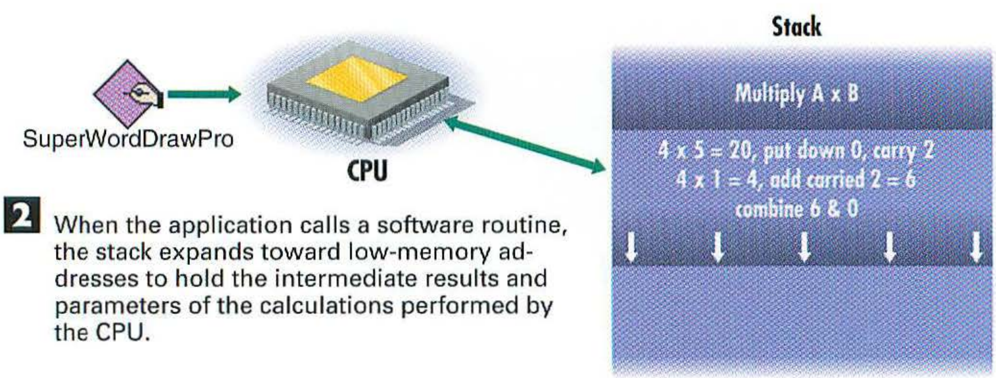


MAC FACT If RAM is in short supply when you open an application, you may get a somewhat paradoxical message telling you that although there isn't enough memory to open the application, you may want to click OK and open it anyway. In this case, the Memory Manager will load the application by shrinking the unallocated space between the stack and heap, as well as by creating a smaller heap. However, you may not be able to open large files or perform all the functions you normally could, and performance could suffer.

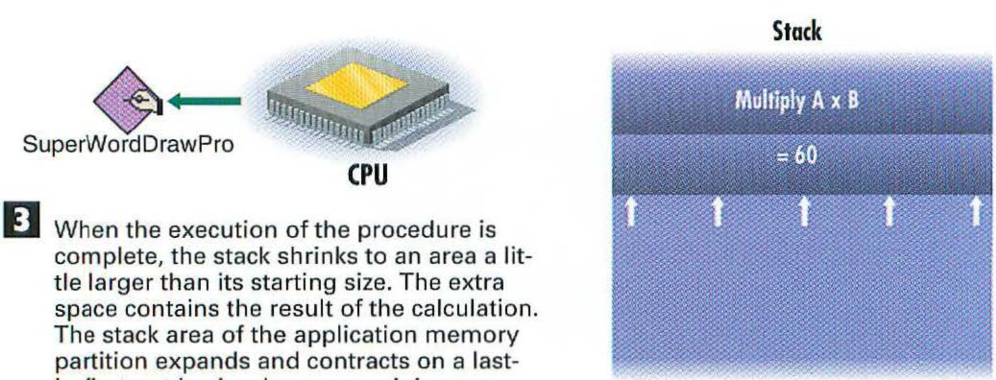
Application Stack



1 When a function begins to execute, the application sends to the stack the values it wants the function to act upon.



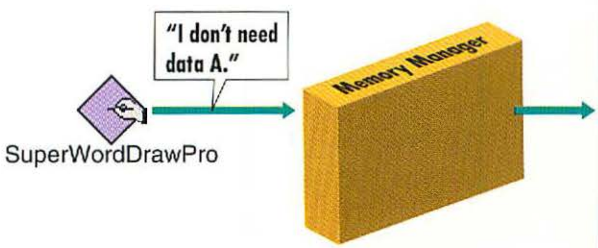
2 When the application calls a software routine, the stack expands toward low-memory addresses to hold the intermediate results and parameters of the calculations performed by the CPU.



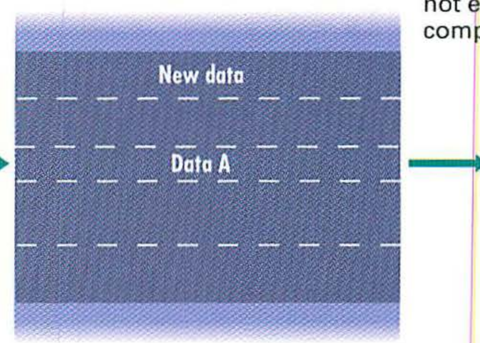
3 When the execution of the procedure is complete, the stack shrinks to an area a little larger than its starting size. The extra space contains the result of the calculation. The stack area of the application memory partition expands and contracts on a last-in-first-out basis, always remaining contiguous and never becoming fragmented.

Application Heap

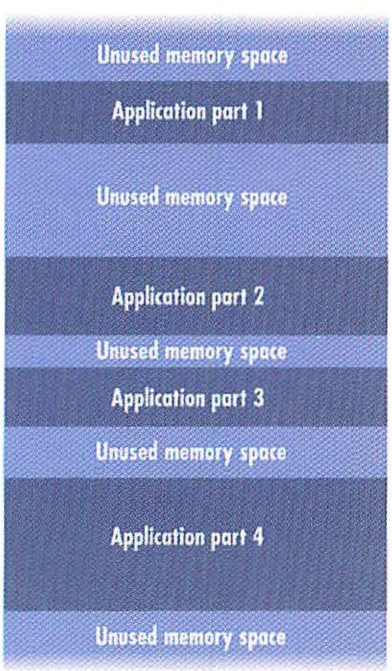
1 When an application no longer requires a certain piece of information, such as a routine or piece of data, the Memory Manager releases the space occupied by the information. This is often in the middle of heap.



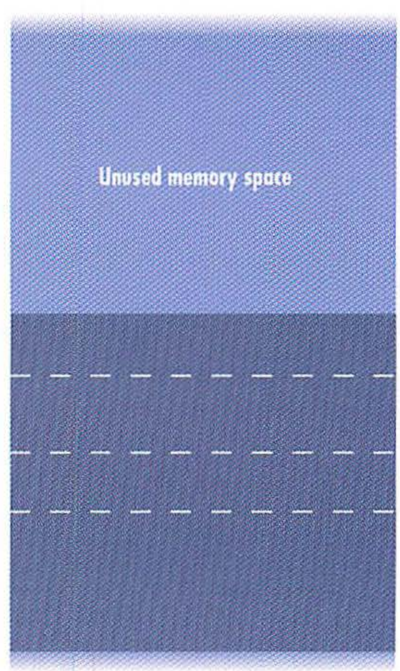
2 When an application needs a resource, code segment, or data from a file, the information is loaded at the top end of the heap. If the heap grows to the heap limit, the Mac displays a message telling you that there is not enough memory to complete the action.



3 When multiple areas of free unallocated RAM are formed in the middle of the heap, the memory is said to be *fragmented.* Fragmentation of memory wastes memory space and can prevent an application from performing certain tasks. (A similar condition on hard disks is called disk fragmentation.)



4 When fragmentation reaches a certain level, the Memory Manager *compacts* the heap by moving blocks of information into the empty spaces.

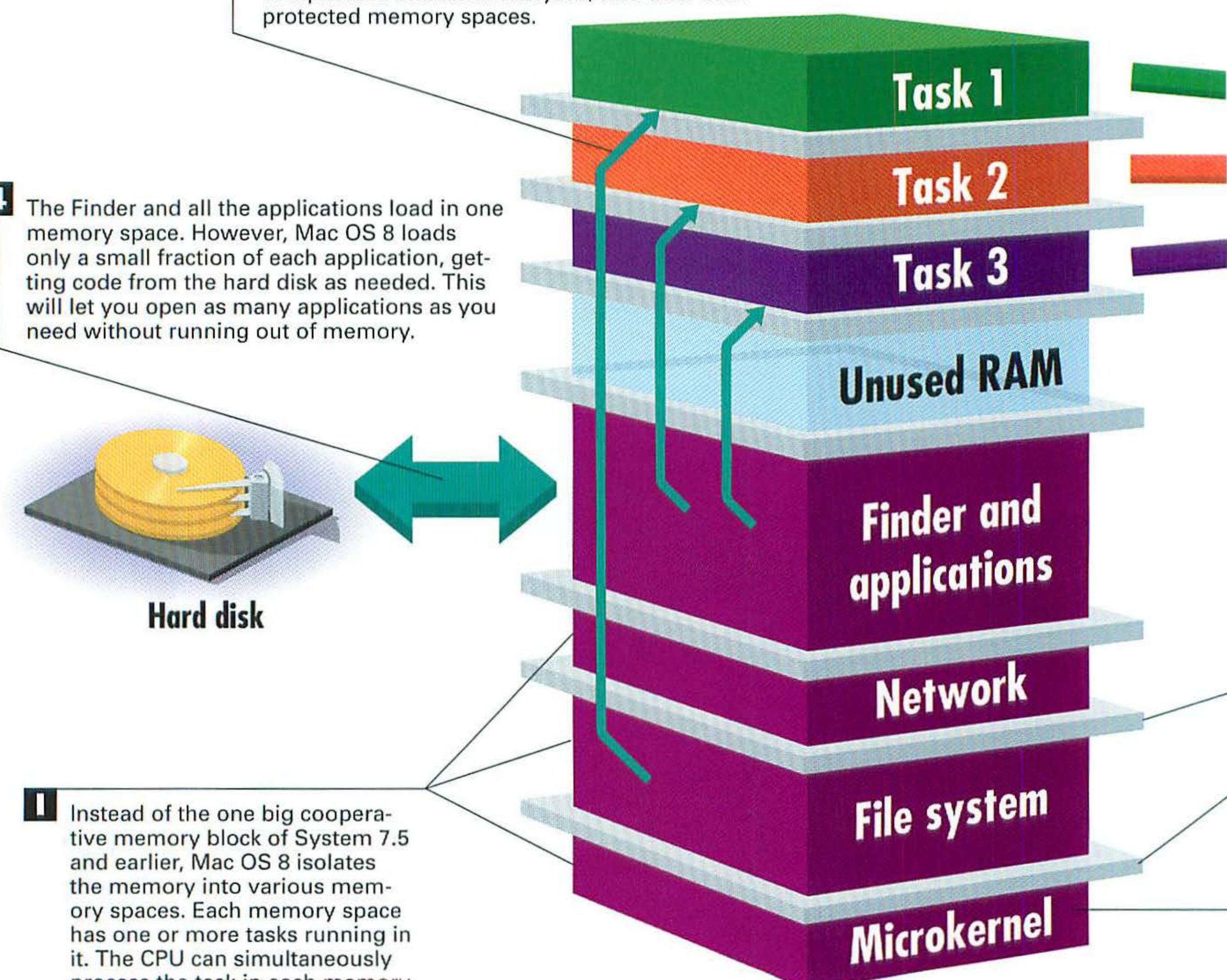


MAC FACT Most crashes on the Macintosh originate in RAM. Crashes will eliminate information in RAM, but don't harm the Mac hardware or data on the hard disk. If the stack grows into the space used by the heap, the heap can become corrupted, crashing the application. Applications can also crash if you've set the partition size too low. System crashes can occur when two programs think they each control the same segment of memory.

Preemptive, Protected Memory in Mac OS 8

5 Applications and operating system processes can "spin off" tasks (such as graphics renderings or complicated statistical analyses) into their own protected memory spaces.

4 The Finder and all the applications load in one memory space. However, Mac OS 8 loads only a small fraction of each application, getting code from the hard disk as needed. This will let you open as many applications as you need without running out of memory.



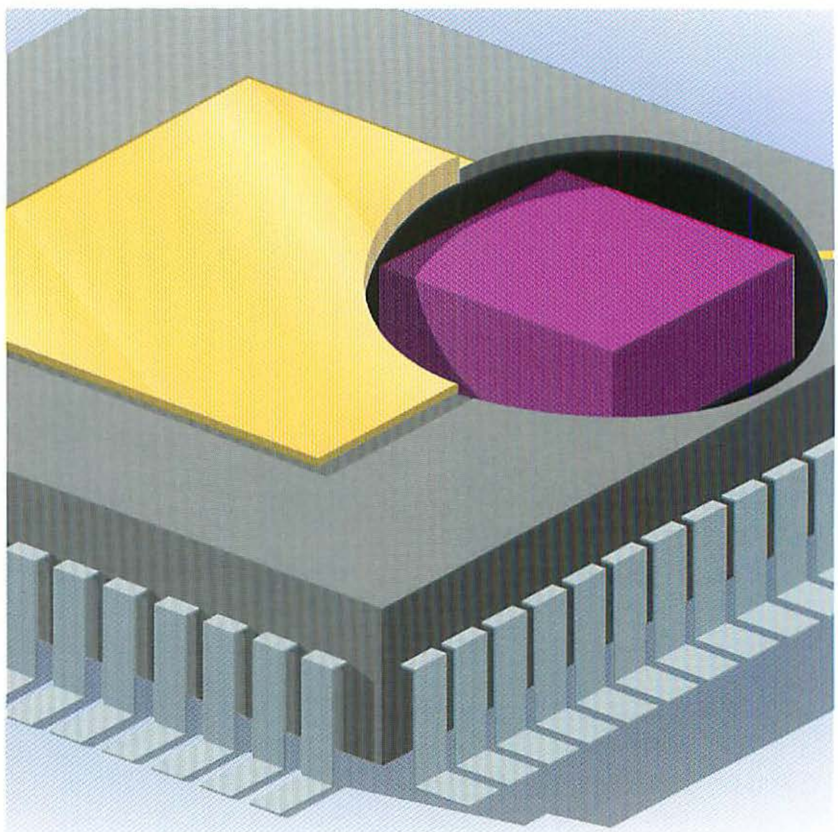
1 Instead of the one big cooperative memory block of System 7.5 and earlier, Mac OS 8 isolates the memory into various memory spaces. Each memory space has one or more tasks running in it. The CPU can simultaneously process the task in each memory space using a process called multitasking.

6 The Thread Manager keeps a very tight schedule for the CPU, filling every tiny slice of empty time with instructions and data. This enables multitasking of applications and processes. In System 7.5 and earlier versions of Mac OS, there are small but numerous periods of time when the CPU waits for information.

2 Unlike previous versions of Mac OS, there is no negotiation for memory addresses. Mac OS 8 protects each memory space from the others. If something inside one memory space crashes, it won't affect the processes running in the other memory spaces, and it won't crash the whole Mac. Instead of restarting the computer, you'll just have to restart the task in that memory space.

3 The Microkernel, file system, and other important operating system functions are isolated in their own protected memory spaces.

CHAPTER 13 How Virtual Memory Works



EVER need to open just one more application, but find you're 100K short of RAM? This is when *virtual memory* comes in handy. A feature of System 7 and Mac OS 8, virtual memory lets you open more files than you actually have room for in RAM. Virtual memory uses space on a hard disk as an annex to the main memory to store information that hasn't been used recently by the CPU. The CPU treats virtual-memory disk space as RAM.

The amount of virtual memory and the disk it resides on is set in the Memory control panel. Once virtual memory is set up and turned on, the Mac acts as if you had added more RAM. In the About This Macintosh window in the Finder, the number displayed for total memory will include the amount of disk space you've reserved for virtual memory.

The drawback to virtual memory is that it requires access to a hard disk, which is hundreds of times slower than RAM. Therefore, you should use virtual memory according to some guidelines. First, set the amount of virtual memory for what you need, and not any more. Virtual memory will run more smoothly with several applications using small-to-moderate-sized documents, rather than when a single RAM-hungry application with a large file is loaded. Optimally, the currently active application should fit in real RAM, and open applications not being used should be in virtual memory. You should also avoid running background tasks in other applications, which may cause frequent swapping between RAM and the virtual memory portion of the hard disk and can lead to very slow performance.

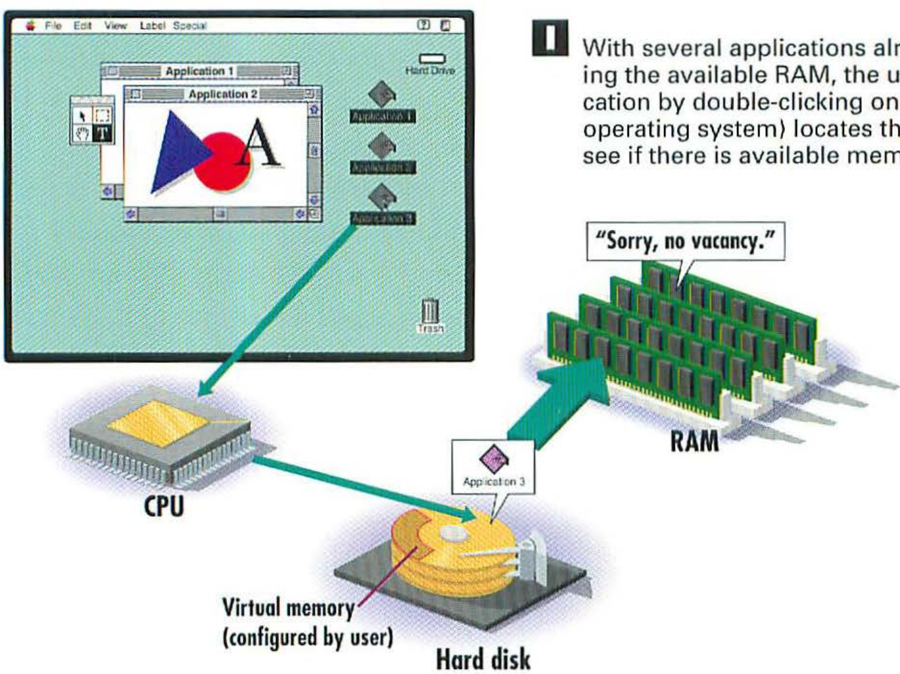
Older 68000-based Macs, such at the SE, the Classic, and the PowerBook 100, cannot use virtual memory. Virtual memory is available in Power Macs and in Macs with 68030 or 68040 CPUs. It's also available in 68020-based Macs (the old Mac) equipped with a Motorola 68851 memory management unit (MMU, also called a paged MMU, or PMMU). 68030s, 68040s, and all PowerPC microprocessors have built-in MMU functions.

Virtual memory in Power Macs is quite a bit faster than in 68040-based Macs. It also enables the System file and applications to open with less memory. For instance, ClarisWorks requires 3MB with virtual memory turned off, but only takes 1.4MB with virtual memory turned on. The Get Info for any Power Mac native application will tell you how much memory you can save with virtual memory.

Mac OS 8 will improve virtual memory even further; it will be faster and more efficient, and will let you have more applications running at the same time. You won't be able to turn off virtual memory in Mac OS 8 as it will be an integral part of the way the Mac handles memory.

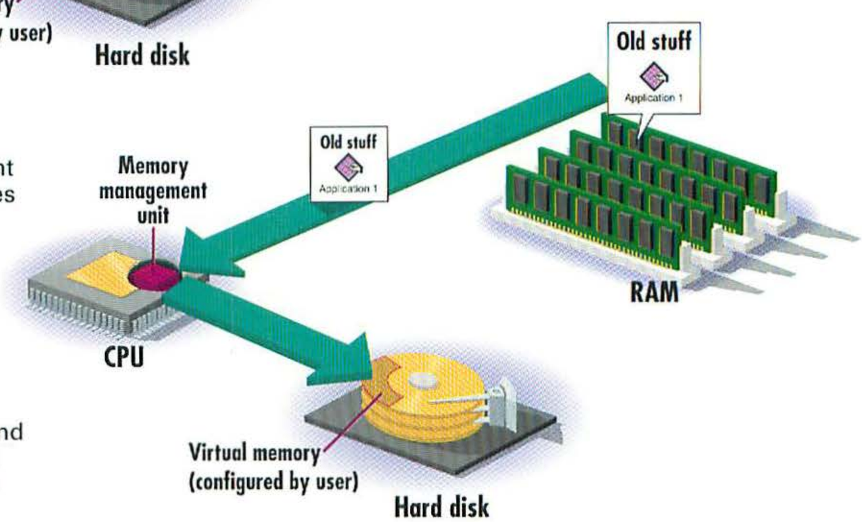
An Application Using Virtual Memory

When a user turns on virtual memory in the Memory control panel and restarts the Mac, the Virtual Memory Manager reserves a part of the hard disk as a virtual memory space. This space is not available to the user for file storage, and information in this space cannot be accessed by the user.

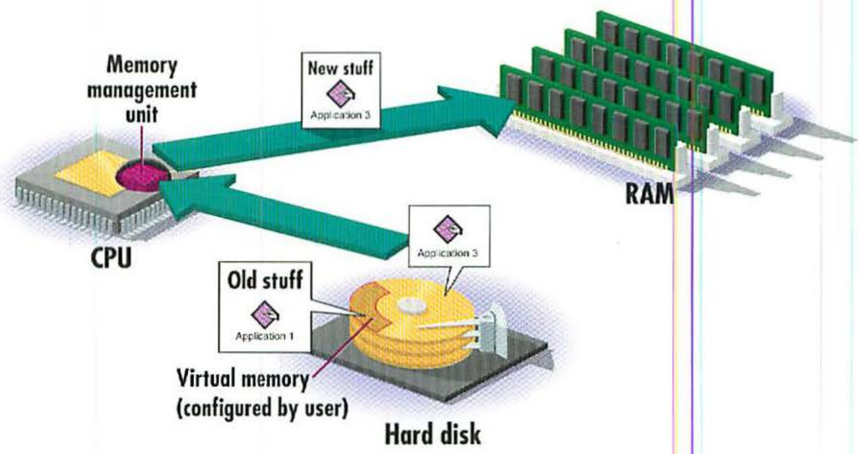


1 With several applications already open and completely filling the available RAM, the user tries to open another application by double-clicking on it. The CPU (directed by the operating system) locates the file on disk. When it checks to see if there is available memory, it finds that there is none.

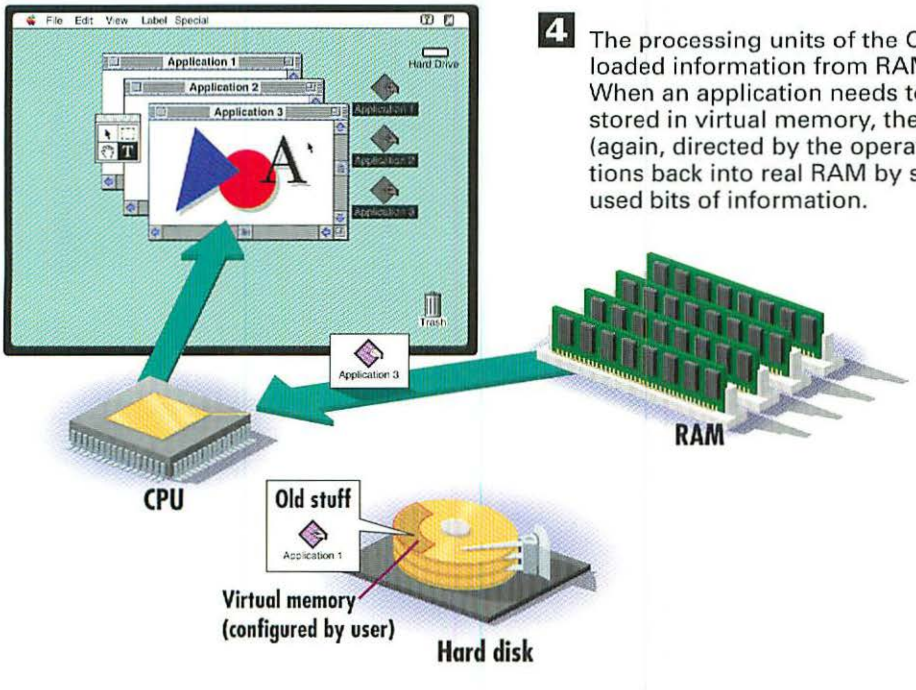
2 The memory management unit inside the CPU locates the information in RAM that has been used least recently and copies it to the hard-disk area designated as virtual memory. This frees up real RAM. The swapping of memory between RAM and virtual memory accounts for the slow performance of virtual memory.



3 The memory management unit then moves the new application from the normal data storage part of the hard disk into real RAM.

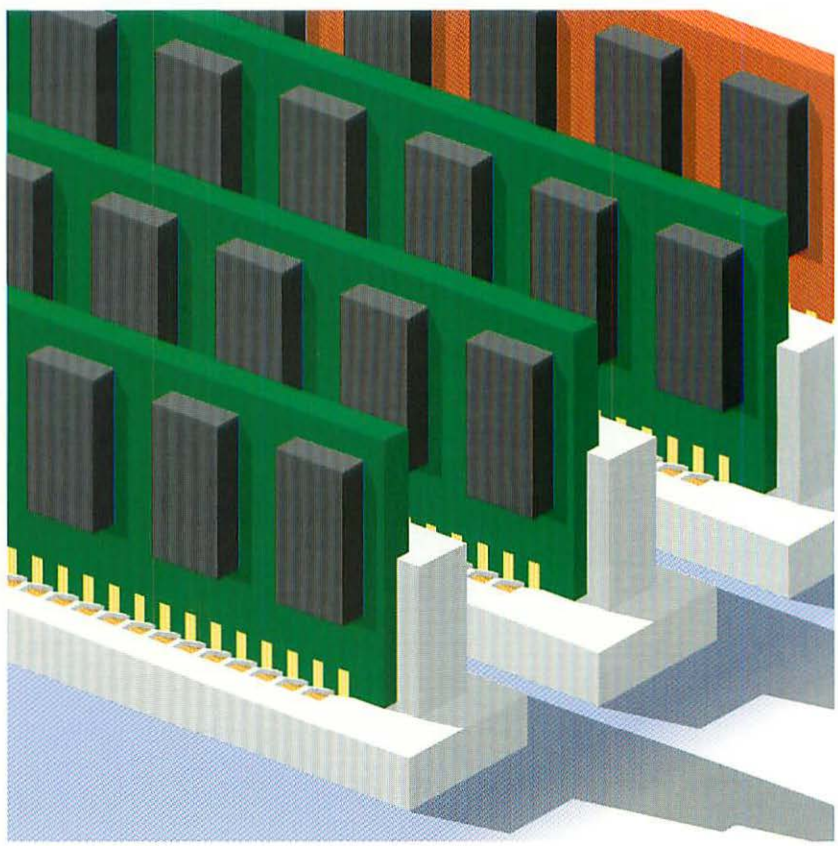


4 The processing units of the CPU can then access the recently loaded information from RAM and launch the application. When an application needs to access the information that is stored in virtual memory, the memory management unit (again, directed by the operating system) brings those portions back into real RAM by swapping them with other, little-used bits of information.



MAC FACT Mac OS isn't the only way to run virtual memory on a Mac. A few applications, including Adobe Photoshop, use their own virtual memory schemes. There are also utilities, such as Connectix RAM Doubler, that use virtual memory in one form or another. Versions of the UNIX operating system for the Mac have run virtual memory since the first 68030 Macs. These include Tenon's MachTen and Apple's now-defunct A/UX.

CHAPTER 14 How a Cache Works



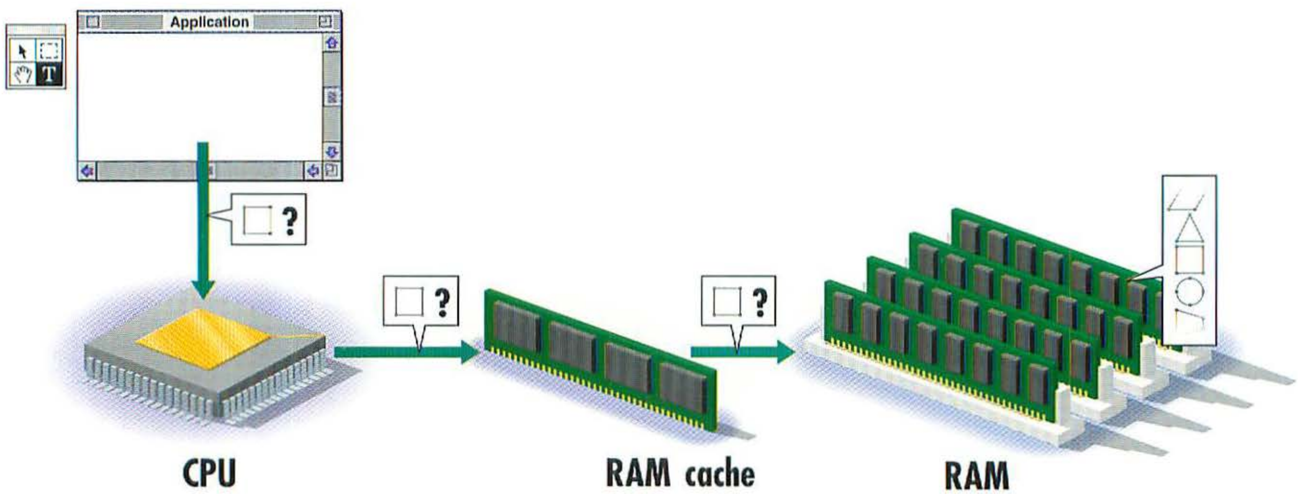
NOT all memory tricks slow things down (as does virtual memory). There is one trick called caching that helps applications work faster and more efficiently. A *cache* (pronounced "cash") is a temporary storage place that sits between the CPU and a storage medium. Its purpose is to feed the CPU information as fast as it requests it. This means that the CPU doesn't have to wait, and performance is improved.

Caches are used in a lot of places in and around your Mac, such as in the CPU itself (as we saw in Chapter 11), as well as in external storage devices such as hard-disk drives and CD-ROM drives. There are two main types of cache that are used in the Mac: a RAM cache and a disk cache. As the names imply, a RAM cache speeds up RAM access, and a disk cache speeds up disk access.

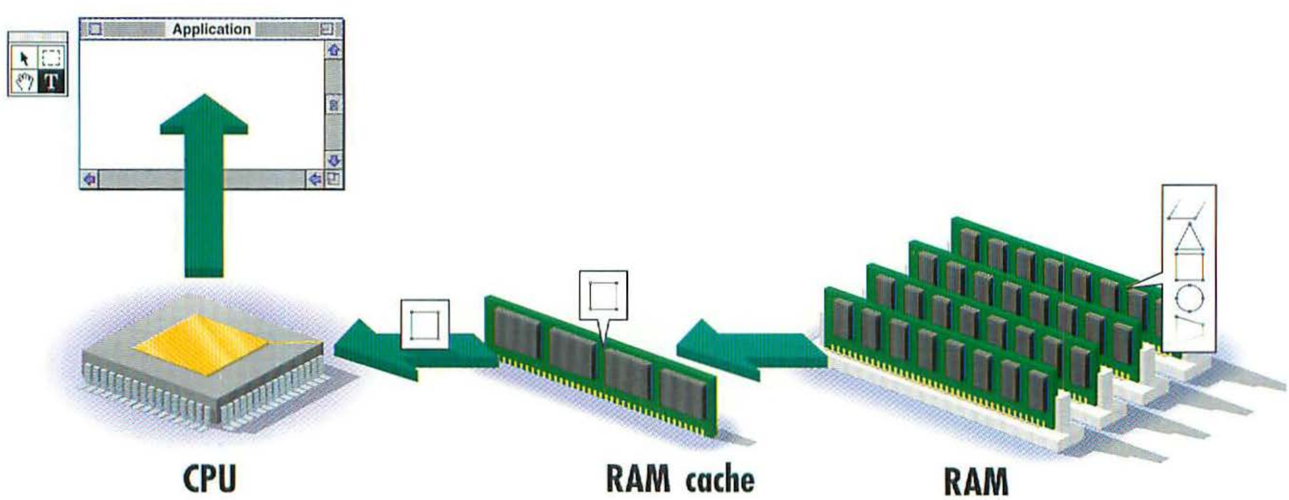
RAM cache is sometimes called L2 cache, short for level 2 cache. This is because the cache inside the microprocessor is considered the level 1 cache. Some Mac models come with a RAM cache already preinstalled. Other models come with an empty slot that can accept a small RAM cache add-in card. Some lower-end Mac models don't use a RAM cache. On these machines, you can often find a microprocessor accelerator board that contains a RAM cache. The cache medium is usually faster than the medium from which it is caching, so a RAM cache is often made of high-speed static RAM, which is at least four times faster than the dynamic RAM used in main memory. Static RAM is expensive, but you don't need much—a 256K or 512K RAM cache can boost performance anywhere from 15 to 60 percent. At a few hundred dollars, adding a RAM cache is the best bang-for-the-buck performance upgrade you can buy.

A disk cache uses a piece of the Mac's RAM to temporarily hold information from a disk storage device. As was mentioned in Chapter 13, it is several hundred times faster to access information from dynamic RAM than from a hard disk, so using a disk cache improves performance. Disk caching is built into the Mac's system software. (System 6 erroneously labeled it a RAM cache, which caused a lot of confusion. Fortunately, Apple corrected this in System 7.) You can set the size of the cache you want to use in the Memory control panel, but you can't shut it off. The optimum disk cache varies with the types of applications you use, but Apple recommends setting 32K of disk cache for every megabyte of RAM installed.

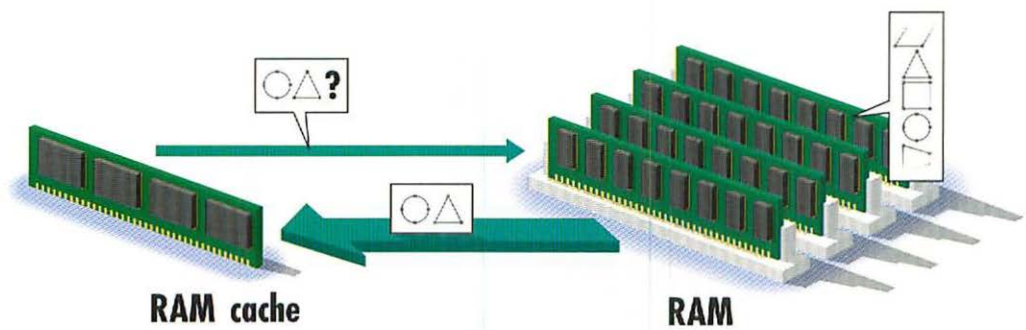
A RAM Cache



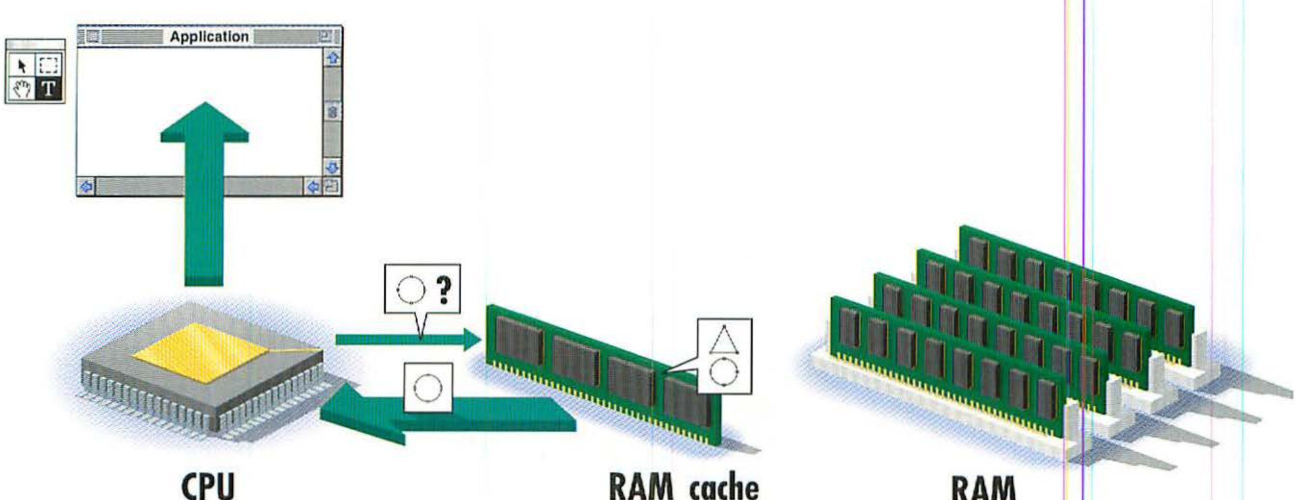
1 When an application requests a piece of information (program code, file data, a resource, or anything else stored in RAM), the CPU sends a query to RAM. The RAM cache intercepts the request and searches itself for the information. If it does not have the requested information, it passes the query along to RAM.



2 RAM passes a copy of the information to the CPU, but it is first intercepted by the cache. The cache makes and keeps a copy, and passes a copy to the CPU.

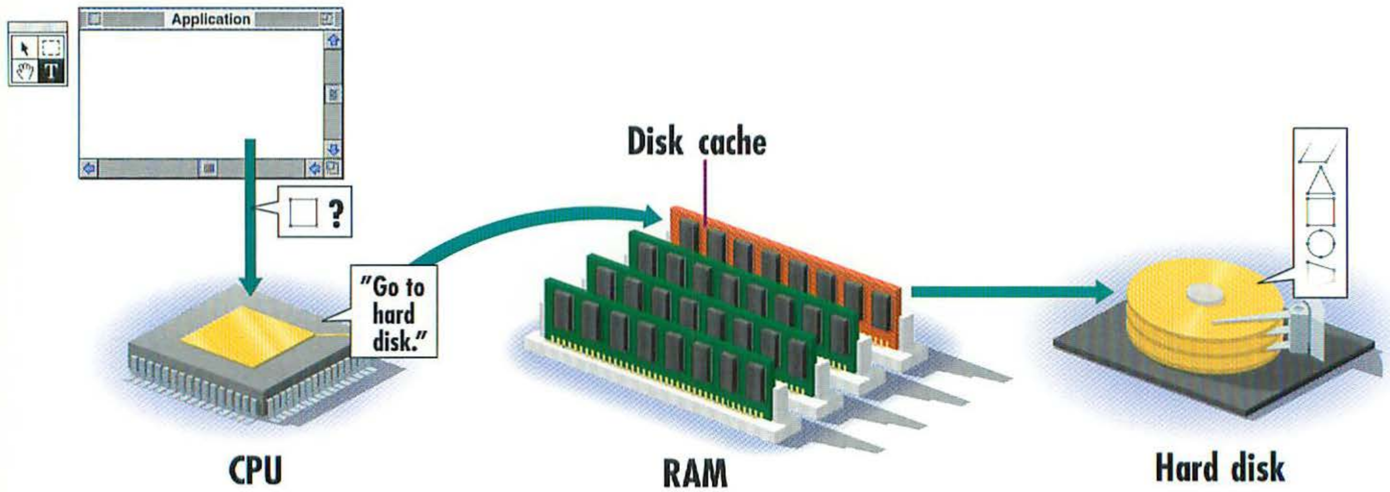


3 The CPU often requires pieces of information that are grouped together in RAM within a short period of time. When the CPU is not being used, the RAM cache fetches information from the RAM addresses near the addresses of the information last requested by the CPU.

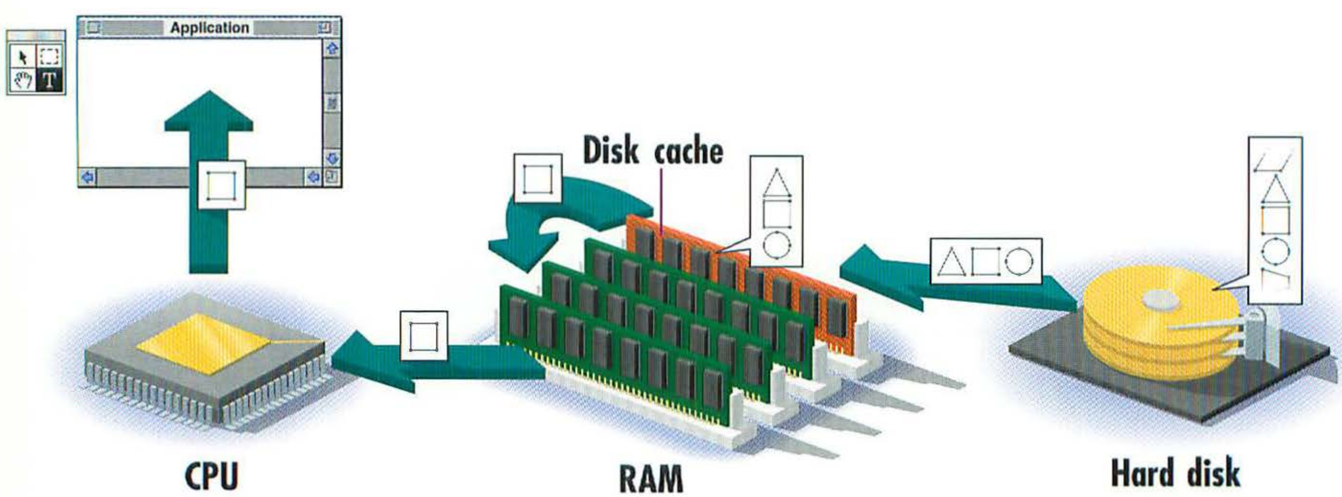


4 When the application requires another piece of information, the CPU sends out another query, which once again is intercepted by the cache. This time, the cache has the requested information and sends it along without having to access RAM.

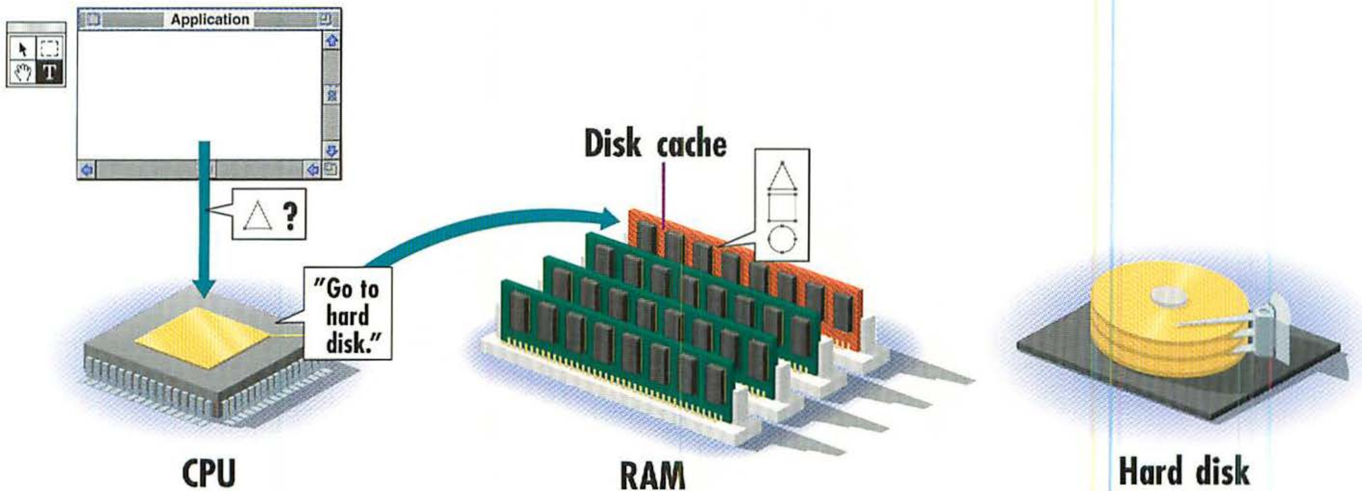
A Disk Cache



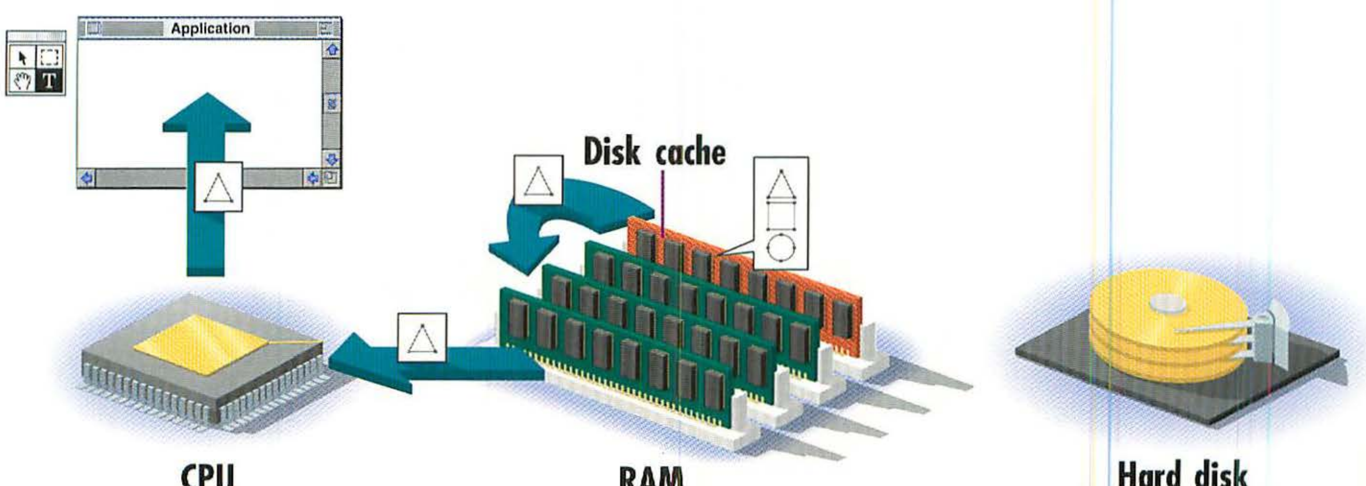
1 Mac OS sets aside at least 32K of RAM to be used as a disk cache, although the user can enlarge this. This space is not available to be used as main memory. When an application makes a request for information that is residing on a hard disk, the disk cache intercepts the request and searches itself for the information. If it does not have the requested information, it passes the query along to the disk.



2 The disk cache reads the requested information from the disk, along with information residing in nearby addresses. The disk cache stores the fetched information in its portion of RAM and copies the requested information into main memory, where it is retrieved by the CPU.

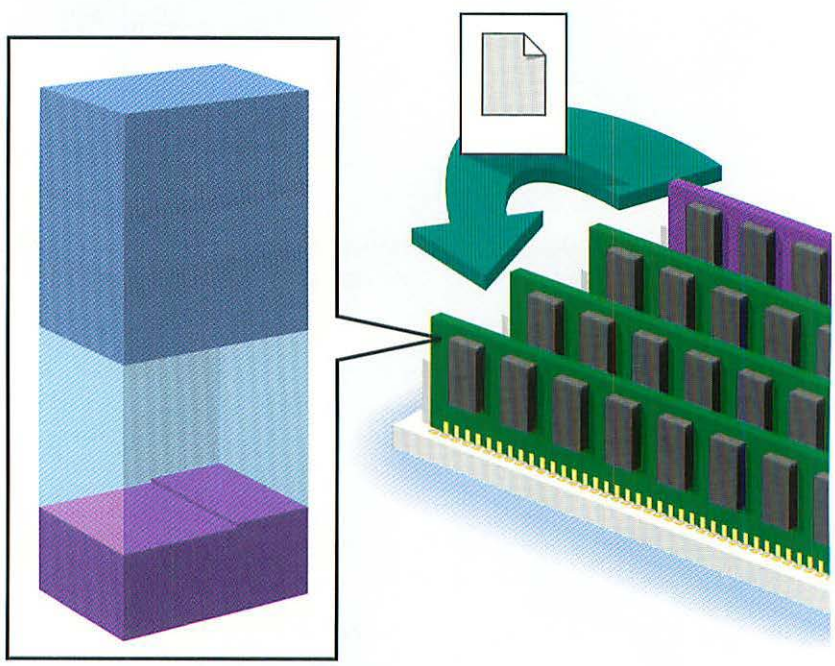


3 When the application requires another piece of information, the CPU sends out another query, which once again is intercepted by the cache.



4 If the disk cache has the requested information, it does not forward the request to the disk. Instead, the disk cache copies the information into main memory, where it is retrieved by the CPU. This is faster than accessing the hard disk.

CHAPTER 15 How a RAM Disk Works



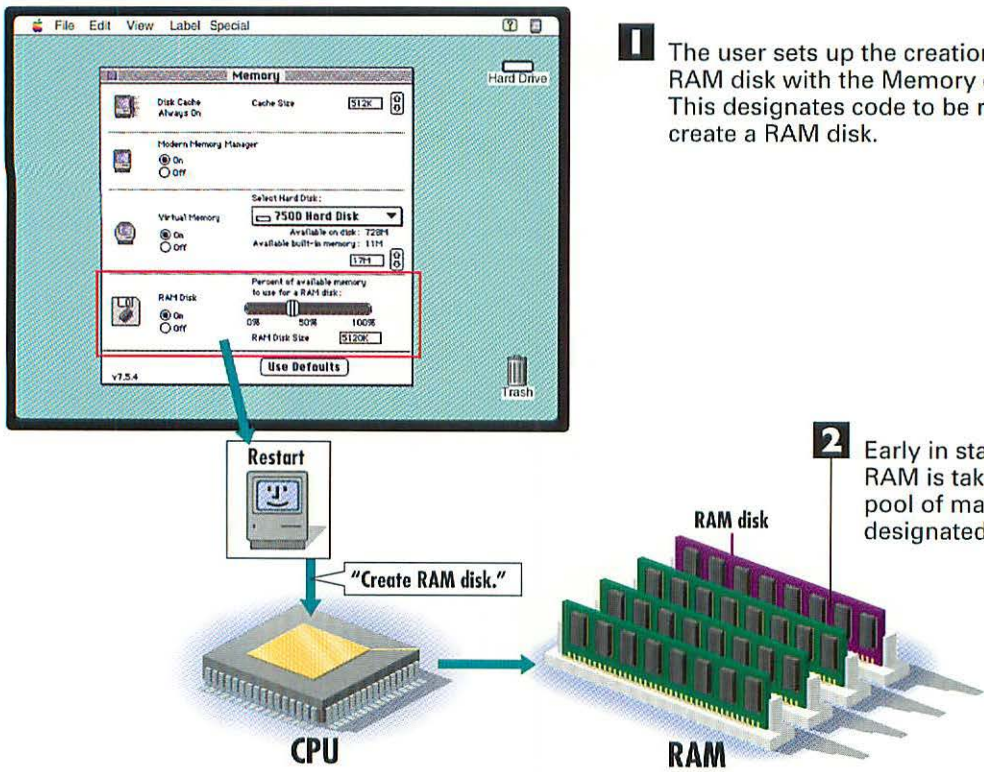
A *RAM disk*, sometimes called a *virtual disk*, is the flip side of virtual memory. Whereas virtual memory takes away a part of a disk and uses it as RAM, a RAM disk takes away part of RAM and uses it as a bootable, desktop-mountable disk. To the user, a RAM disk looks and acts like an ordinary disk drive; you can mount it on the desktop and drag files to and from it. You can create a RAM disk via the Memory control panel. You can also buy add-on RAM disks as expansion boards or external devices. Either way, a RAM disk gives you the speed of RAM with the easy accessibility of a disk drive.

One way to get an idea of the comparative performance of RAM disks and disk drives is to consider *seek time*, which is the time it takes the storage device to locate a piece of data within it. For moderately fast hard-disk drives, the seek time is usually in the 10- to 15-millisecond range. Because RAM has no moving parts, a RAM disk's seek time is almost zero. The lack of moving parts also means that a RAM disk uses far less power than a hard disk, making it a great way for PowerBook users to stretch the life of their batteries.

The principle behind the RAM disk is similar to that of a disk cache (discussed in Chapter 14). A disk cache also takes part of RAM, but the user can't access a disk cache directly, as one can with a RAM disk. The data stored in a RAM disk is not as temporary as data in a cache, which starts erasing old data as soon as it fills up. However, RAM-disk data is not as permanent as disk-drive data, since a RAM disk loses everything stored within it when the computer's power is shut off. (A restart, however, will not erase the contents of a RAM disk.)

There are several ways to get around this problem. One way is to simply copy your work from the RAM disk to your hard disk before you shut down. You can also put your Mac in sleep mode instead of turning it off. This keeps power going to the RAM disk while turning off the hard drive and display. However, since a system crash will erase the RAM disk, it is vital that you back up the data on a RAM disk before you put a Mac to sleep. Some third-party vendors offer software that will load specific files from a hard-disk drive to a RAM disk every time you start up. External devices sometimes provide an uninterruptable power supply that will continue providing power when you shut down the Mac and for a short time after a power failure.

RAM Disk

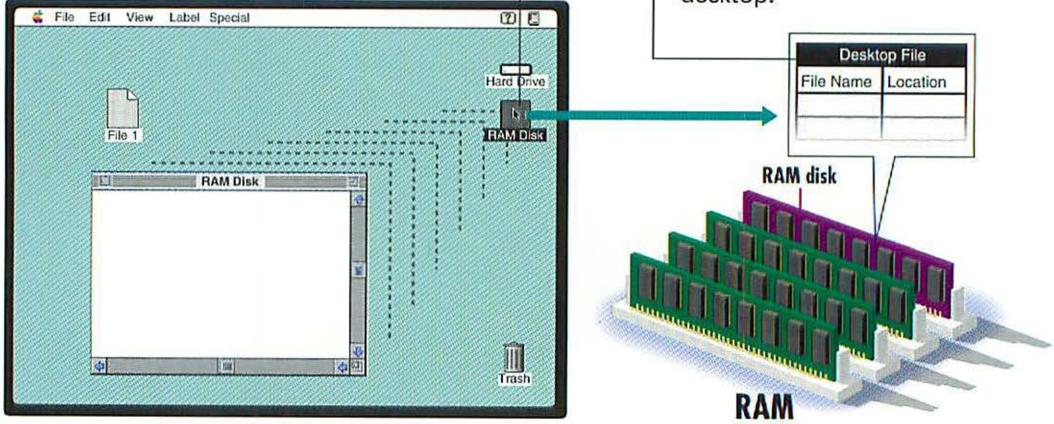


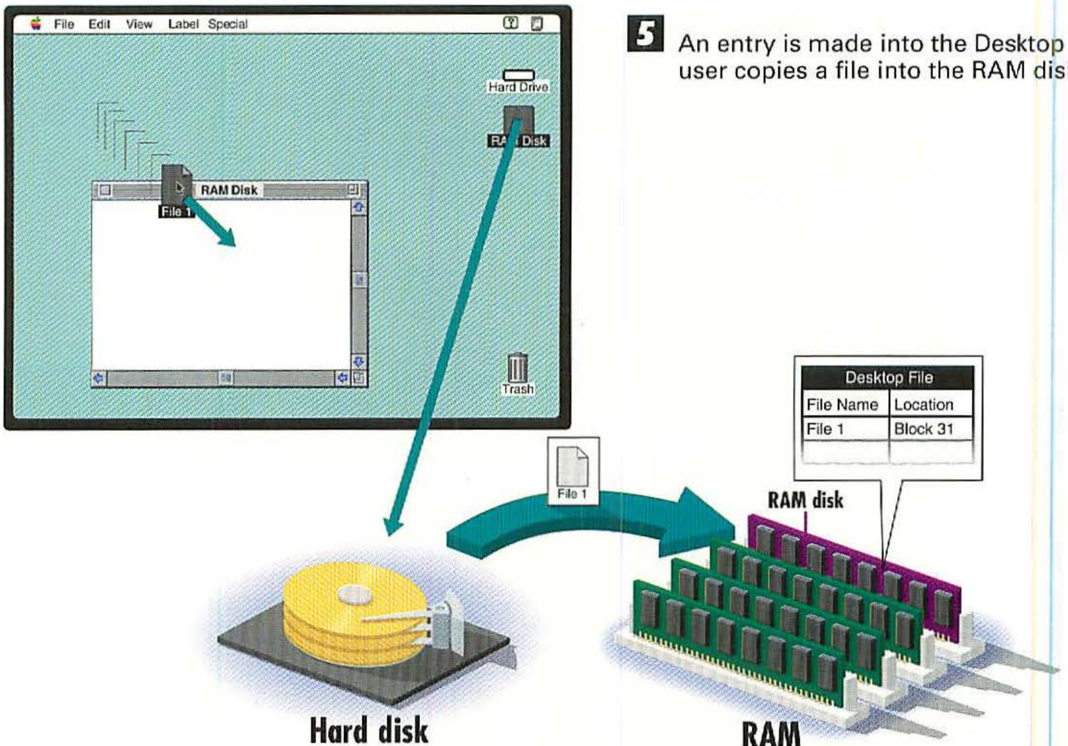
1 The user sets up the creation and size of a RAM disk with the Memory control panel. This designates code to be run at startup to create a RAM disk.

2 Early in startup, a piece of RAM is taken away from the pool of main memory and is designated as a RAM disk.

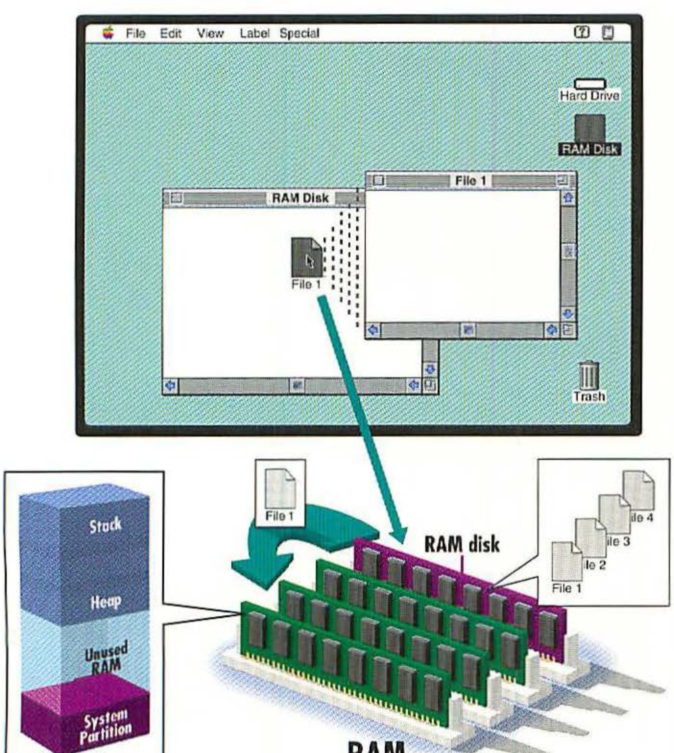
3 A Desktop file is created and stored in the new volume, which mounts on the desktop.

4 When the user double-clicks on the RAM-disk icon, its Desktop file is consulted to see what files it contains, and the window opens. In the case of a new RAM disk, there are no entries in the Desktop file because the RAM disk has no files in it yet.



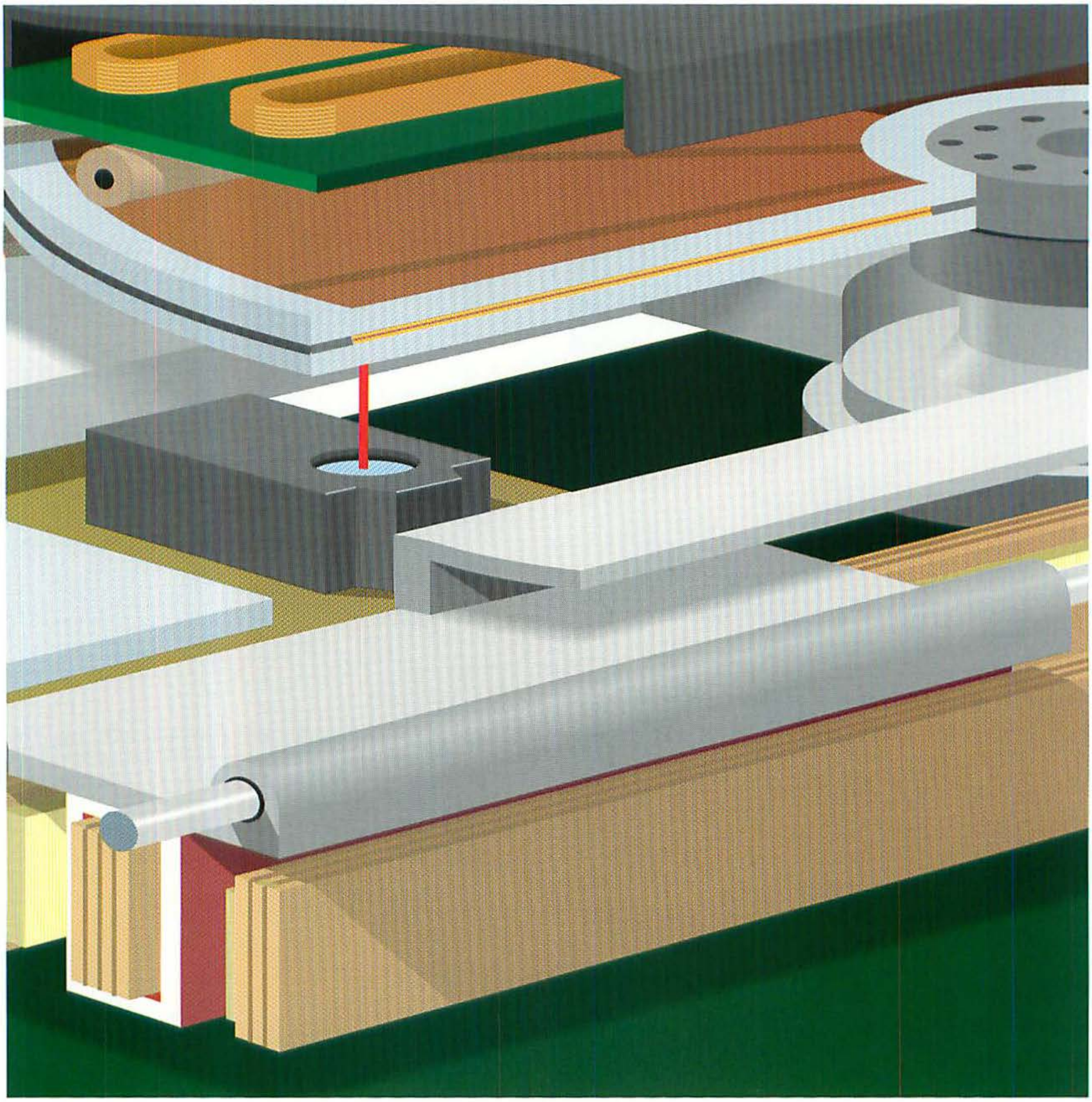


5 An entry is made into the Desktop file when the user copies a file into the RAM disk.



6 Double-clicking on a file in a RAM disk loads the file from the RAM-disk portion into real RAM. The RAM disk stores the file as a document, just as disk drives do, rather than in the stack/heap format of an application's partition in main memory.

MAC FACT To get the most out of a RAM disk, keep a copy of the application you use most frequently and a small System folder on it, and make it the startup disk. Because the System folder contains resources used by applications, putting a System folder on the RAM disk prevents the application from accessing the hard disk. This improves overall performance and increases the battery life in PowerBooks.



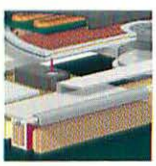
PART 4

DISK STORAGE

FOR long-term memory, the Mac uses the spinning disks inside disk drives, which save data even after the power goes off and the disks stop spinning. These include the hard disks, floppy disks, CD-ROMs, and other types of disks that you use to store your software, your work, and your entertainment.

Disk drives come in a variety of capacities, sizes, speeds, and purposes. Some types of disks offer storage that will last longer than others, while others offer low cost. You can remove some disks (such as floppy disks and CD-ROMs) from their drives for easy transporting. The disks in hard drives are sealed inside an airtight enclosure; trying to get to them will ruin the drive and void the warranty.

Hard-disk drives are the workhorses of mass storage. They are the fastest type of long-term storage media, delivering information to RAM faster than other types of disks. This is why they are the main place you keep your data, applications, and system software. Using multiple platters, hard disks are also some of the largest-capacity storage devices available. Even 3½-inch drives commonly hold several gigabytes.



The original Mac did not have a hard-disk drive—the System folder and applications of 1984 could fit on a single 400K floppy disk. Today, many of the Mac's system extension files are bigger than the first Mac's startup disk.

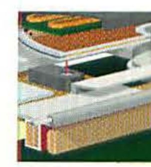
Although still an inexpensive method for software distribution, the floppy disk's storage capacity is increasingly inadequate for today's software. Anyone who has installed a 30MB application from a dozen or more floppy disks can attest to the limitation of floppies. Fortunately, many manufacturers distribute software on a single CD-ROM disc, which has the capacity to hold the equivalent of hundreds of floppy disks. The 650MB capacity of a CD-ROM makes it possible to distribute storage-intensive forms of data, such as video and sound. Because CD-ROM drives are standard in most new Macs and constitute inexpensive add-ons for older Macs, CD-ROM drives are almost as common as floppy drives.

CD-ROM drives are fundamentally different from hard drives and floppy drives in the way they store information. Floppy and hard disks are both made of magnetic media; the drives use electromagnets to read and write the 1's and 0's that make up the data. CD-ROM drives are optical in nature, using laser light to read data from the disc. Optical media is immune to the deleterious effects of stray magnetic fields, which can corrupt or erase data on magnetic media. This makes optical media a more permanent type of storage than magnetic media. In fact, CD-ROM usually is a read-only media; you can't copy anything to it or erase anything from it. However, writable CD-ROMs do exist.

Other types of optical storage devices offer the permanence of CD-ROMs while letting you erase and write data more easily. These erasable optical discs come in the form of a cartridge rather than a naked disc like the CD-ROM. Erasable optical cartridges come in a variety of capacities, from a few hundred megabytes up to several gigabytes.

You can also buy removable cartridges that are magnetic instead of optical. Data written on magnetic removable cartridges is more vulnerable than data on optical cartridges, and magnetic cartridges don't have the large capacities of some optical drives. However, magnetic cartridges are faster than optical disks; this can make a difference in viewing QuickTime movies, for example. The two most popular magnetic cartridge technologies are Syquest drives from Syquest and ZIP drives from Iomega.

Syquest and ZIP drives are quite a bit cheaper than optical drives, but the cartridges generally cost more per megabyte. Another difference between optical and magnetic media is that you can use cartridges from one optical drive in another manufacturer's drive, while the magnetic removable drives are proprietary. You can use a Syquest cartridge only in a Syquest drive, and a ZIP cartridge only in ZIP drive. Generally, people use removable magnetic cartridges for the short-term moving of files and when they need to launch a file from the cartridge itself. Optical storage is better suited for archival storage and for collecting large amounts of data.



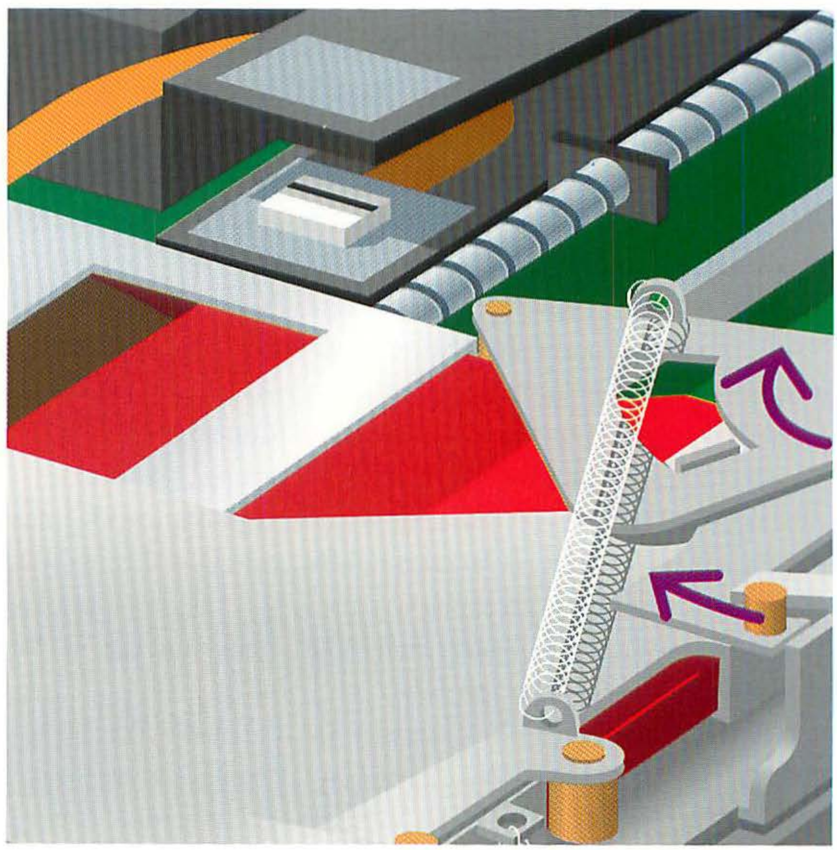
Regardless of their differences, all disk drives behave similarly when connected to a Mac. Each disk appears on the Finder's desktop as an icon called a *volume*. With any disk, whether hard, floppy, or erasable optical, you can use the Finder to view files and folders located on a disk. You also copy files to and from any type of disk using the same drag and drop technique. A process called *formatting* arranges the data on all disks in a manner that Mac OS File Manager can understand. Most disks for the Macintosh come preformatted from the manufacturer according to a standard called the hierarchical file system (HFS). However, you can also insert a non-HFS floppy disk formatted on a DOS or Windows PC. The PC Exchange control panel enables the Finder and the File Manager to recognize and mount these non-HFS floppy disks.

Whether located inside or outside the Mac, most disk drives connect to the computer in the same way, using the Mac's SCSI (small computer system interface hardware, pronounced "scuzzy"). SCSI allows you to connect seven disk drives to your Mac by simply daisy-chaining one to the other with cabling. Although all Macs use SCSI, some models also use a system called IDE (integrated drive electronics) for the internal hard drive. IDE is a fast, inexpensive hardware method of controlling hard disks. IDE doesn't let you daisy-chain, but Macs with IDE drives also have SCSI ports to let you add external drives.

CHAPTER

16

How a Floppy-Disk Drive Works



AT first glance, a Mac floppy disk (known more demurely as a "diskette") is a 3½-inch square piece of stiff plastic, appearing to be neither floppy nor disk shaped. However, if you break open the plastic case, you will find a thin, flexible, plastic disk that is indeed floppy.

There are three types of floppy disks the Mac can use. The Mac's original floppy was a single-sided disk that held 400K. Although today's Macs can read 400K disks, the single-sided disks are now an historical artifact. Apple replaced the 400K disk drive in the Mac 512KE with a drive that reads a double-sided 800K disk. The most common type of floppy, the double-sided, double-density disk, holds 1.44MB. Apple has been shipping 1.44MB floppy-disk drives since the middle of the old Mac SE's production run, and all Macs since then have contained the double-density drives. The 1.44MB floppy-disk drive can also read and write to 800K and 400K disks, as well as to non-HFS floppies formatted for DOS.

Floppy-disk drives have been standard equipment in Macs since the first Mac. Some of the early Mac models had two internal floppy drives, and some had an external floppy port to connect an additional floppy-disk drive. Some PowerBook models don't contain internal floppy-disk drives, but can connect to an external drive.

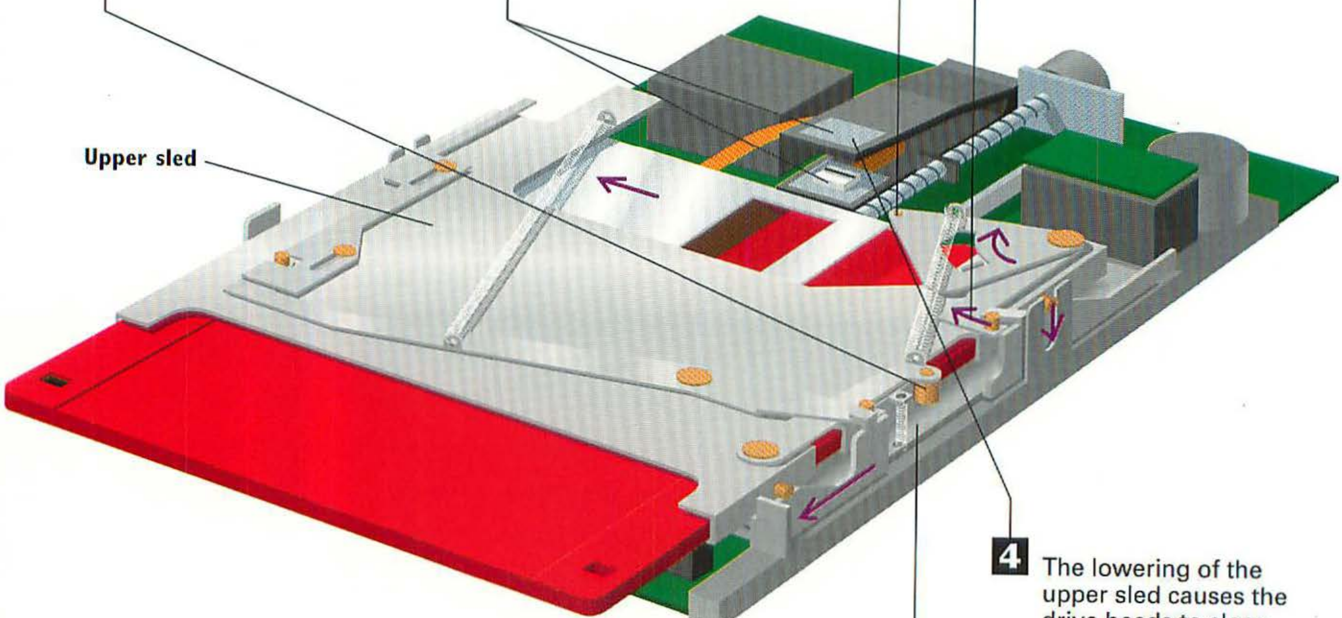
Even in today's Macs, floppy disks aren't known for their speed. Floppy disks spin as fast as 360 revolutions per minute, which is a tenth of the rate (or less) at which hard disks spin. Floppy-disk drives are connected to the logic board via a slow serial connection, which sends data one bit at a time. Mechanically, however, floppy-disk drives are quite sophisticated. The Mac's internal process for inserting and ejecting floppies resembles the operation of a Rube Goldberg device: A gear moves a lever, which fits into a slot, which releases a spring-loaded pin, which slides a sled, and so on. These machinations support the Mac's automatic ejection feature, which shoots the disk out at you when you drag the floppy icon to the Trash. Of course, nothing works perfectly all the time, so the drive also supports a manual method of ejecting a floppy. Experienced Mac users will recognize this procedure as the paper-clip trick: Insert the end of a paper clip into the small hole at the right side of the floppy-disk drive, and the disk pops out. It's low-tech, but effective.

Mac Floppy-Disk Drive

1 When a floppy disk is inserted correctly, the diagonally cut corner (not shown) on the disk case moves aside a pin on a lever, letting the disk pass farther into the drive. If the disk is inserted backwards or upside down, the pin will hit the straight edge of the disk and prevent it from entering the drive.

2 When the disk is almost inside the drive, the edge of the disk's shutter door hits a pin attached to a spring-loaded lever. As the disk moves farther in, the lever pivots clockwise, pushing the pin to open the disk's metal shutter and exposing the thin, flexible disk inside.

3 This same pivoting lever moves another pin that holds the lower sled in place. With the pin moved, springs move the lower sled toward the front of the drive, causing the spring-loaded upper sled to fall. This moves the floppy below the level of the insertion slot.

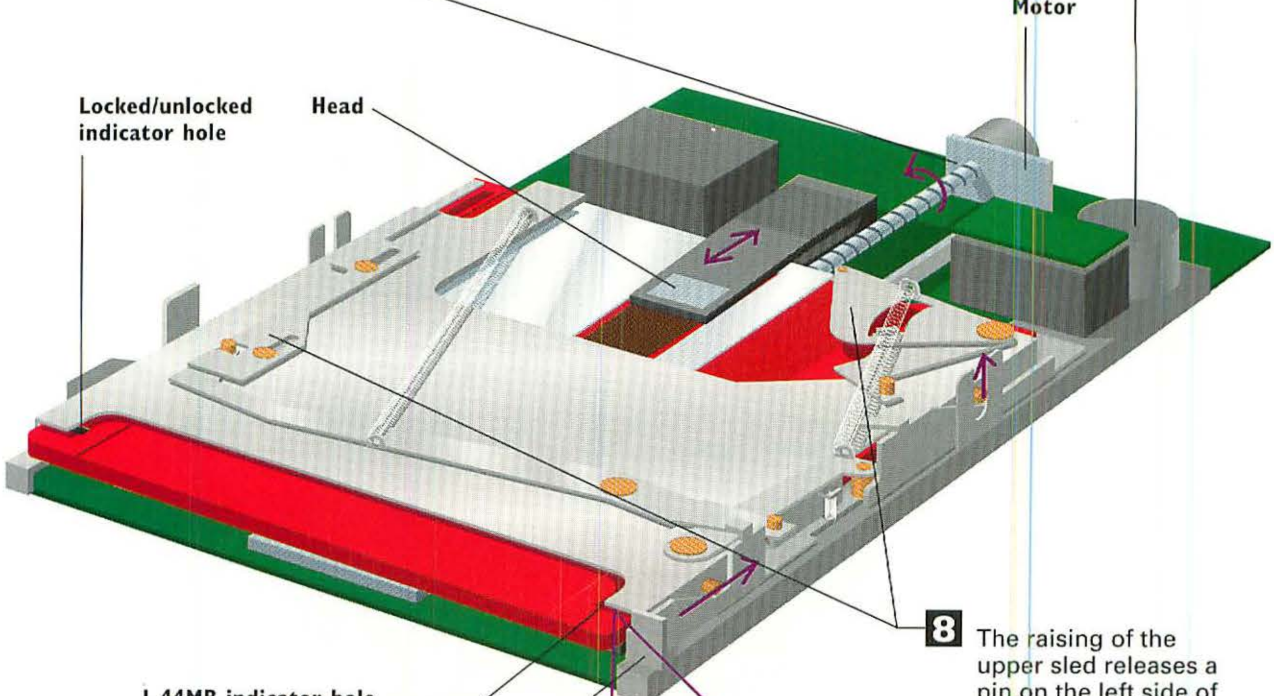


4 The lowering of the upper sled causes the drive heads to close in on the disk. At the same time, a turntable under the center of the disk spins up.

MAC FACT The Macintosh was the first computer to feature 3½-inch floppy-disk drives. Before the Mac, the vast majority of personal computers used flexible 5¼-inch floppies, which are not as convenient or sturdy. Today, 3½-inch floppies are widely used on all types of desktop and laptop computers.

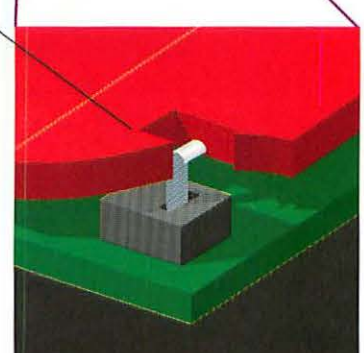
7 When the operating system tells the drive to eject the disk, a motor turns a small gear (not shown) that pulls the lower sled away from the front of the drive; this raises the upper sled. When you insert a paper clip, it presses the manual eject plate, moving the lower sled manually.

6 A motor turns a corkscrew gear that moves the heads back and forth across the spinning disk to read and write data.

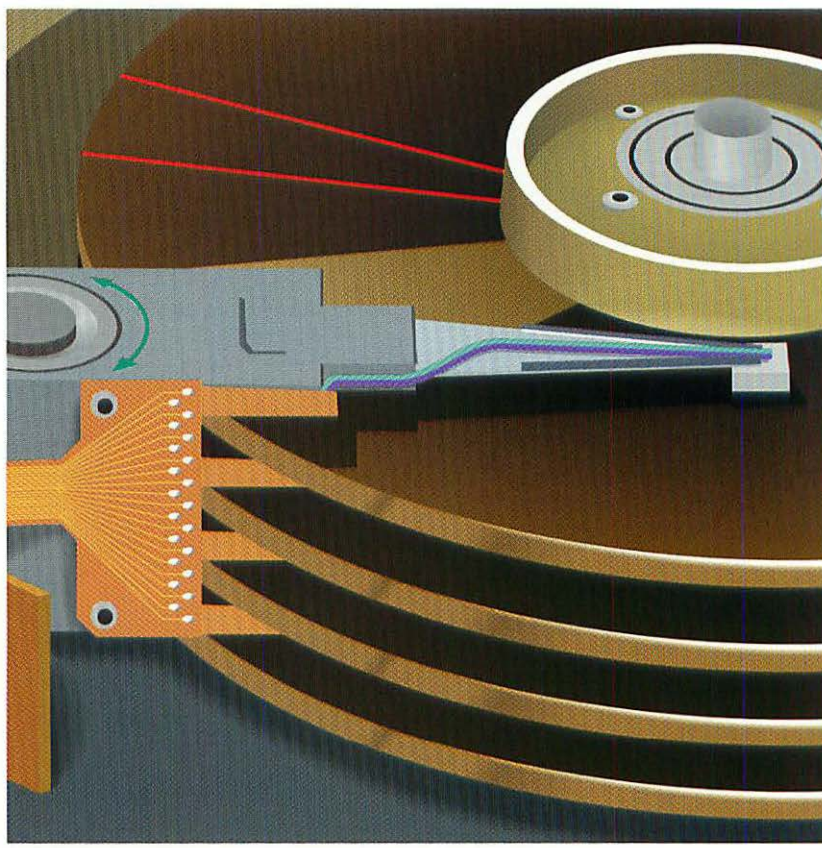


8 The raising of the upper sled releases a pin on the left side of the disk (not shown), causing spring-loaded levers on each side to push the disk out of the drive.

5 At each corner of the disk, a small arm attached to a switch comes in contact with the disk. A square hole on the right side (in our view) of the disk allows the arm to go through it, and the switch is not activated. This sends a signal to the Mac that this is a 1.44MB floppy. (If the floppy does not have a hole in this corner, the switch is activated, telling the Mac that it is an 800K floppy.) A similar hole on the left side contains a small shutter. If the shutter is open, the arm goes through the hole, signaling a locked disk and telling the operating system to prevent the user from writing data to or erasing from the disk. A closed shutter indicates an unlocked disk.



CHAPTER 17 How a Hard-Disk Drive Works



THE hard-disk drive is the method of data storage most Mac and PC users continually depend upon to hold and retrieve software and the work we produce. A high-precision machine, the hard-disk drive is the fastest mechanical storage device available, second only to RAM (which is nonmechanical) in speed, but far less expensive.

Inside a humming hard-disk drive are from one to many spinning rigid aluminum platters, coated on both sides with a magnetic material. The coating contains particles of oxides of iron or other magnetic material, which hold the 0's and 1's of binary data. Data is written on the disks by magnetizing areas on the disk surface, similar to the way information is recorded and read on audio and video tape.

The disks spin together at a constant rate somewhere between 3,600 and 5,400 revolutions per minute (rpm), depending on the drive, though new technological advances are enabling manufacturers to build faster drives every year. By comparison, floppy disks spin at variable speeds at around 360 rpm.

Moving rapidly back and forth over the surface of the disks are tiny electromagnets called *heads*, which read and write data. The very high precision of a hard-disk drive is apparent when you consider the fact that a 3½-inch disk spinning at 5,400 rpm is passing under the heads at the equivalent of more than 55 miles per hour. Like a car crash at these speeds, a *head crash*, which occurs when a head plows into a platter, is very destructive. A head crash can occur when a platter wobbles due to old age or from a jolt to the drive.

A head crash can also be caused by a speck of dust. The distance between head and platter is only a matter of several dozen microns (millionths of a meter). At this scale, a speck of dust is a huge boulder that can carve a trench in the disk media, ruining swaths of data. To prevent dust getting anywhere near the platters and heads, they are permanently encased in an airtight metal case, which also conducts heat away from the disks.

The platter-and-head assembly is attached to a circuit board containing the drive's controlling electronics, which together are referred to as the hard-disk *mechanism*. The mechanism can be placed directly inside the Mac or in an external case along with a power supply and fan.

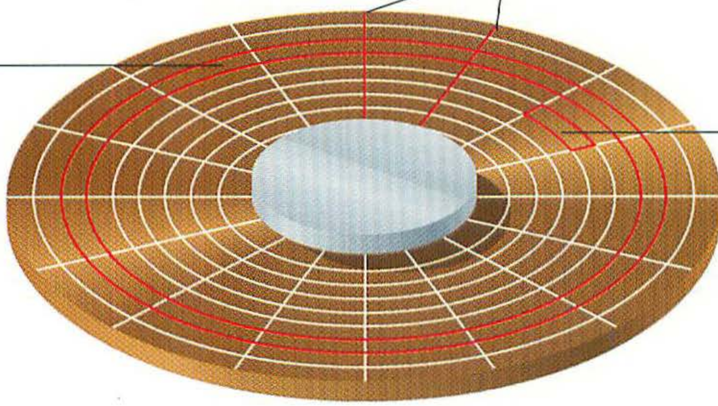
Before any data can be written, the disk must be formatted with software to set up the organizational structures in which the data will be arranged on the disk. Formatting also creates information on the disk that is loaded into the Mac at startup. This information helps the Mac locate files stored on the drive.

Hard-Disk Drive Formatting

Physical Formatting

1 When a hard disk is formatted, concentric rings called *tracks* are created on the disk. They are similar to the areas on a phonograph record taken up by individual songs. Tracks on a hard-disk drive are much thinner than phonograph tracks; hard-disk platters can contain 600 tracks per inch or more. Tracks across multiple platters are called *cylinders*.

2 Tracks are physically divided into *sectors*, which are areas that can contain 512 bytes of data each. In standard formatting, sectors are arranged in pie-like slices, so that sectors near the center take up less disk space than sectors near the edge of the disk. However, all sectors contain the same amount of data in standard formatting.



3 The size of a disk determines the size of a *block*, which is the smallest amount of data transferred at one time by the drive. The larger the disk, the larger the block. A block can be the width of one or more whole sectors, but never a fraction of a sector.

Logical Formatting

4 After the *physical formatting*, the formatting software performs *logical formatting* to create various directories and indexes on the disk that tell the Mac where to find files. Logical formatting creates five areas: the boot blocks, the volume information blocks, the extents directory, the catalog tree, and the data area. These "structures" take up disk space, which is why a disk with no files on it will not have its full capacity available to the user.

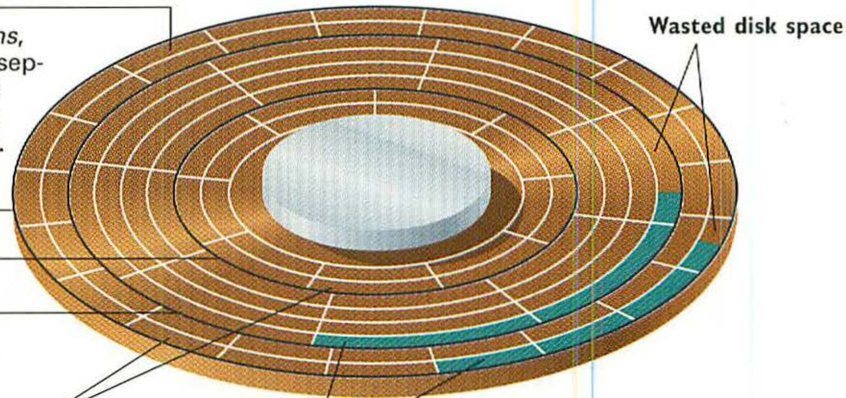
MAC FACT Some advanced formatting software will let you specify the size of blocks, so you can keep them small on large partitions. This makes for a more efficient use of hard-disk space. However, small blocks on large partitions can slow down performance with certain applications due to the processing overhead of keeping track of more blocks. Optimization utilities are available that move your system and application files to the faster outer tracks for better overall performance.

9 The *data area* takes up most of the disk and is where you store your files.

8 The *catalog tree* is a directory that stores the locations of files on the platters. The Mac operating system uses the catalog tree to locate files when needed.

Partitions

1 Most formatting software can create *partitions*, which are sections of a disk drive that act as separate Mac volumes of a fixed capacity. A disk partitioned into three volumes will appear as three hard-disk icons on the Finder's desktop.

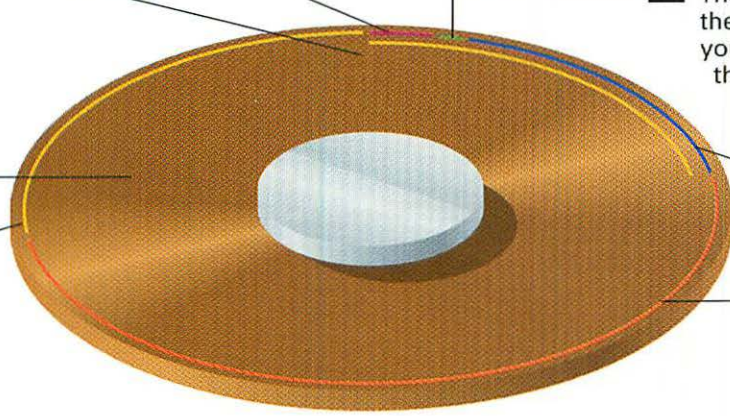


2 Block sizes get bigger with the size of the partition. For instance, partitions of 20MB or less use 1 sector per block (512 bytes per block). Partitions of up to 60MB contain 2 sectors per block (1,024 bytes, or 1K, per block), and partitions of up to 100MB use 3 sectors per block (1.5K per block).

3 Smaller partitions (which use smaller blocks) tend to be more efficient than larger partitions. This is because files can only be stored in whole blocks; any space left over in a block is wasted. For instance, a 3.5K file will occupy four 1K blocks, taking up 4K of disk space and wasting 0.5K of disk space. However, on a bigger partition, the same 3.5K file would require three 1.5K blocks (for a total of 4.5K of disk space), wasting 1K of disk space.

5 The *boot blocks*, which are always the first two blocks of the disk, identify the disk as a Macintosh disk and contain information used during the startup process. They also contain the SCSI driver, which enables the Mac to communicate with the hard drive over the SCSI bus.

6 The *volume information blocks*, which always follow the boot blocks, contain the volume name (the name you give to your disk) and number of files stored on the disk. This is followed by the volume bitmap, which identifies the used and unused blocks.



7 The *extents directory* contains the location of blocks that are located next to each other, called *contiguous blocks*. When you copy a file to a hard-disk drive, the extents directory looks for contiguous blocks to write the file to.

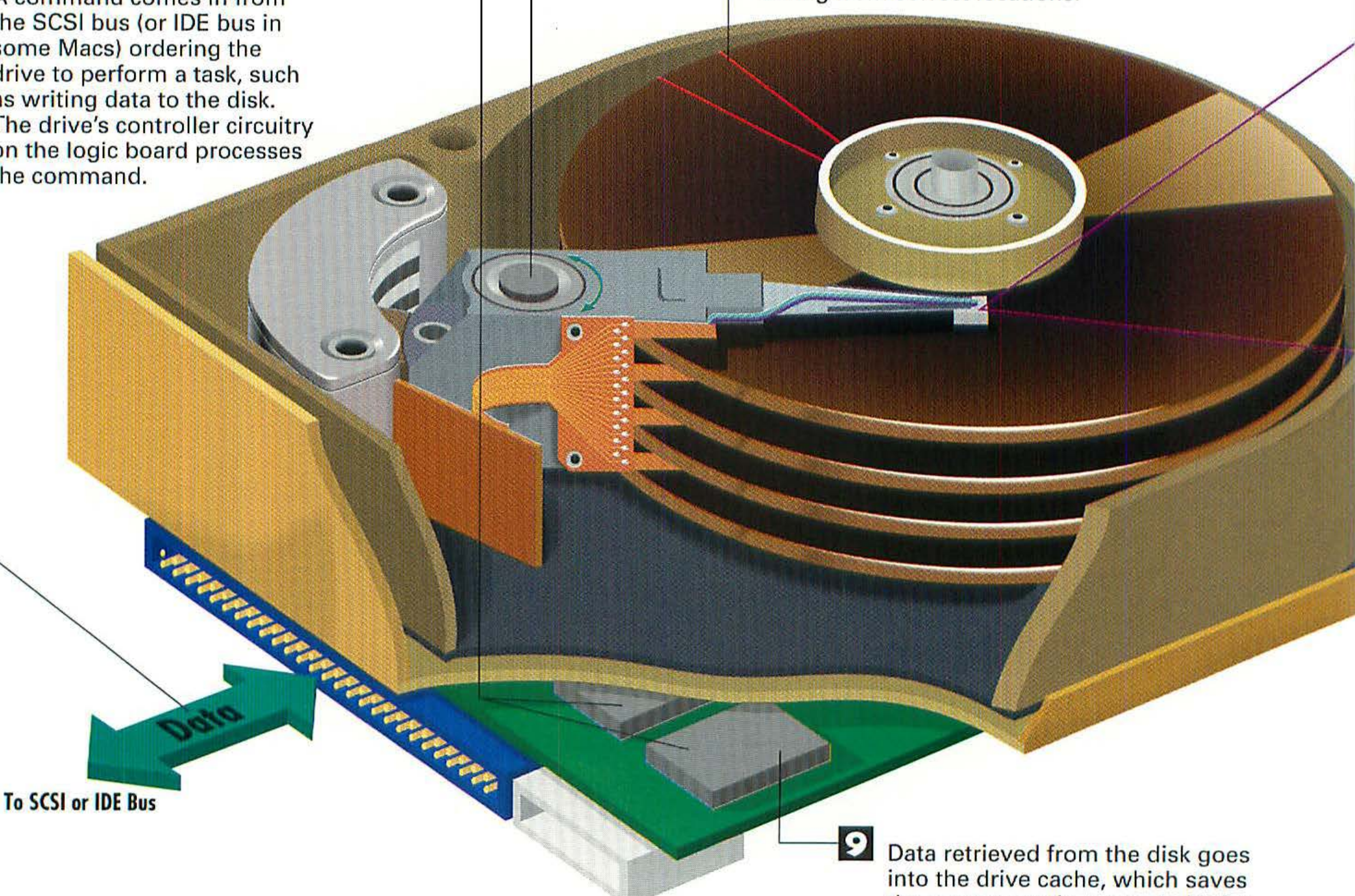
Hard Disk Operation

2 Some drives store incoming write commands in a write cache to speed up performance. The *write cache* holds the write command while telling the Mac's CPU that the task has been completed, and it writes the data when it gets a chance. The benefit is that the CPU doesn't have to wait for the actual writing to take place before sending more commands.

3 The drive's logic board sends an electric current to the head actuator motor to move the heads rapidly back and forth over the spinning platters. All the heads are connected to the same actuator, and they move in unison. The actuator is held by a spring. When the current increases, the heads move toward the center of the spinning disks. When the current decreases, the spring pulls the heads back toward the outer edge of the disks.

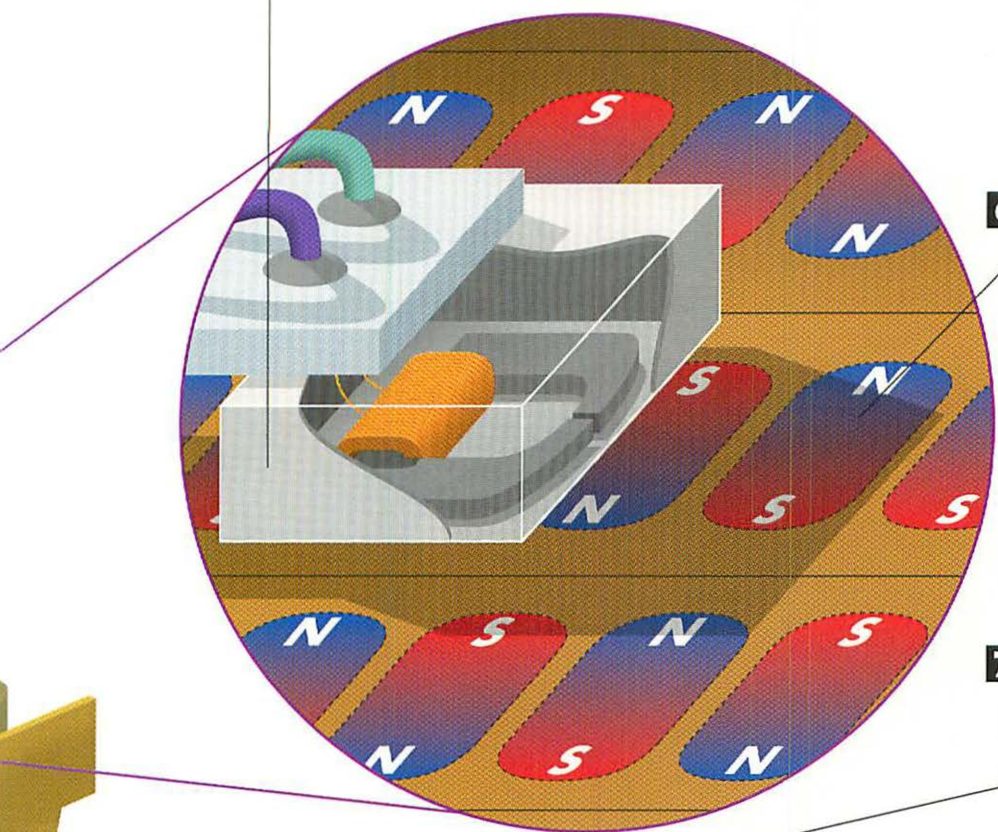
4 Sector boundaries are marked by strongly magnetized lines. The heads count these boundaries to determine which sector they are currently over and to keep from wandering from correct locations.

1 A command comes in from the SCSI bus (or IDE bus in some Macs) ordering the drive to perform a task, such as writing data to the disk. The drive's controller circuitry on the logic board processes the command.



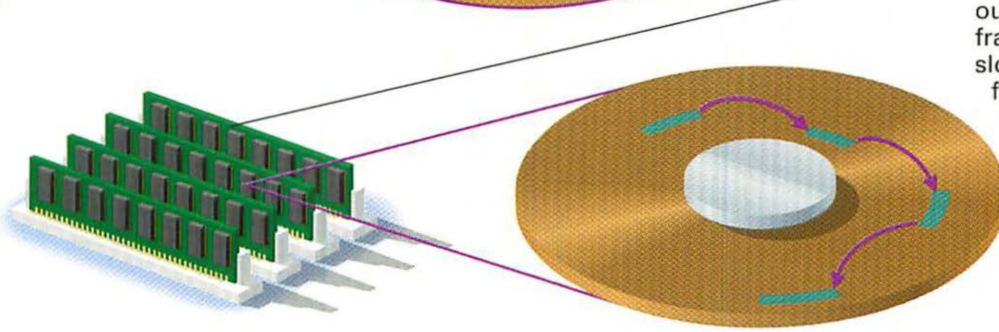
9 Data retrieved from the disk goes into the drive cache, which saves the most recently requested information. If the Mac asks for information that is already in the cache, the drive sends the data without having to access the platters.

5 When the specified sector passes under a head, a current flows through the coil to produce a magnetic field, turning the head into a magnet. The head magnetizes a small area of the disk under the head, so that "north" poles of the magnetic particles in the area are all facing in one direction. This magnetized area represents 1 bit.



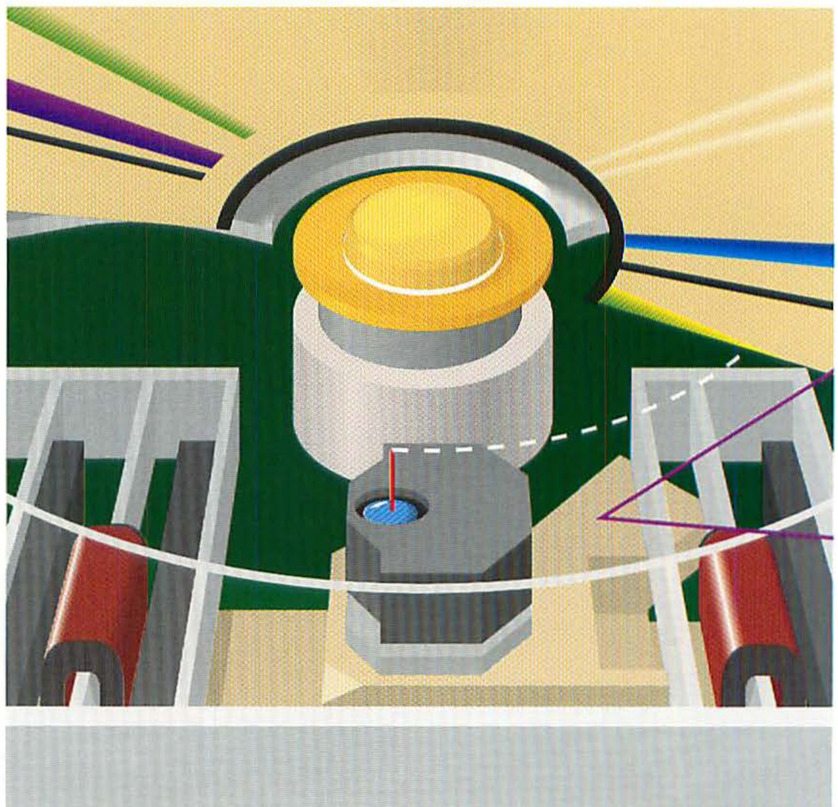
6 Reversing the direction of the current in the coil under another area on the platter reverses the magnetic field, causing the north pole of the area to face the opposite direction. An area with a north pole oriented in one direction represents a 1, and an area with a north pole oriented in the opposite direction represents a 0.

7 The volume bitmap, now in RAM, is used to locate free blocks on the disk. The extents tree looks for blocks that are next to each other to write the file. If most of the files on the disk are stored in noncontiguous blocks, the disk is said to be fragmented. A fragmented disk is slower since the heads must move farther between blocks when reading a file. Disk optimization software can defragment a disk by moving data around so that files are contiguous.



8 Reading data is the reverse of the above process: The catalog directory loaded into RAM is used to find the blocks on the disks where the data is stored. When the heads pass over the magnetized bits, a current is produced in the coil. Current flowing in one direction is read as a 0, while current produced in the opposite direction is read as a 1. This technique of moving magnetic material near wire coils to produce an electric current exploits the same electromagnetic phenomena used to produce electricity in power plants.

CHAPTER 18 How Removable Storage Works



YOU *can* take it with you. Replace the hard drive's multiple platters with a single disk, put it in a removable cartridge, and you have mass storage with the convenience of a floppy disk. Magnetic removable cartridges come in several formats, holding from 44MB to 1GB. The most popular ones use 3½-inch disks just slightly bigger than a floppy.

Now replace the precision mechanics of a hard-disk drive with the precision optics of a laser-guided telemetry system and you have the basis for optical storage. Optical-disc drives use a series of lenses and mirrors to guide a tiny laser beam to microscopic areas on a single rotating disc. The precise aiming of the laser enables data bits to be written much closer together than on magnetic disks. Sony can put a bit in 0.41 microns of space for a data density of 1.3 gigabytes per square inch.

There are two major types of optical storage device in use today. *Magneto-optical storage,* also called erasable optical storage, allows you to read and write data, as you do with a hard or floppy disk. *CD-ROM* (compact disc read-only memory), based on CD audio technology, is used for distributing software, data, and multimedia presentations.

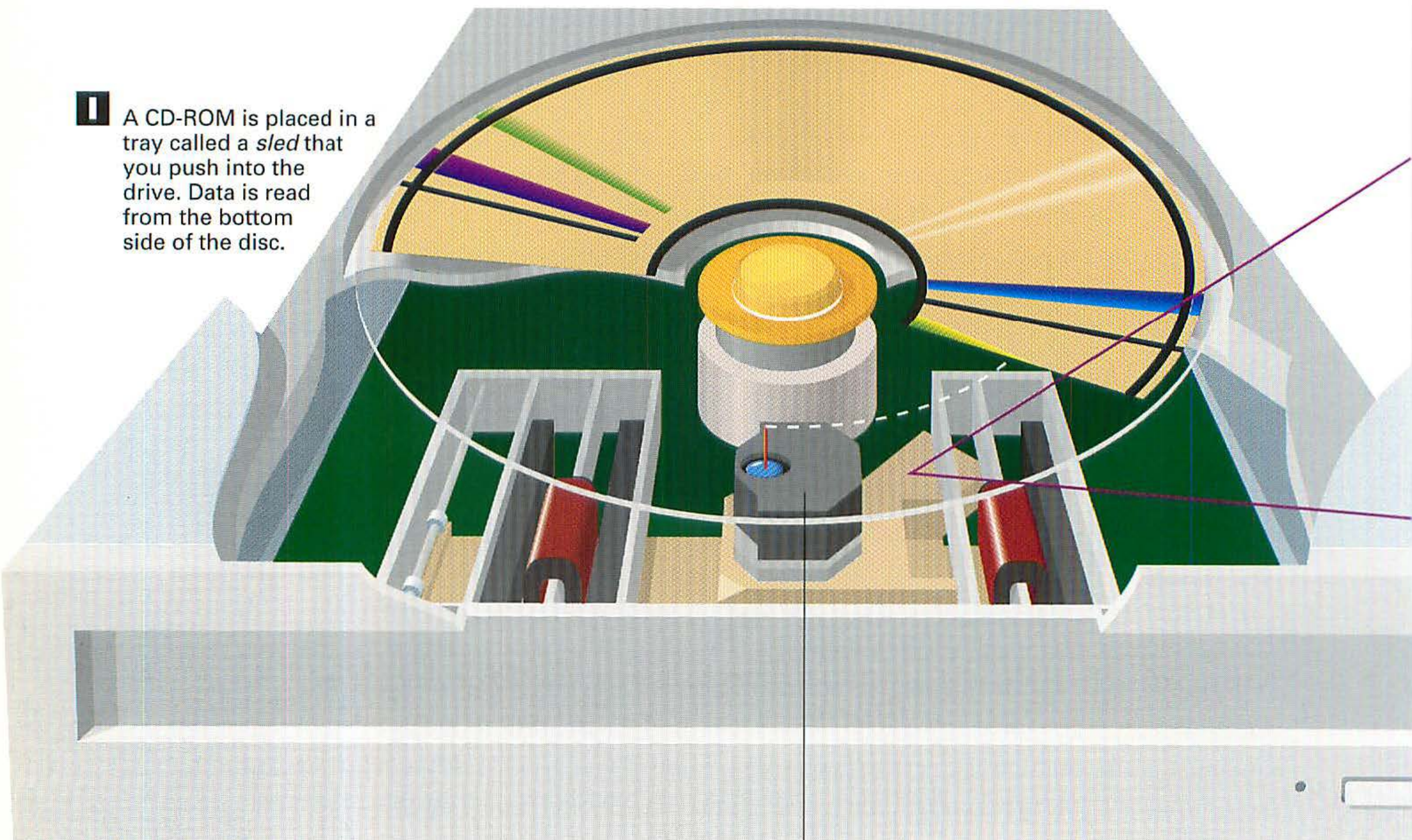
Both types use removable discs. Erasable optical discs come in cartridges of two standard sizes, 5¼ inch, which can hold several gigabytes of information, and 3½ inch, which can hold up to 650MB. CD-ROMs look like audio CDs and can hold up to 650MB of data, or about 200,000 pages of text. In addition to data discs, today's CD-ROM drives can read audio CDs and Kodak's Photo CDs, which hold digital images of photographs.

In addition to holding more data, optical discs offer more durable storage than magnetic disks or tape. Whereas it is not uncommon for hard-disk drives to fail after five years, magneto-optical media is estimated to hold data for 30 years and can sustain 10 million read/write cycles. The estimated lifespan for CD-ROM is 100 years, making it an attractive alternative to film-based storage, such as microfiche and microfilm.

On the other hand, magnetic removable drives are less expensive and faster than optical drives, though they are not as fast as hard drives. CD-ROM is the most popular form of removable storage; it is also the slowest, next to floppy disks. The first CD-ROM drives had a maximum throughput of 150K per second, the same as audio CD. Manufacturers doubled the throughput by spinning CD-ROMs twice as fast in so-called double-speed, or 2X drives. But even today's 4X and 6X CD-ROM drives are still slower than magneto-optical and magnetic drives.

CD-ROM Storage

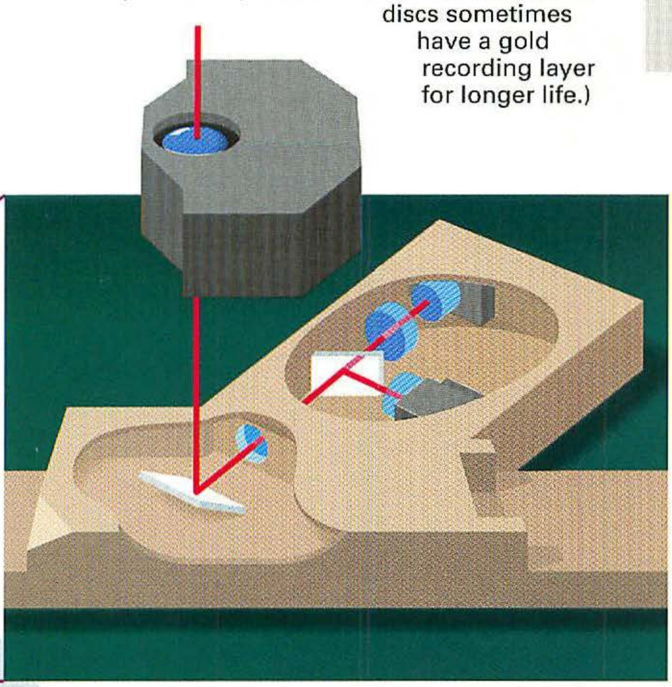
1 A CD-ROM is placed in a tray called a *sled* that you push into the drive. Data is read from the bottom side of the disc.



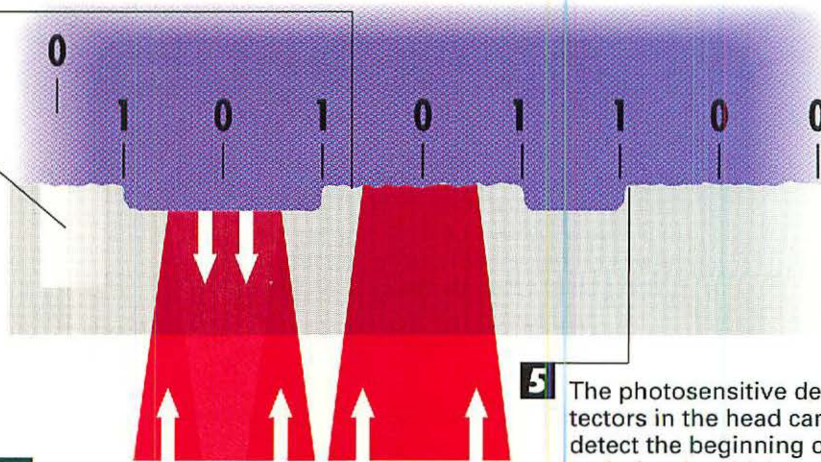
MAC FACT Many commercially available CD-ROMs can be run on both Macs and Windows machines because the data is written in a standard format that both operating systems can understand. The first multiplatform CD-ROM standard is called High Sierra because it was hammered out by a group of manufacturers and software publishers near Lake Tahoe, California, in the Sierra Nevada. High Sierra was later formalized in an international standard called ISO 9660.

2 The read head moves on tracks back and forth under the disc and directs a beam of near infrared light from a laser diode through a series of lenses and mirrors.

3 Light passes through a protective plastic layer on the CD and is reflected off of a recording layer, usually made of aluminum. (Premium discs sometimes have a gold recording layer for longer life.)

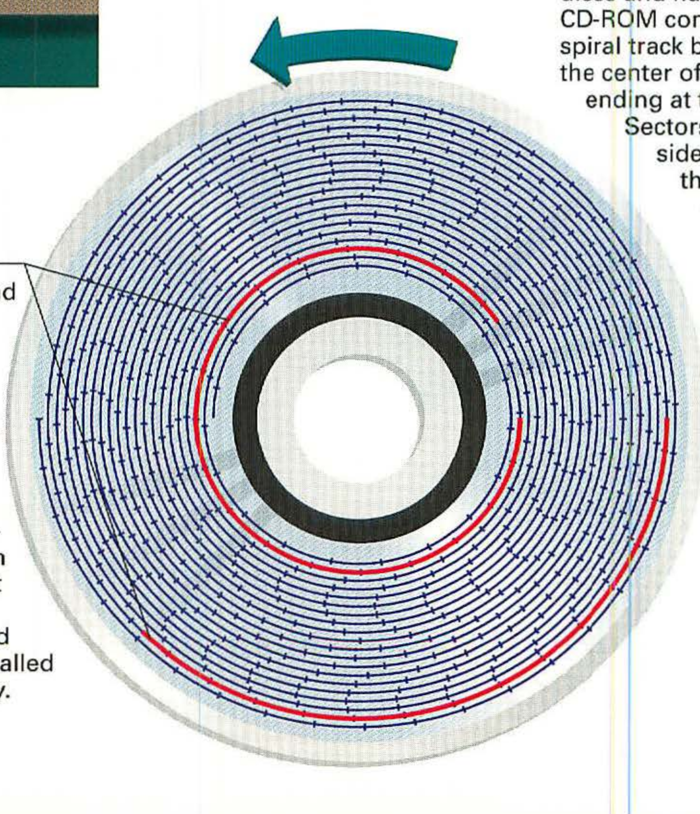


4 Data is recorded as pits pressed into the disc, which average about a micron in length.



5 The photosensitive detectors in the head can detect the beginning or end of a pit, which represents a 1. Areas where there is no change in elevation represent 0's.

6 Unlike magneto-optical discs and hard disks, a CD-ROM contains a single spiral track beginning near the center of the disc and ending at the outer edge. Sectors near the outside of the disc are the same length as those near the inside.



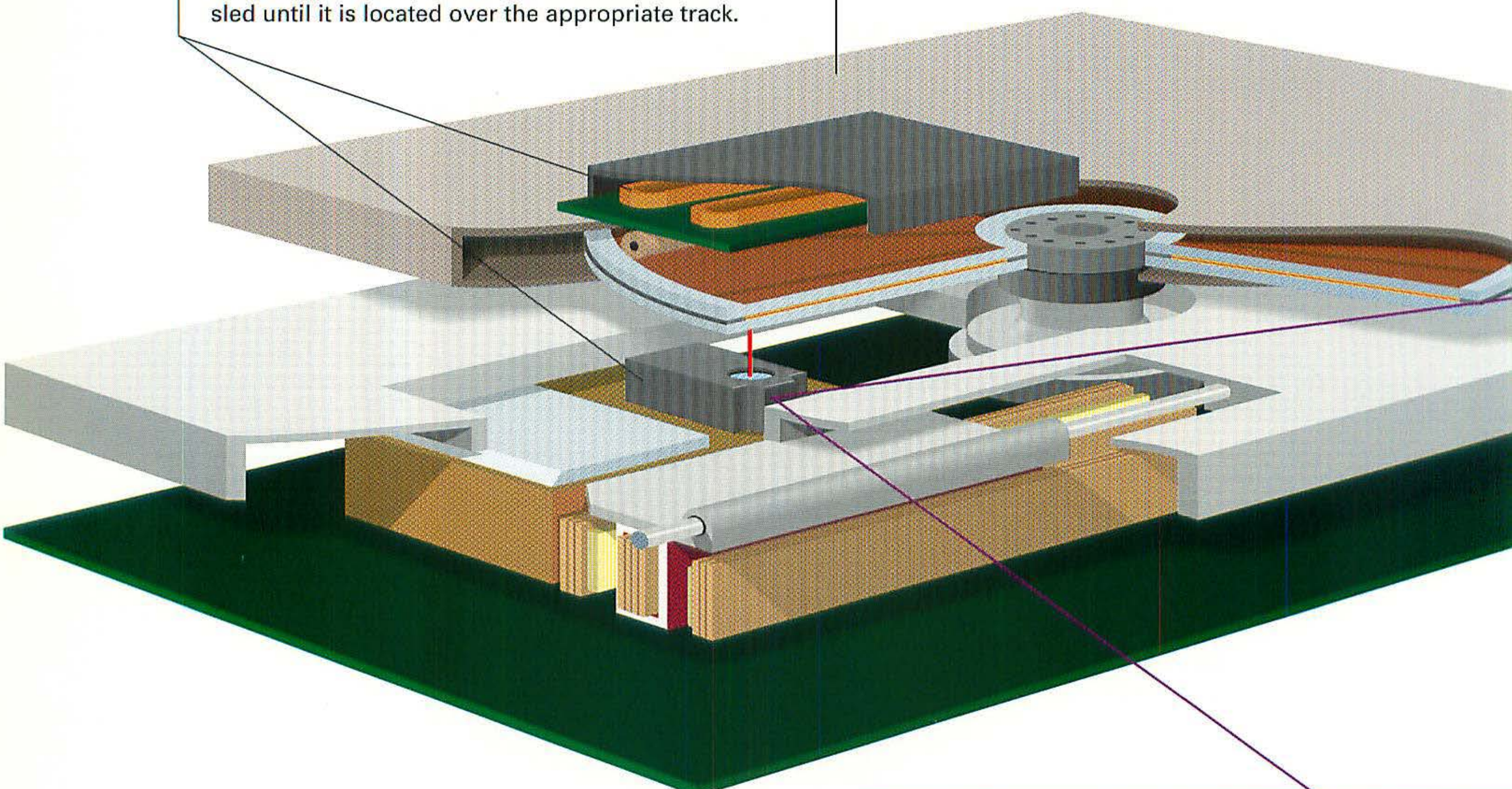
7 Whereas hard disks and erasable optical discs spin at a constant rotational speed, CD-ROM drives spin at a variable rotational speed. As the head moves toward the edge, where there are more sectors per revolution, the disc rotation slows to keep constant the number of sectors passing under the head every second. This is called constant linear velocity.

Magneto-optical Storage

1 Magneto-optical discs are encased in a protective cartridge much like that on a floppy disk, only bigger. When the cartridge is inserted in the drive, the disk spins at a constant speed of 3,000 rpm or more, depending on the drive.

2 To write data, the drive uses an electromagnet and a laser head positioned on opposite sides of the disc. (Hence the term magneto-optical.) The head slides back and forth over the disc on a sled until it is located over the appropriate track.

4 The laser light passes through a layer of protective plastic (or sometimes glass), which is about 1.2 millimeters thick, to the recording layer of the disc. This layer contains a special magnetic material that can change its magnetic polarity only at temperatures above 150 degrees centigrade. To write a bit of data, the laser heats a spot for a short time—about 800 nanoseconds (billionths of a second).

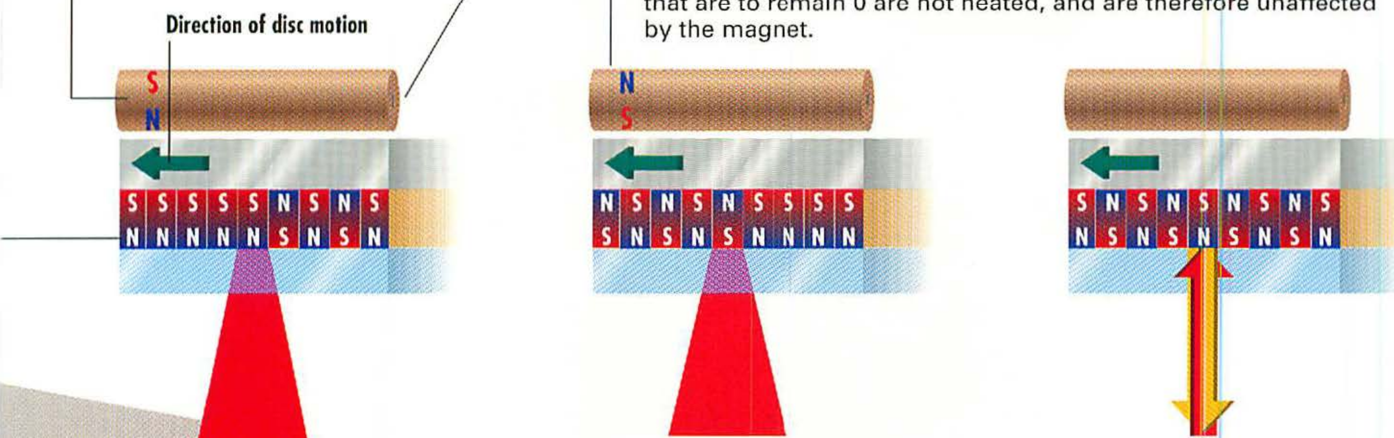


3 The lens head directs a beam of near infrared light from a laser diode through a series of lenses and mirrors. Photosensitive detectors are used to check for accurate positioning of the head and to read data.

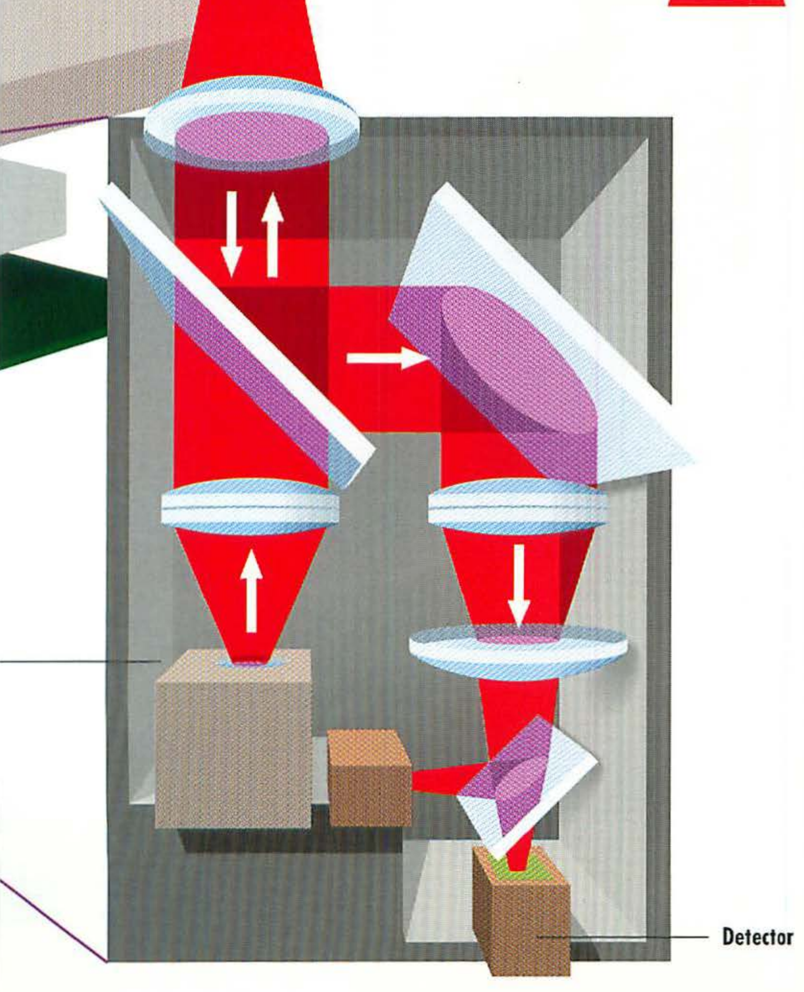
5 At the same time, the electromagnet turns on, creating a magnetic field surrounding the area on the disc representing a bit. Only the bit at 150 degrees centigrade is affected by the magnet.

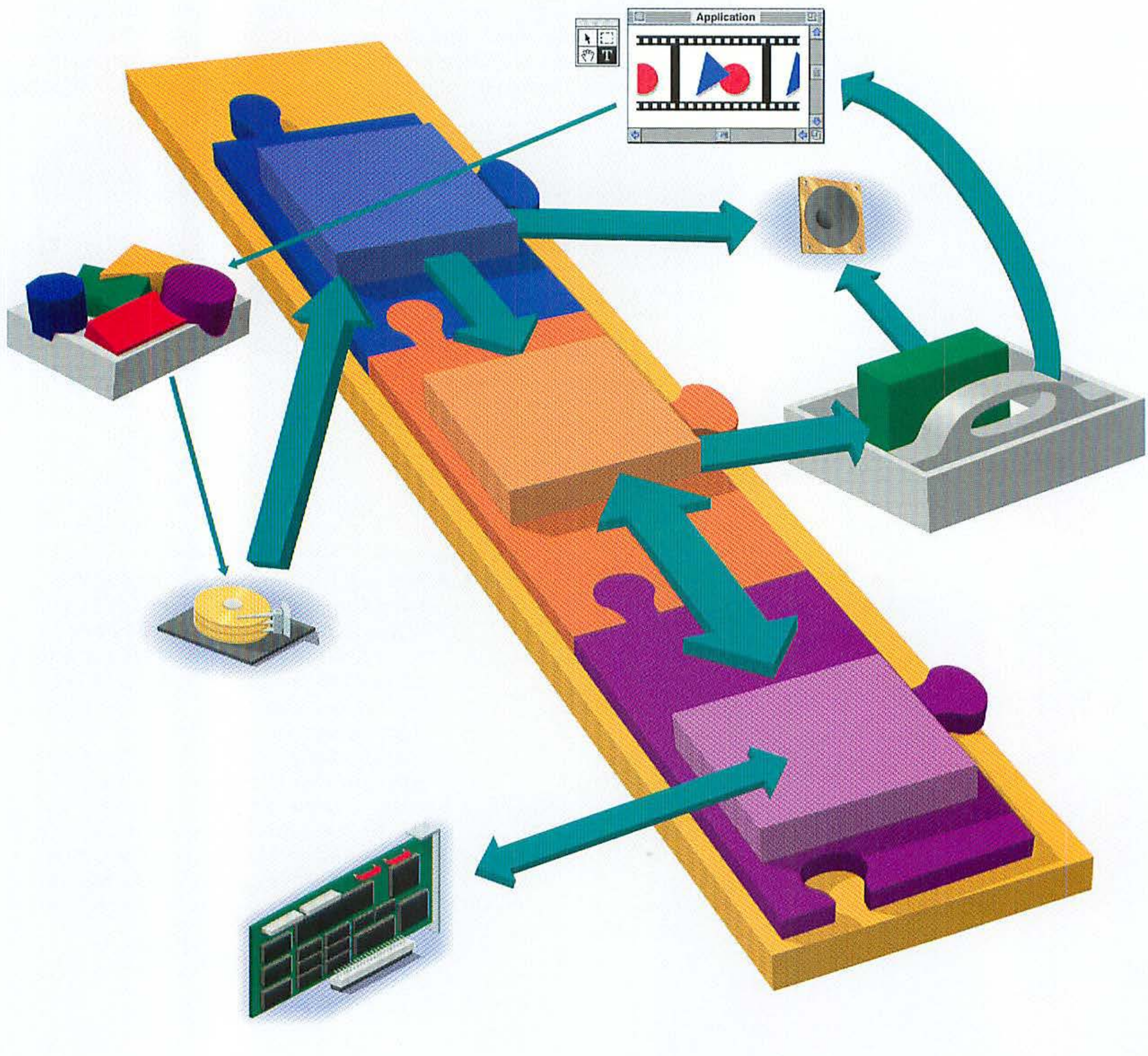
6 To write data, two passes must be made: an erase pass and a write pass. During the erase pass, the magnet is turned on and all bits become magnetized with their north poles facing down, representing all 0's.

7 During the write pass, the magnet switches polarity so that the north pole faces up. Bits that need to be changed to 1 are heated with the laser, and their polarity is reversed by the magnet. Bits that are to remain 0 are not heated, and are therefore unaffected by the magnet.



8 A much less powerful laser beam is used for reading data. The light passes through the protective and recording layers and bounces off of the reflective layer back toward the head. The laser light is polarized, meaning that the light waves are all oriented in one direction. As this polarized light passes through the recording layer, the magnetic material rotates the direction of the light waves. Bits magnetized as a 1 rotate the light clockwise, and 0 bits rotate the light counterclockwise. The photosensitive detectors can recognize the difference.




Application

PART

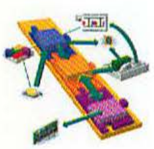
5

GETTING INFORMATION IN AND OUT

THE tangled spaghetti of cables in the back of the Mac is all about communication. These cables connect the peripheral devices that enable us to have a two-way conversation with the Mac's CPU. We send commands and data through keyboards, pointing devices, scanners, microphones, and video cameras. The Mac responds through video monitors, printers, and speakers. Moving data between the CPU and these devices is known in tech-speak as input/output or, simply, I/O.

Other I/O devices expand the capabilities of the CPU, communicating with it even when you are not. Hard-disk and optical-disc drives let you store massive amounts of data, and internal accelerator and video cards can give you faster speed. Modems and network connectors allow your Mac to communicate with other computers. To the CPU and RAM, however, it doesn't matter what type of device is doing the talking—it's all I/O to them, bits coming in and bits going out.

The various I/O devices we can connect to the Mac plug into several different types of ports. A *port* consists of a connector and the circuitry on the logic board that sends and receives signals to and from the devices. Most ports have connectors on the outside of the Mac, but some have connectors on the inside. Others, such as SCSI ports, have both internal and external connectors for internal and external storage devices.



One type of internal device you can add to your system is an *expansion card.* Adding expansion cards lets you tailor your Mac to your own specific needs. For example, someone who needs to run Microsoft Windows can add a 486 or Pentium card to one of the Mac's slots. A scientist can collect massive amounts of data with a Mac by adding an analog-to-digital card. Some expansion cards can even add more ports to your Mac.

To the Mac, an expansion card is just another I/O device. You plug expansion cards into an *expansion slot,* which is really just another port with its connector on the inside of the Mac. Most Mac models contain one or more of three types of general-purpose expansion slots found in Macs—the processor direct slot (PDS), NuBus, or the peripheral component interconnect (PCI).

Both internal and external I/O devices plug into connectors that are soldered to the Mac's logic board. Each port uses a uniquely shaped connector that prevents you from plugging the wrong device into the wrong port. Not only would a device not work if it were plugged into the wrong port, but the wrong voltages going to the wrong connector pins could also harm both the peripheral device and the Mac's circuitry.

Mac ports can be either *direct ports,* which support a single device, or *bus ports,* which connect multiple devices. Direct ports include the modem port, sound port, and the PDS expansion slot. With bus ports, however, only one of the multiple devices can communicate with the CPU at any given moment. All the devices on a bus constantly negotiate with each other to determine which one will do the talking. You can do two things at once—for instance, move your mouse and type on the keyboard—because each device sends data at intervals that are a fraction of a second in length.

To connect several devices, bus ports sometimes use multiple connectors that feed into the same digital circuits inside the Mac, like several on and off ramps on a freeway. The NuBus and PCI expansion slots use multiple connectors. Other bus ports, such as the SCSI port for hard disks, connect multiple devices to one port by daisy-chaining one device to the next. The Apple desktop bus (ADB) for keyboards and mice is one port that can be daisy-chained, which may have more than one connector on a particular Mac model.

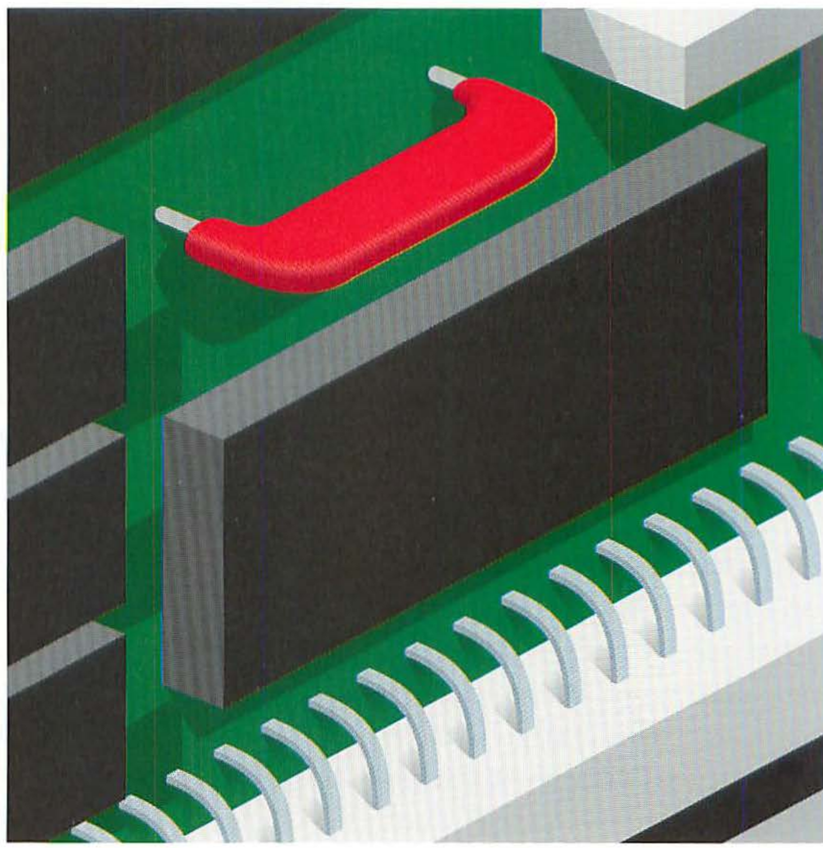
Whether a port is the direct or bus type, most of the Mac's ports move more than one signal at one time through the various pins on the connectors. Some of these lines of communication transmit electrical power to the peripheral device. Others, called *address lines*, provide timing and signaling information, while *data lines* move the information with which you are working. All together, a port can move millions of signals per second across a connector.

To keep track of all the lines of communication, each port has an I/O controller chip that regulates the flow of traffic across a port. The controller chip reads the address lines to determine which peripheral device a signal is coming from or heading to. The port circuitry also translates the various types of signals running back and forth across the connectors into a form the CPU and RAM can read.

All the ports communicate with the CPU and RAM through a master bus called the system bus, a data superhighway on the logic board. A set of managers in the operating system directs the traffic flowing across the system bus and in and out of the Mac. Higher-level system software can then direct the CPU to properly process and display large amounts of complex data, such as multimedia movies and sound.

A key feature of all ports and expansion slots on the Mac is ease of setup. Installing a peripheral device usually involves nothing more than plugging it in and sometimes dragging files to the System folder. Usually there are no DIP switches to set or jumpers to remove. This is because each device carries information about itself. The ports can recognize a device, read this information, and respond appropriately. The ports know which device is in which connector. This means there are no conflicts between devices, which is often the case with Windows-based PCs. The design of the peripheral devices is partly responsible for the Mac's easy setup, but most of the credit goes to the design of the ports and their tight integration with I/O system software.

CHAPTER 19 How SCSI Works



IT was an engineer's sense humor that applied a disreputable epithet to one of the Mac's fastest ports, the *SCSI port* (pronounced *scuzzy*). Short for small computer system interface, SCSI allows you to connect up to seven hard-disk and removable drives, and even some scanners, to a single connector. You do this by simply plugging each new SCSI peripheral into the back of the last peripheral, forming a string of devices called a *SCSI chain.* The SCSI port is a *bus*, a common pathway shared by multiple devices.

Before the release of the Mac Plus in 1986, the only way you could connect a hard disk to a Mac was via a *serial connection*, which sends one bit at a time. SCSI is significantly faster than a serial connection because it is a *parallel connection*, transferring a whole byte (8 bits) at a time. However, SCSI connections require a little more care than serial connections.

A *terminating resistor* is required on the first and last devices on the bus to identify the extent of the bus and to prevent signals from reflecting back on the bus after reaching the last device. A terminator connector is usually plugged into a device's SCSI connector, though some hard drives have internal terminators. The Mac supplies termination for the internal hard-disk drive. Power (5 volts) for the external resistor is supplied by the peripheral.

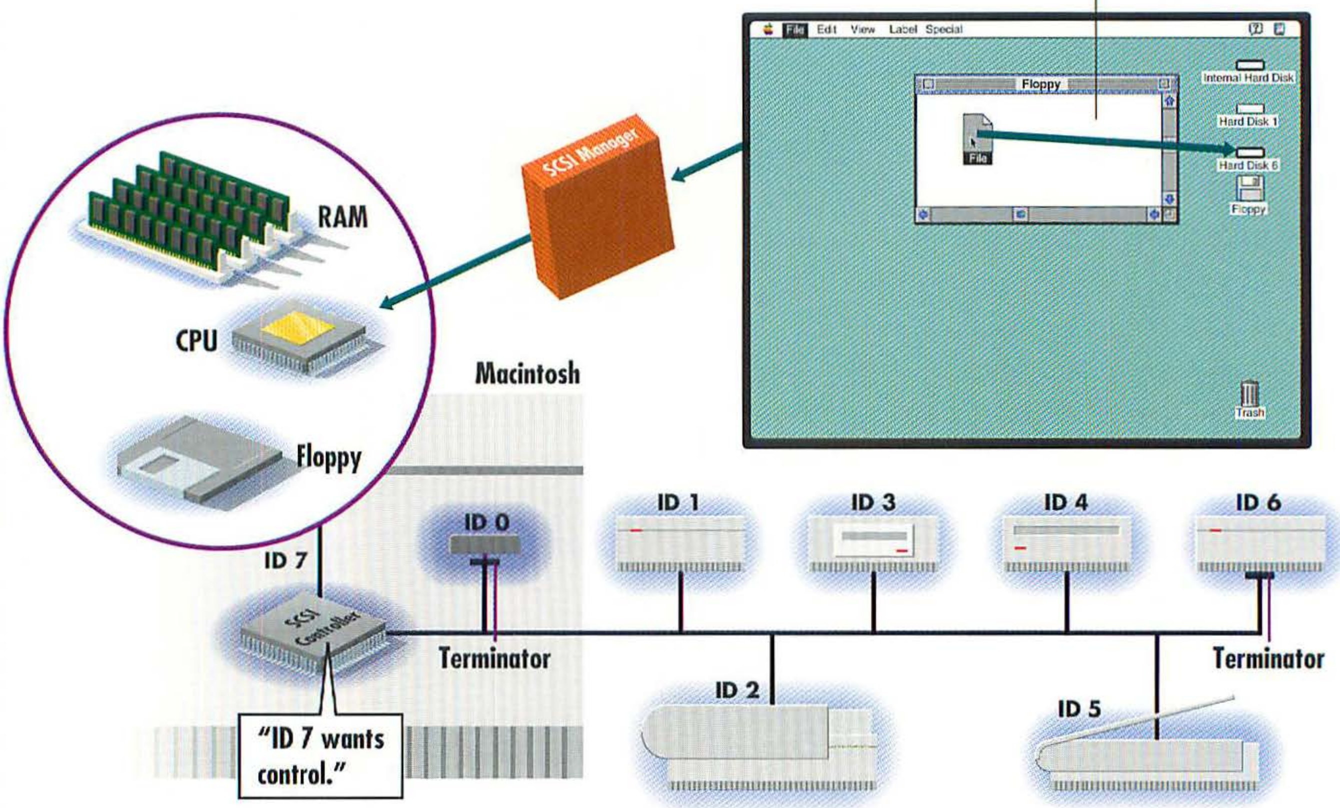
Signals from the Mac are sent out on the SCSI bus by the operating system's SCSI Manager, which commands the SCSI controller chip on the Mac's logic board. Each peripheral has a SCSI driver that is loaded into the Mac's system partition in RAM at startup. With the drivers loaded into the Mac's RAM, the Mac knows which devices to look for on the SCSI bus.

The SCSI Manager now includes compliance with the SCSI 2 standard, which allows for two higher-speed hardware options, Fast SCSI and Wide SCSI. Fast SCSI, used in some Mac models, doubles the data transfer rate from 5MB per second to 10MB. Wide SCSI doubles the width of the data path from 8 bits to 16 bits, thus doubling the speed. Though there are no Mac models with Wide SCSI, you can add it to your Mac with a Wide SCSI expansion card and a drive that supports it.

Some higher-end Mac models have two SCSI buses—one internal and one external. Theoretically, this lets you add up to 14 devices. In these models, the internal SCSI bus is a Fast SCSI bus, and the internal hard disk is on the internal Fast SCSI bus. The internal CD-ROM drive can be on either the internal or the external bus, depending on the Mac model.

SCSI Chain

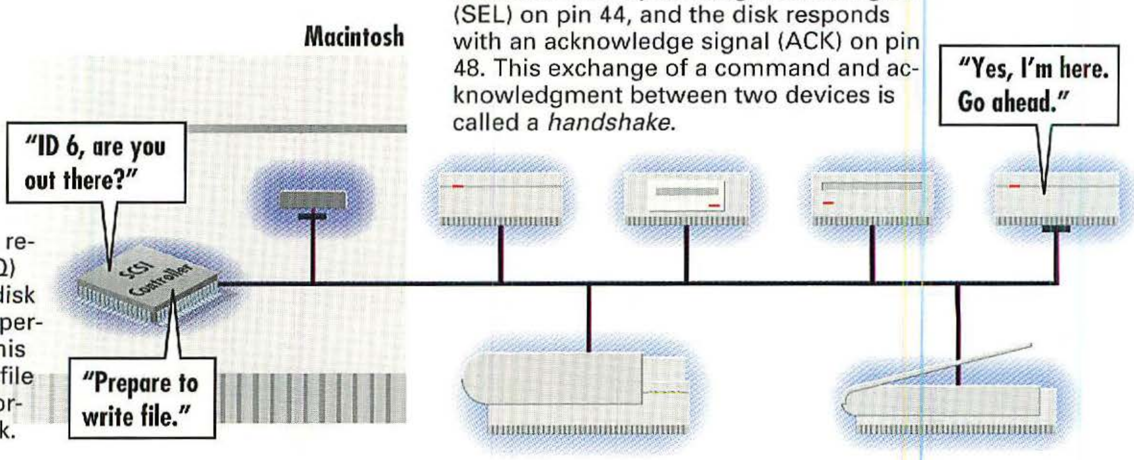
1 When a user drags a file from the floppy drive to a hard disk, the SCSI Manager tells the SCSI controller chip to initiate a SCSI transfer. The SCSI controller treats the Mac logic board—including CPU, RAM, and floppy drive—as a device with SCSI ID 7. An internal hard disk is treated as a separate SCSI device, usually (but not always) with the SCSI ID 0.



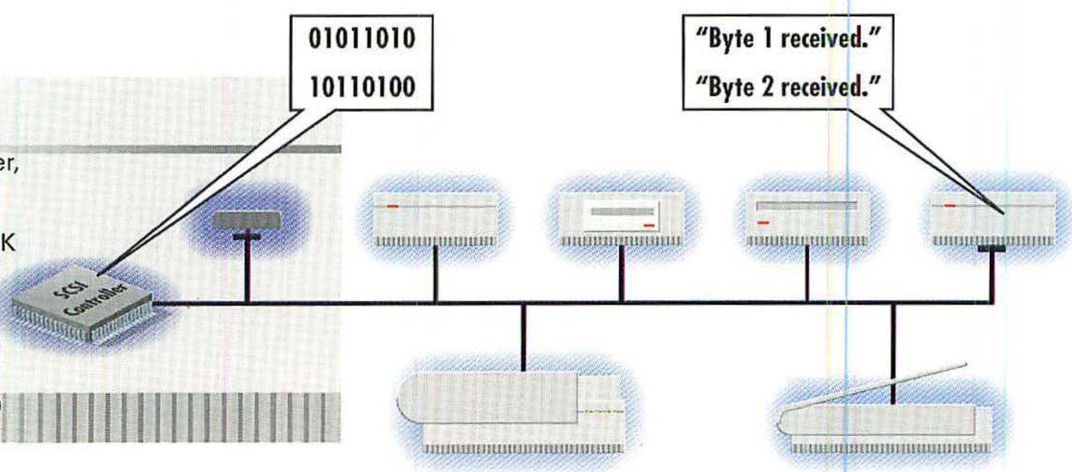
2 Since only one SCSI transaction can occur at a time, the SCSI controller monitors for a free bus. If it detects a transaction occurring, it waits. When it hears nothing, the SCSI controller sends out a signal to gain control of the bus. If no termination signal is detected, it will not be able to see any of the other SCSI devices. If more than one SCSI device seeks control at the same time, the one with the highest number wins, and the SCSI Manager queues the requests from the other devices. Since the Mac has the highest possible ID, it always wins.

MAC FACT SCSI was designed to be a plug-and-play method of adding storage devices to the Mac, but it can be temperamental, particularly when you have multiple devices on the bus. Next to proper termination, cables play the biggest role in setting up a SCSI chain that works. Using cables that are more heavily shielded inside helps ensure signal integrity. It also helps if none of the cables are longer than 3 feet. However, the maximum allowable length for a SCSI chain is 20 feet. This means if you have seven devices, each cable can be no longer than 2.85 feet on average. No matter how many external devices you have, the best cables to use for trouble-free operation are 1.5 feet long.

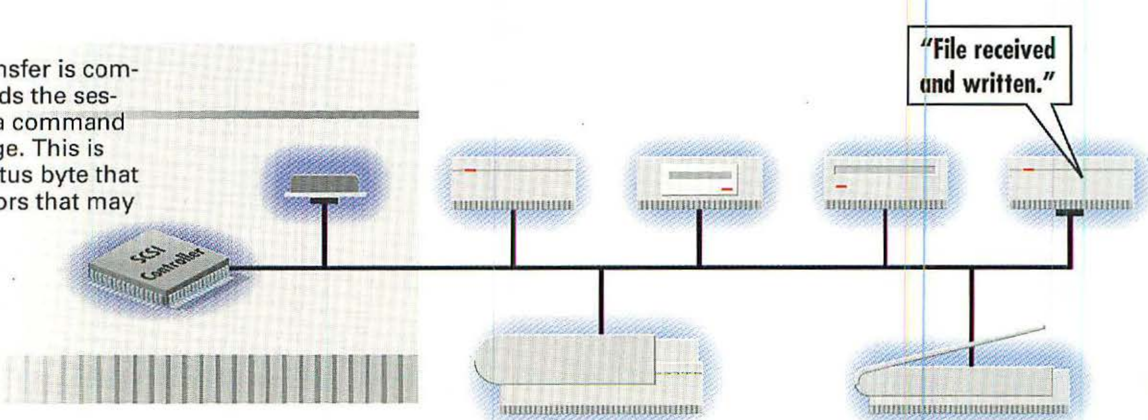
3 The Mac seeks the disk it wishes to communicate with by sending a select signal (SEL) on pin 44, and the disk responds with an acknowledge signal (ACK) on pin 48. This exchange of a command and acknowledgment between two devices is called a *handshake*.



4 The Mac sends a request signal (REQ) on pin 48 to the disk drive telling it to perform a task—in this case, to accept a file and write the information to the disk.



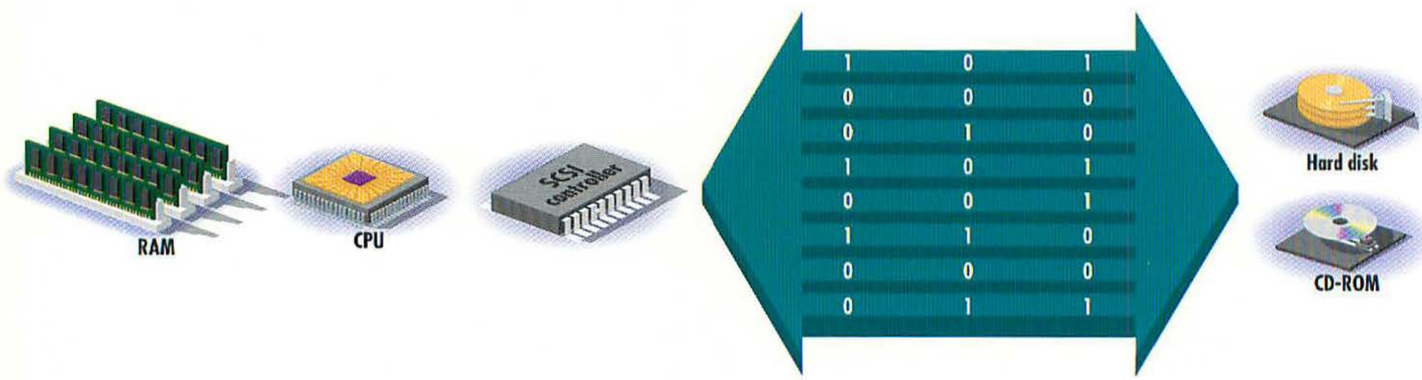
5 The Mac then sends the file over the SCSI bus in an *asynchronous* manner, which means the Mac and disk exchange a handshake with REQ/ACK signals for every byte of data sent. If the data needs to be loaded into RAM, the SCSI Manager moves data directly into RAM, freeing the CPU to do other things until the transfer is complete.



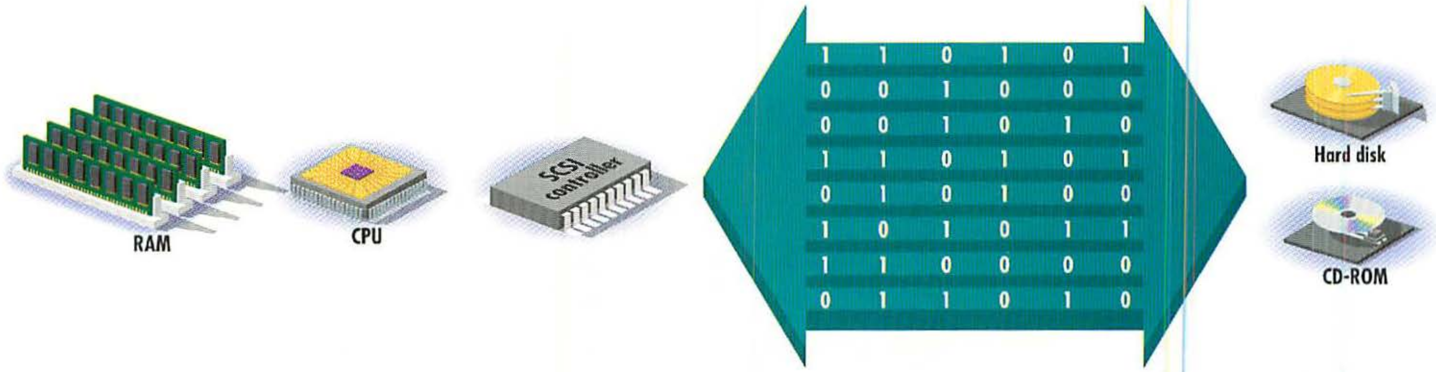
6 When the file transfer is complete, the disk ends the session by sending a command complete message. This is followed by a status byte that indicates any errors that may have occurred.

Fast and Wide SCSI 2

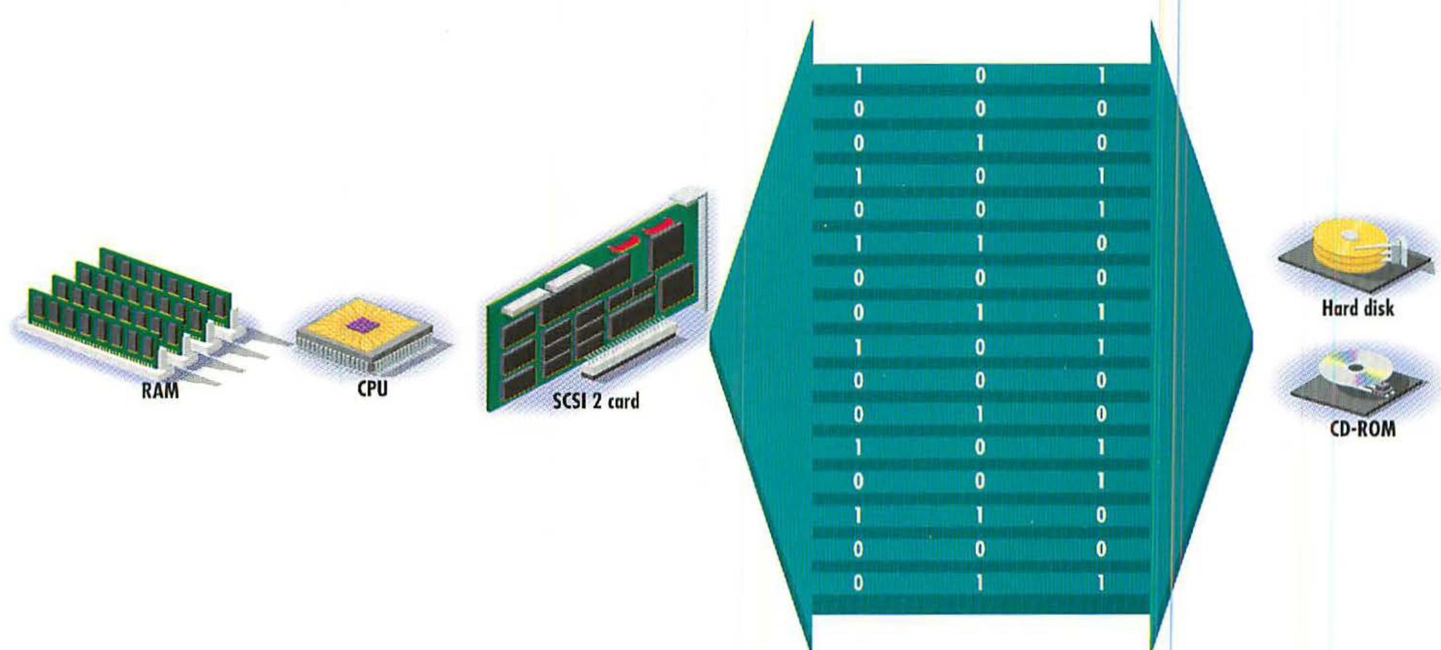
SCSI 2 can speed data transfers with two methods, Fast SCSI 2 and Wide SCSI 2. In either case, the SCSI controller chips in both the Mac and the drives must be SCSI 2 compatible to take advantage of the increased speed. Some devices can use both Fast and Wide SCSI 2 together for greatly accelerated hard-disk access.



Ordinary SCSI SCSI transmits bits of information in a parallel fashion, moving groups of 8 bits (called a byte) between the Mac's RAM and storage drives. The SCSI controllers in the Mac and the disk drives can put a maximum of 5 million of these bytes (5MB) on the SCSI bus every second.



Fast SCSI 2 Fast SCSI 2 controllers can put bytes on the bus at twice the rate of ordinary SCSI, at 10MB per second, with shorter time intervals between each byte. The bytes move just as fast as they do in ordinary SCSI—near the speed of light—but the interval between each byte is shorter. Fast SCSI 2 comes with certain Mac models, including the Power Mac 7500.

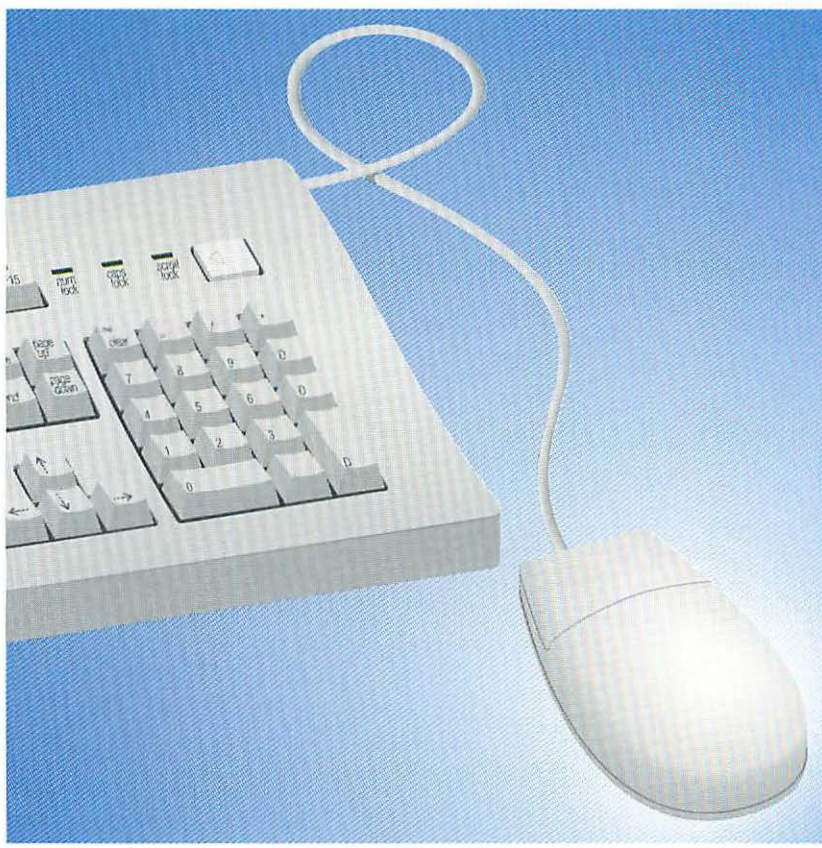


Wide SCSI 2 Since no Mac models currently come with built-in Wide SCSI 2, to take advantage of a Wide SCSI 2 drive you must add an expansion card containing the SCSI 2 controller. Wide SCSI 2 puts data on the bus at the same rate as ordinary SCSI, but doubles the number of bits that are sent at one time to 16. Thus, the number of 8-bit bytes that are moving over the bus is effectively doubled. Future Wide SCSI 2 devices will go to 32 bits. When the expansion card and disk drive support Fast SCSI 2 as well as Wide SCSI 2, the SCSI controllers put 16-bit bytes on the bus at the faster rate.

CHAPTER

20

How the Keyboard, Mouse, and Trackpad Work



BECAUSE they are the primary data-entry and control devices, the keyboard and the mouse are the peripherals you spend the most time with. Since the introduction of the Mac SE in 1987, all Macs have used the Apple desktop bus, or *ADB*, to connect the computer to the mouse and keyboard. The built-in trackballs and trackpads in PowerBook laptops also use ADB. Some Macs have a single ADB port, though most models have two.

ADB offers a big advantage over the keyboard port of PCs and pre-1987 Mac models, in that you can daisy-chain input devices from one to the other, connecting more devices than the number of ports. This lets you plug in multiple keyboards or mice, as well as trackballs and pen tablets. ADB can theoretically support up to 16 devices, but Apple recommends no more than three devices per port for optimal performance.

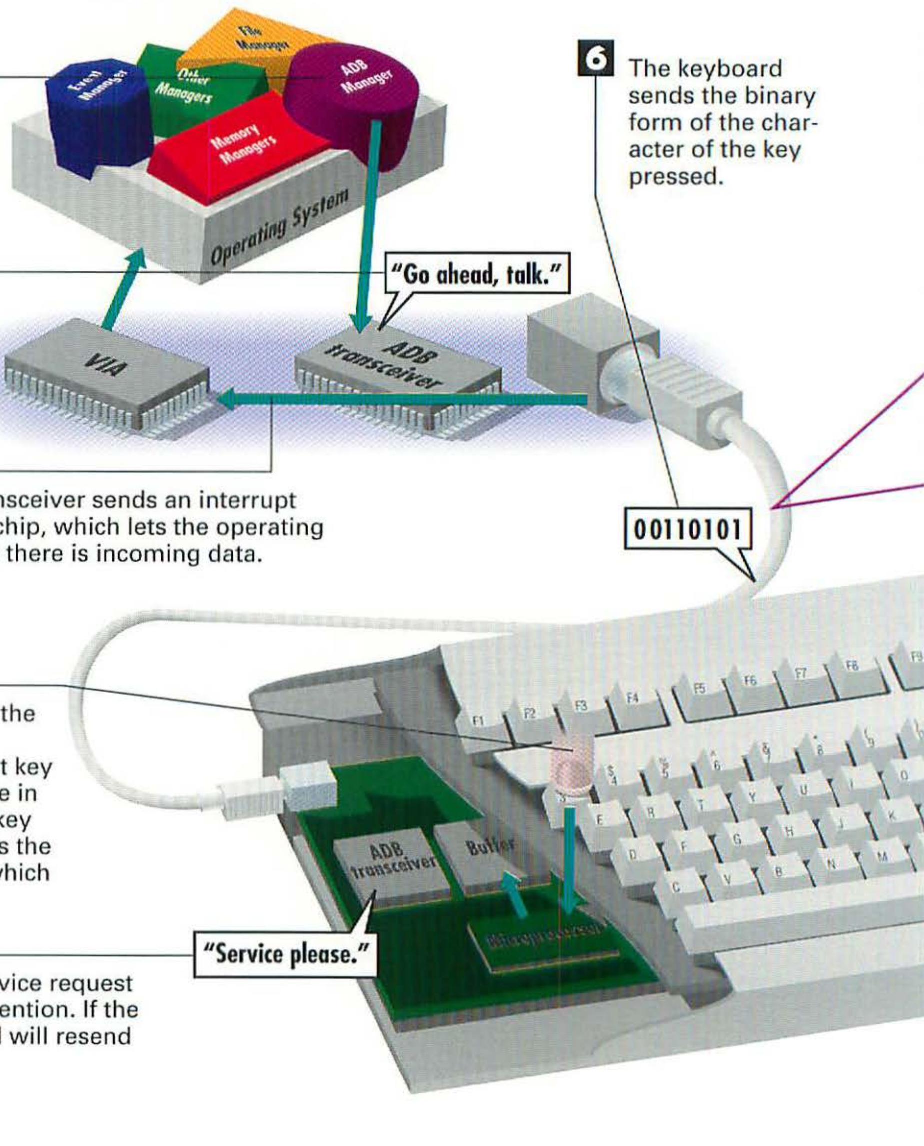
All ADB devices, even those connected to different ADB ports, communicate over a single bus. Each device has a unique identifying address, and messages can be sent to more than one address at the same time. Unlike other buses, all devices on ADB are not created equal. The Mac is the boss and can't be interrupted. If a device wants attention, it must ask the Mac for permission to speak.

Like a local area network, ADB uses a data transmission method called asynchronous serial communications. *Asynchronous* means that bytes of data aren't sent automatically according to a timed sequence, but instead are sent when a response is received from the target device. *Serial* means that each bit in a byte is sent one after the other, instead of at the same time, as in a parallel bus such as SCSI.

The ADB is not a particularly fast port, but the amount of data being transferred is minuscule compared to the data handled by a SCSI bus—the ADB transfers single characters at a time instead of whole pages of text or graphics. What ADB lacks in performance it makes up for in simplicity and ease of use. The connector contains a mere four pins: one to send the ADB signals, one to supply 5 volts to power the ADB devices, a third to act as a ground, and a fourth to enable the user to start up the Mac from the keyboard.

The Apple Desktop Bus

Every Mac has a single Apple desktop bus (even those with two ADB ports), which uses asynchronous communications among all connected devices. The Mac logic board and each input device all contain an ADB transceiver chip, which converts bus signals to something the devices can understand. The Mac also has one or more versatile interface adapter (VIA) chips, which handle most of the trade of information between the CPU and RAM and the ports and storage devices. ADB devices contain a microprocessor and a small amount of memory to hold user commands until they can be transferred to the Mac.



1 When a user presses a key on the keyboard, the key activates a switch under the key, which causes a change in the current flow from that key to the keyboard's microprocessor. The change in current flow tells the microprocessor which key was pressed. The binary code that represents the character is stored in the keyboard's buffer, which can hold five or six characters.

2 The keyboard sends a service request signal to get the Mac's attention. If the Mac is busy, the keyboard will resend the request.

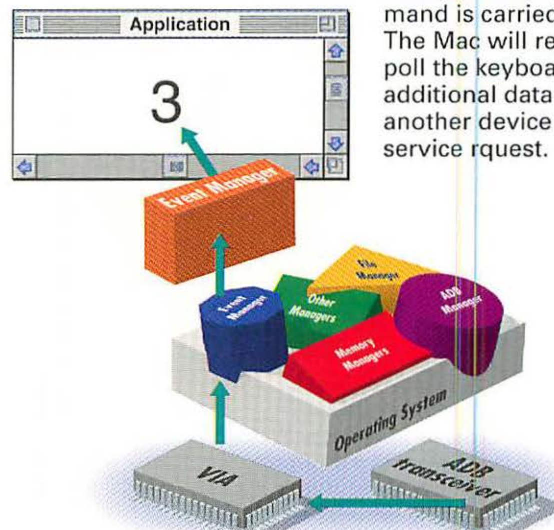
3 The Mac ADB transceiver sends an interrupt signal to the VIA chip, which lets the operating system know that there is incoming data.

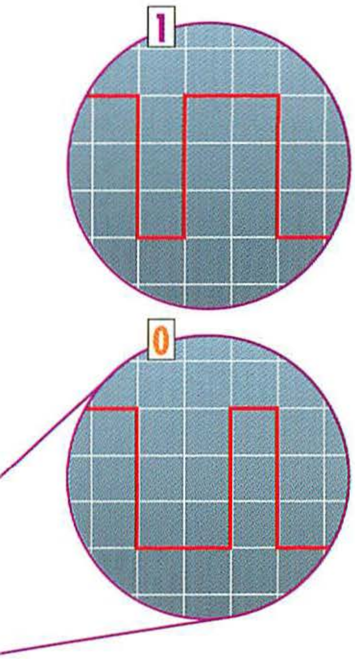
4 The operating system's ADB Manager uses the ADB transceiver to poll each device to see which one is requesting attention. The keyboard responds.

5 The Mac sends the keyboard a talk command.

6 The keyboard sends the binary form of the character of the key pressed.

8 The ADB transceiver in the Mac receives the key-press and passes it to the VIA, which sends a message to the operating system's Event Manager. This message is then passed on to the Toolbox Event Manager. Finally, the key is displayed on screen as a character or, if the user pressed a key combination, a command is carried out. The Mac will regularly poll the keyboard for additional data until another device sends a service rquest.





7 A data bit is sent as a low voltage followed by a high voltage. A 0 is represented by a long low voltage (65 microseconds) followed by a short high voltage. A 1 is represented by a short low voltage (35 microseconds) followed by a long high voltage. Commands also use this scheme, varying the lengths of the voltages in different ways.

MAC FACT Pre-ADB Macs (the original 128K Mac to the Mac Plus) used two separate ports for the mouse and keyboard. The keyboard port used *synchronous* transmission, in which data signals were sent on a timed schedule according to a clock on the logic board. Although users could add a separate numeric keypad to the Mac Plus keyboard, daisy-chaining of devices was prevented by the need to perform the complex task of synchronizing signals from multiple devices. Synchronous communication was abandoned in favor of the asynchronous approach of ADB in the Mac SE and the non-Mac Apple IIGS.

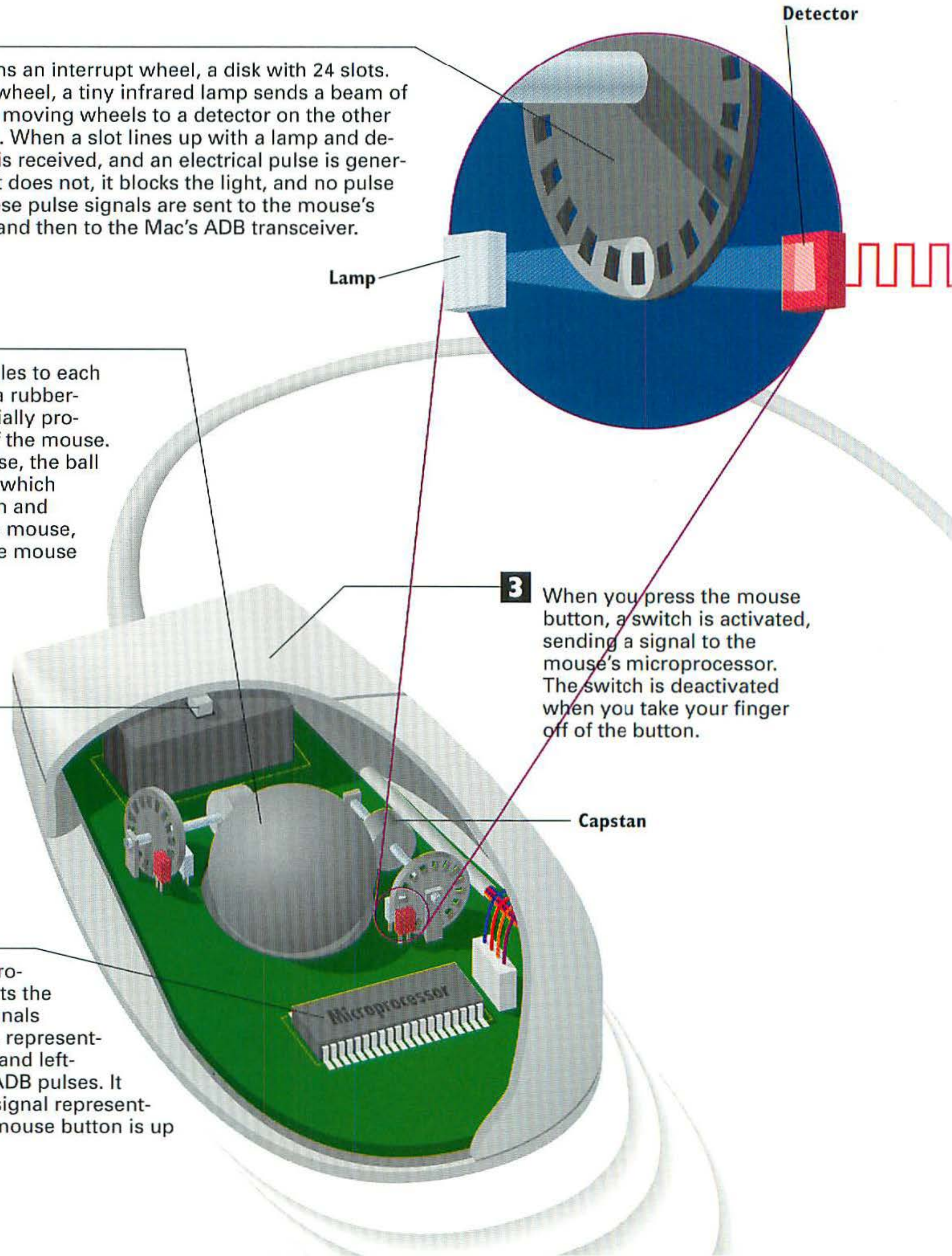
ADB Mouse

1 Two capstans at right angles to each other are in contact with a rubber-coated steel ball that partially protrudes from the bottom of the mouse. When you move the mouse, the ball rotates the two capstans, which measure the up-and-down and side-to-side motion of the mouse, as well as the speed of the mouse in each direction.

2 Each capstan turns an interrupt wheel, a disk with 24 slots. For each slotted wheel, a tiny infrared lamp sends a beam of light through the moving wheels to a detector on the other side of the wheel. When a slot lines up with a lamp and detector, the beam is received, and an electrical pulse is generated. When a slot does not, it blocks the light, and no pulse is generated. These pulse signals are sent to the mouse's microprocessor, and then to the Mac's ADB transceiver.

3 When you press the mouse button, a switch is activated, sending a signal to the mouse's microprocessor. The switch is deactivated when you take your finger off of the button.

4 The mouse's microprocessor converts the lamp/detector signals from each wheel, representing the up-down and left-right motion, to ADB pulses. It also transmits a signal representing whether the mouse button is up or down.

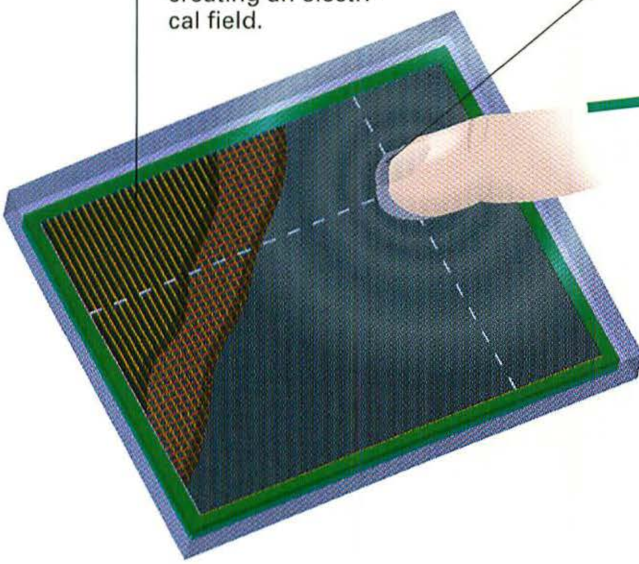


ADB Trackpad

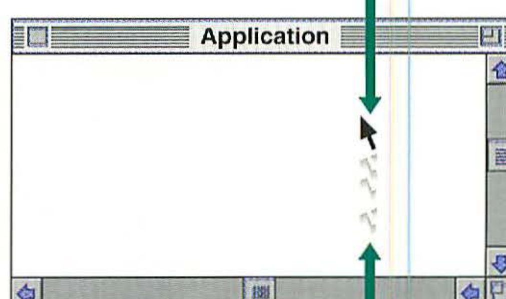
1 A trackball is basically an upside-down mouse, but a trackpad has no moving parts. Under the trackpad's top protective layer, two layers of electrical conductors form a grid. A high-frequency AC current passes through the conductors in the grid, creating an electrical field.

2 Placing your finger on the trackpad disturbs the electrical field, which in turn alters signals going through the conductors. This is called *field distortion sensing* or *capacitance-sensing.*

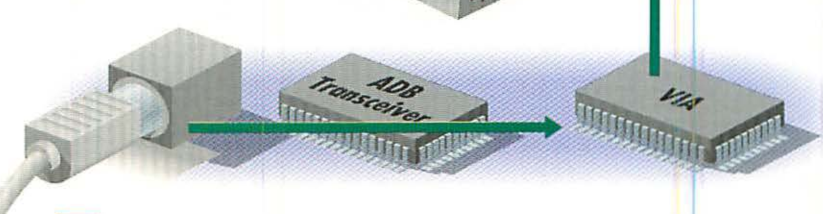
3 A controller chip receiving the altered signals calculates where the center of your finger is, and makes it the control point for the cursor.



4 The controller chip measures how far and fast you move your finger across the grid. On some PowerBook models, the controller will interpret a tap on the trackpad as a mouse click.



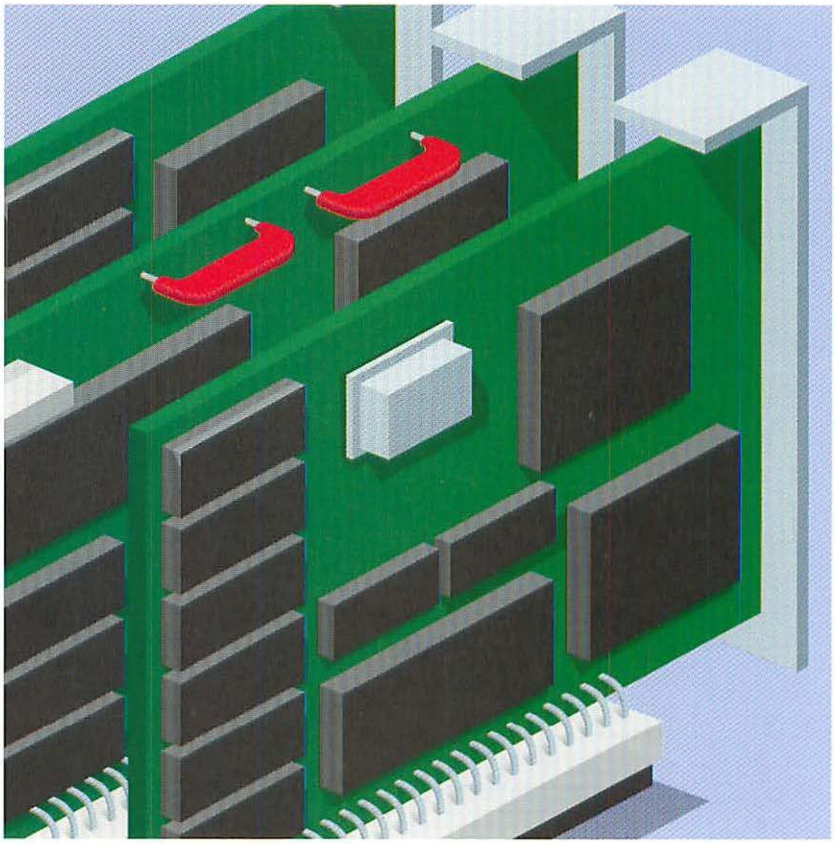
MAC FACT The trackpad's field distortion sensing may seem high tech, but it is similar to the principle behind a 1920s-era electric instrument called the Theremin. This instrument was used for the sound effects in the Beach Boys's "Good Vibrations" and in many 1950s science fiction movies. While the trackpad sends out cursor control signals when you move your finger on it, the Theremin changes pitch when you move your hand next to one antenna, and changes volume when you move toward the other.



5 The Mac's ADB transceiver chip receives signals from a mouse or trackpad and passes them on to the VIA. The operating system and user interface toolbox decode the ADB signals back into numbers for left-right and up-down motion, and move the cursor.

CHAPTER 21

How Expansion Slots Work



SOME Mac users never need to add hardware to their machines. The circuitry for display, sound, networking, and other functions is already built into the logic boards of most Mac models. However, some people may need to tailor the Mac to their own needs by adding new circuitry to export video, acquire scientific data, or even run Windows on their Mac. Fortunately, you don't need to solder new chips to your Mac's logic board to extend its functionality. You can simply plug in an expansion card.

Every Mac contains several different types of *expansion slots*, internal connectors that accept expansion cards. Some slots hold a specific card, such as RAM SIMMs or a modem, while others are general-purpose expansion slots that can accept a wide variety of cards. General-purpose expansion slots provide the expansion card with operating power and with access to the Mac's RAM. Mac expansion cards are self-configuring, designed to be just as easy to use as the rest of the Mac. Just plug them in and they work.

The Mac has three types of general-purpose slot: NuBus, PCI (peripheral component interface), and PDS (processor-direct slot). Most Mac models use at least one of these slot types, though some models contain two types. Although all three types of cards transfer data and address lines a full 32 bits at a time, the three slots operate at different speeds.

NuBus and PCI are *buses*, which means you can have multiple slots in each Mac. NuBus can theoretically handle up to 14 NuBus slots, though no Mac has ever used more than six slots. You can also add an external expansion chassis to get additional NuBus slots. PCI has a theoretical limit of 1,536 slots, more than enough for any computer. Unlike NuBus and PCI, PDS is not a bus, but is a single-slot interface that can have no more than one PDS connector per Mac.

NuBus and PDS were the Mac's first general-purpose expansion slots, making their debut in the 1987 Mac II and Mac SE, respectively. NuBus was invented primarily by Texas Instruments and is specified by an industry standard called IEEE 1196. PCI is a newer expansion slot standard created by Intel and now controlled by the PCI Local Bus Special Interest group, of which Apple is a member. PCI has the advantages of being faster than NuBus and of being widely used in PCs as well as in Macs. PCI is a cross-platform standard, which means you can often use the same PCI card in either a Mac or a PC without modification. (Some cards need special software drivers installed on the Mac or PC.) Because there are so many PCs around, manufacturers can sell PCI cards for less than the equivalent NuBus or PDS card.

NuBus and PCI expansion cards are interchangeable from slot to slot and from Mac to Mac without requiring you to change hardware or software settings. The only limitation is the size of the card: The slots in some Mac models can only fit 7-inch expansion cards, and not the full-sized 12-inch cards.

Once again, PDS differs from NuBus and PCI, and is not standard across all Mac models. Though some Mac models use the same PDSs, many PDS cards only work in a specific Mac model. Macs that use different microprocessors have incompatible PDSs. However, even Macs with the same processor can have different PDSs. This is because unlike NuBus and PCI, PDS is a direct connection to the system bus on the Mac's logic board. This gives a PDS expansion card direct access to the address, data, and control lines of the CPU and RAM.

Although this direct access makes PDS cards incompatible with other PDS slots, it does give them a speed advantage over NuBus and PCI. The processing circuitry in NuBus and PCI adds some delays that PDS technology doesn't encounter. For instance, NuBus and PCI combine the data and address lines that are separate on the Mac logic board and in PDS. In addition, most PDS cards can bypass the Mac's CPU to load information directly into RAM, a trick called direct memory access (DMA) that only higher-end NuBus cards can do.

The extra speed of PDS is the reason Apple included both PDS and NuBus on some Mac models. Other Mac models, such as the Power Mac 6100, contain a PDS that you can convert to a NuBus slot by adding an adapter card. This adapter will let you install a standard NuBus card, but is slower than using a PDS card.

Apple has lessened the need for PDS by adopting the PCI bus, which can approach the speed of PDS, which varies with each Mac. PCI slots can transmit data at a maximum rate of 132MBps, about three times NuBus's 40MBps. In actual operation, the numbers for sustained throughput are smaller, but are still in a 3-to-1 ratio: 90MBps for PCI, 32MBps for NuBus. This is because PCI can send information at a speed of 33 million times per second—33 MHz—three times NuBus's 10 MHz operation. NuBus is also less efficient than PCI. NuBus stops the flow of information to talk to the CPU to arrange the transfer of each block of data (steps 4 and 5 on the next page). PCI arbitrates each new data transfer at the same time that it is moving data, without stopping the flow of data.

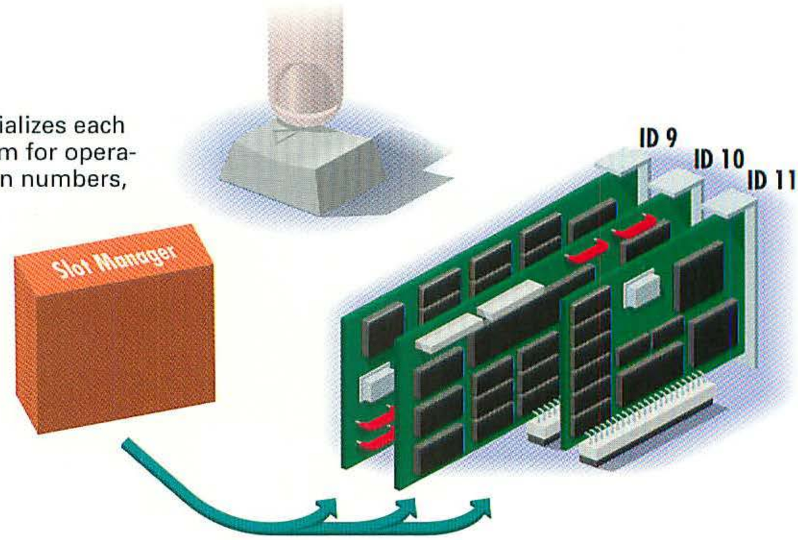
Expansion cards for NuBus, PCI, and PDS look very similar, except for the connector. The NuBus specification calls for a 96-pin DIN (Deutsche Industrie Norm) connector in three rows of 32 pins. The number of pins on PDS cards varies between different versions, usually from 96 to 120, also set in three rows. However, the signals on the pins can be different on PDSs with the same number of pins, making the PDSs incompatible.

PCI cards don't have actual pins, but use 47 pairs of metal "fingers" on either side of the card's edge, like those used on RAM SIMMs. A finger-style pin isn't really a pin at all, but a strip of tin-lead alloy, sometimes plated with a thin layer of gold. These fingers are more prone to wearing out if you frequently remove and replace the card (something you aren't very likely to do). The advantage of the finger-style pins is that you can't accidentally bend them, making the cards easier to install. The finger-style pins are another thing that make PCI expansion cards less expensive than NuBus and PDS cards.

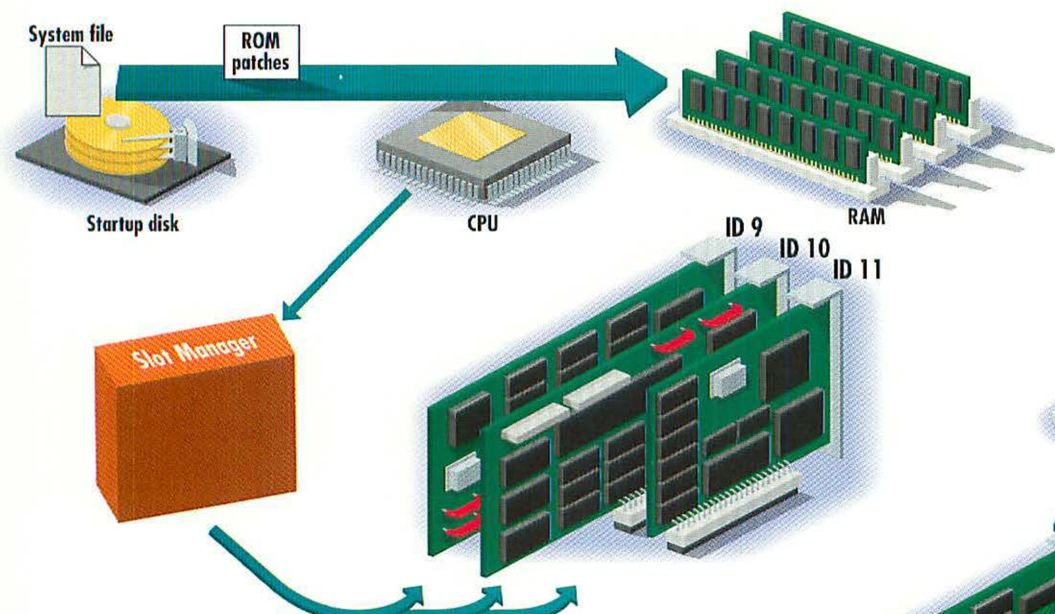
Because of the advantages in speed, price, ease of use, and cross-platform capability, PCI is likely to completely replace NuBus in Macs within the next few years. However, you'll still be able to use old NuBus cards by adding an external NuBus expansion chassis connected to a PCI card in the Mac. But as in the PC world, PCI is likely to become the de facto Mac expansion standard.

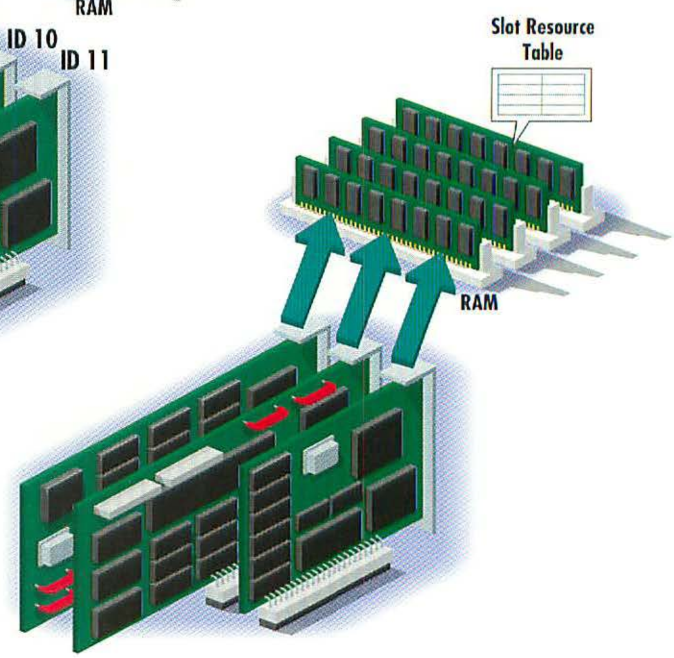
NuBus

1 At startup, the Slot Manager initializes each NuBus card, which prepares them for operation and gives them identification numbers, usually between 9 and 14.



2 The Slot Manager reinitializes the cards after ROM patches (updated operating system code) are loaded from the System file into RAM. This is in case some of the cards require the ROM patches for certain operations.

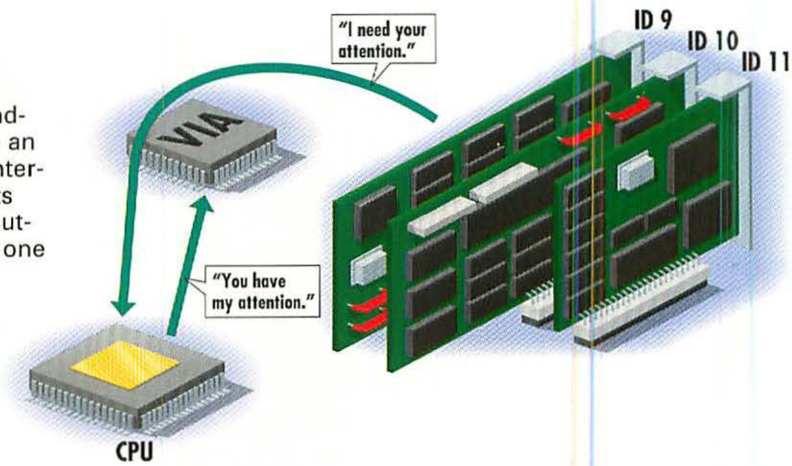


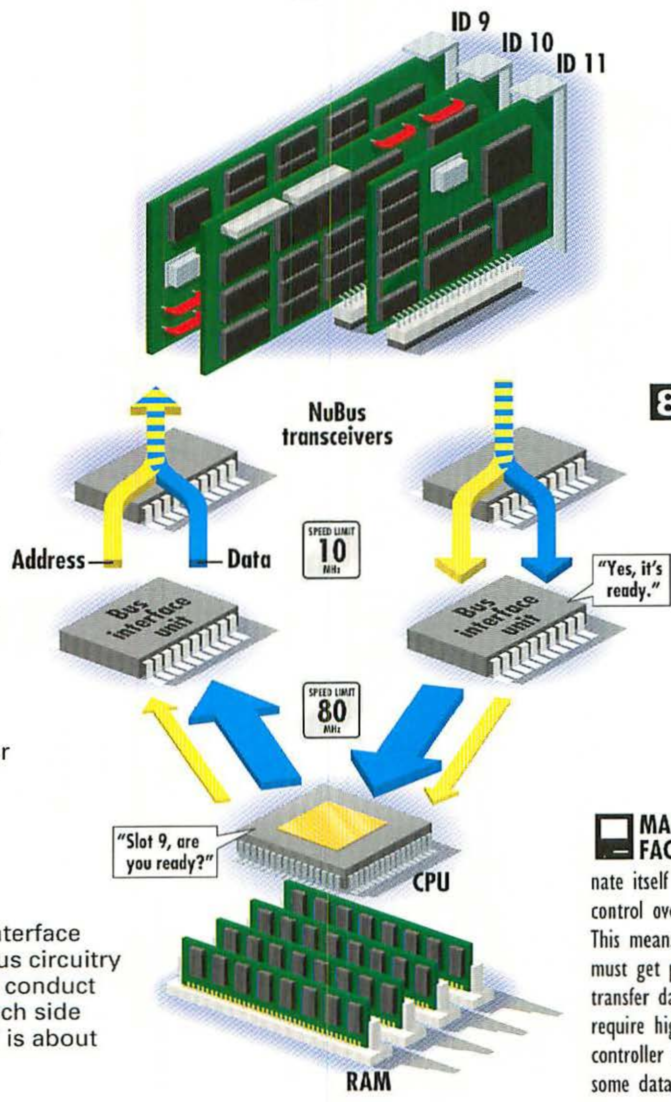


3 In each card's ROM, the Slot Manager reads code called *sResources,* which is software that can be used by the system and by applications to work with the card. The Slot Manager creates the Slot Resource Table (a list of all sResources available), which is stored in RAM. Applications then contact the Slot Manager to access these routines.

4 When a card wants to send or receive data, it gets the CPU's attention by sending it an interrupt signal. Each slot has an interrupt line that goes to a versatile interface adapter (VIA) chip, which connects the CPU and RAM to the Mac's input/output circuitry. (Some Mac models have one VIA chip, others have two.)

5 The CPU checks with the VIA to find out which card wants attention.





6 The CPU and the bus interface units (BIUs) in the NuBus circuitry on the Mac logic board conduct handshaking, letting each side know that data transfer is about to be begin.

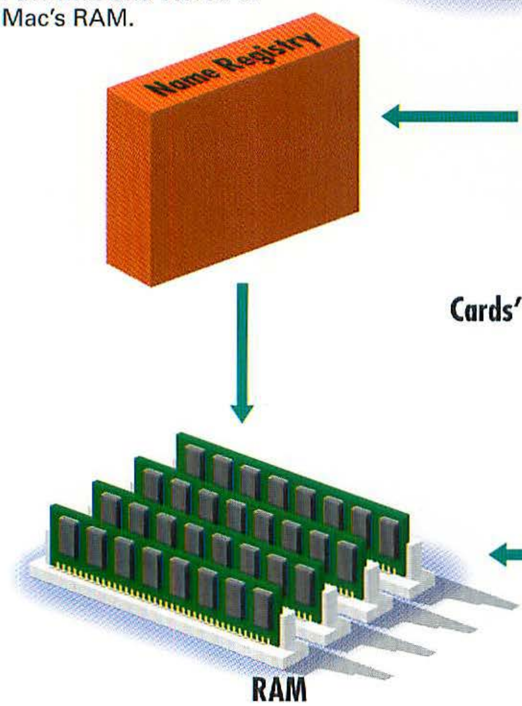
7 When data transfer begins, the BIUs synchronize the flow of data between the NuBus circuitry, which moves data at 10 million cycles per second, and the rest of the logic board, which moves data asynchronously at a faster rate. One BIU sends data to the cards, while another sends data toward the CPU.

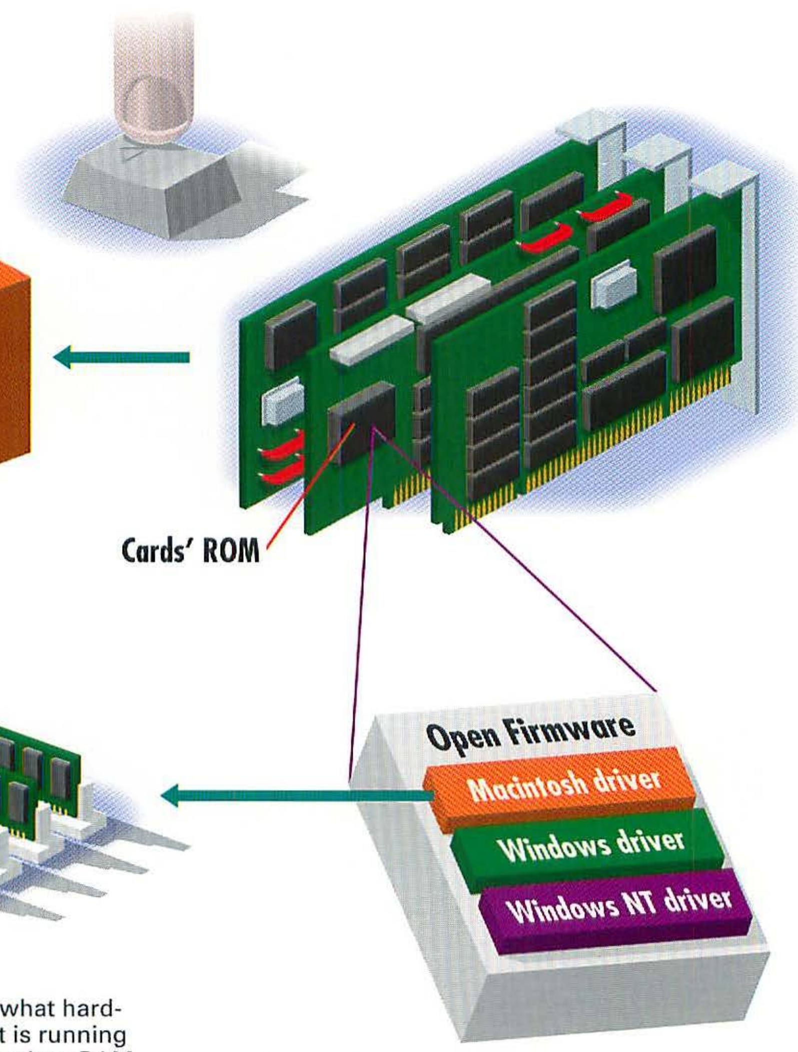
8 NuBus transceivers on the Mac logic board buffer signals until they are ready to be sent, and then they *multiplex*, or combine, the address and data information received from the logic board by weaving them together. The transceivers also *demultiplex*, or separate, the information received from the NuBus cards into separate address and data lines.

MAC FACT The Mac allows a NuBus card to designate itself as *bus master*, which gives the card total control over the flow of information in the NuBus. This means that other cards—as well as the Mac—must get permission from the bus master card to transfer data. Bus masters are built into cards that require high-speed data transfer, such as specialized controller cards for high-speed hard-disk drives and some data acquisition cards.

PCI Bus

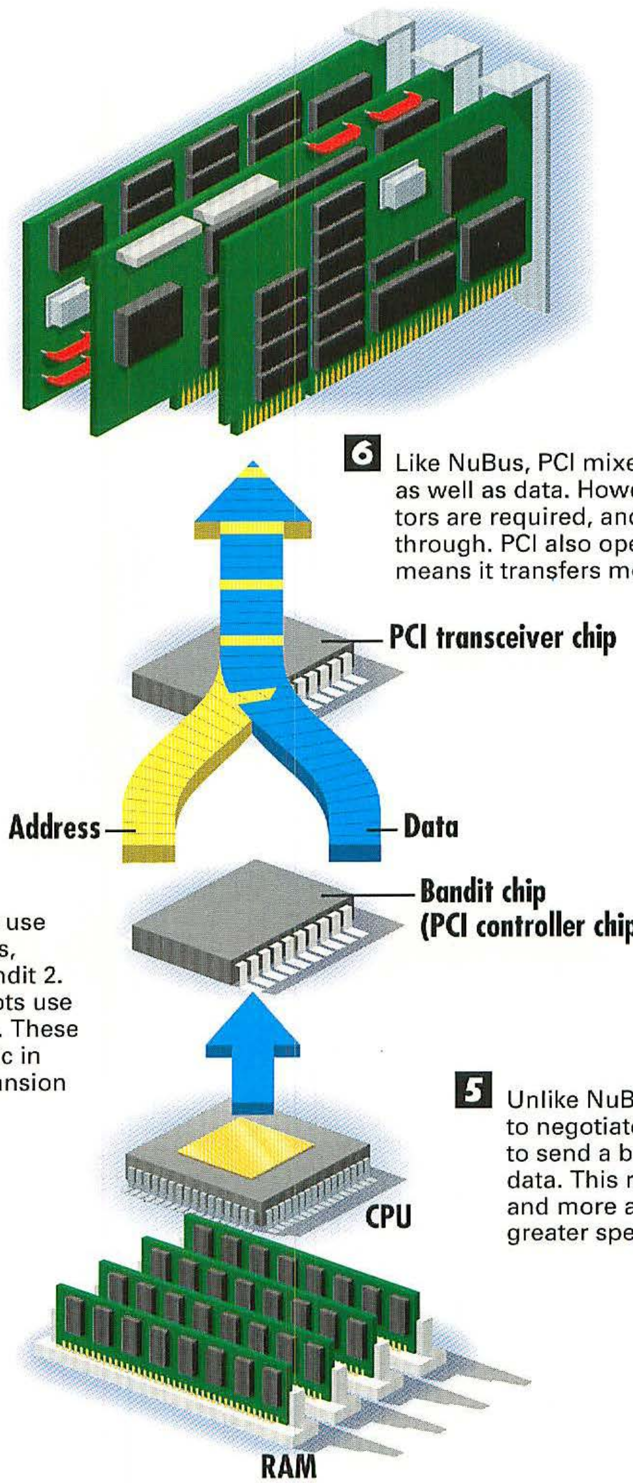
1 At startup, before the operating system loads, code called Open Firmware in the PCI cards' ROM contacts code called the Name Registry in the Mac. (PCI does not use NuBus's Slot Manager.) The configuration and the location of the PCI cards are registered with the Mac and stored in the Mac's RAM.





2 Open Firmware looks to see what hardware and operating system it is running in. It then loads the Mac driver into RAM. On a Windows machine, the same card would load a Windows driver into the PC's RAM. Open Firmware is an industry standard, IEEE 1275.

3 Cards that don't need to be active at startup, such as network cards, don't need a driver in the card's ROM. These cards can use a driver that takes the form of an extension file in the System folder, which is loaded later in the startup process.



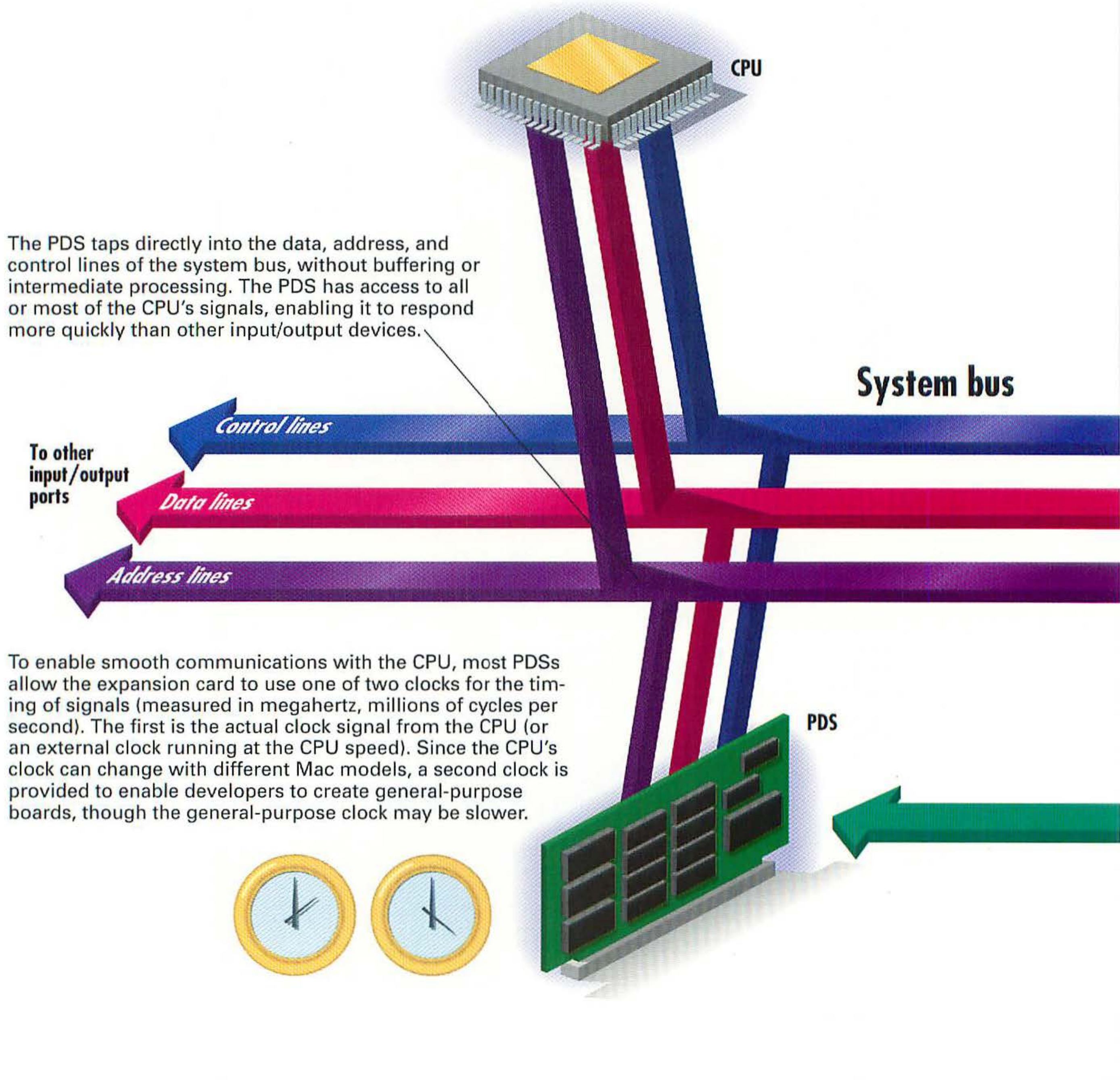
6 Like NuBus, PCI mixes address and locations, as well as data. However, fewer address indicators are required, and more data can get through. PCI also operates at 33 MHz, which means it transfers more 1's and 0's than NuBus.

4 Macs with six PCI slots use two PCI controller chips, called Bandit 1 and Bandit 2. Macs with three PCI slots use one PCI controller chip. These chips manage the traffic in and out of the PCI expansion cards.

5 Unlike NuBus, the PCI only has to negotiate once with the CPU to send a burst of many pieces of data. This results in less talking and more action, and therefore greater speed.

Processor Direct Slot (PDS)

Modern Macs use 64 of the 100 or so PDS pins for 32 bit data and 32 bit addressing (1 bit per pin). Several pins are used to supply power and ground to the card. The remaining pins are connected to signal lines, which transfer control information.

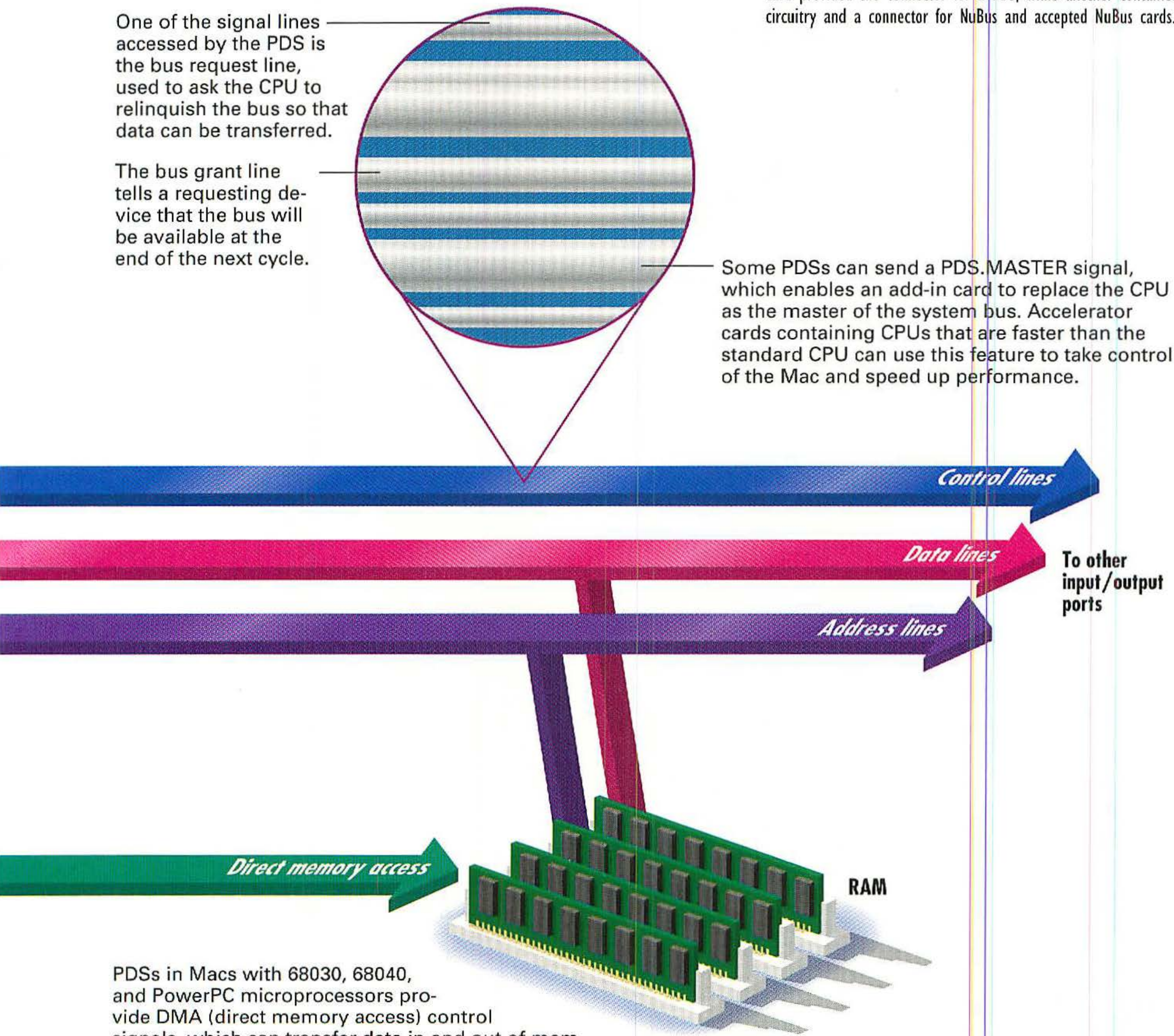


MAC FACT The now-discontinued Mac IIsi had an expansion slot that was something of a cross-dresser. Whereas the lower-end Mac LC had a PDS, and the higher-end IIci used NuBus, the Mac IIsi had a single slot that could act as either a PDS or a NuBus slot. The slot itself was neither; you needed to add an intermediate connector card to provide the actual slot. One type of intermediate card provided the connector for a PDS, while another contained circuitry and a connector for NuBus and accepted NuBus cards.

One of the signal lines accessed by the PDS is the bus request line, used to ask the CPU to relinquish the bus so that data can be transferred.

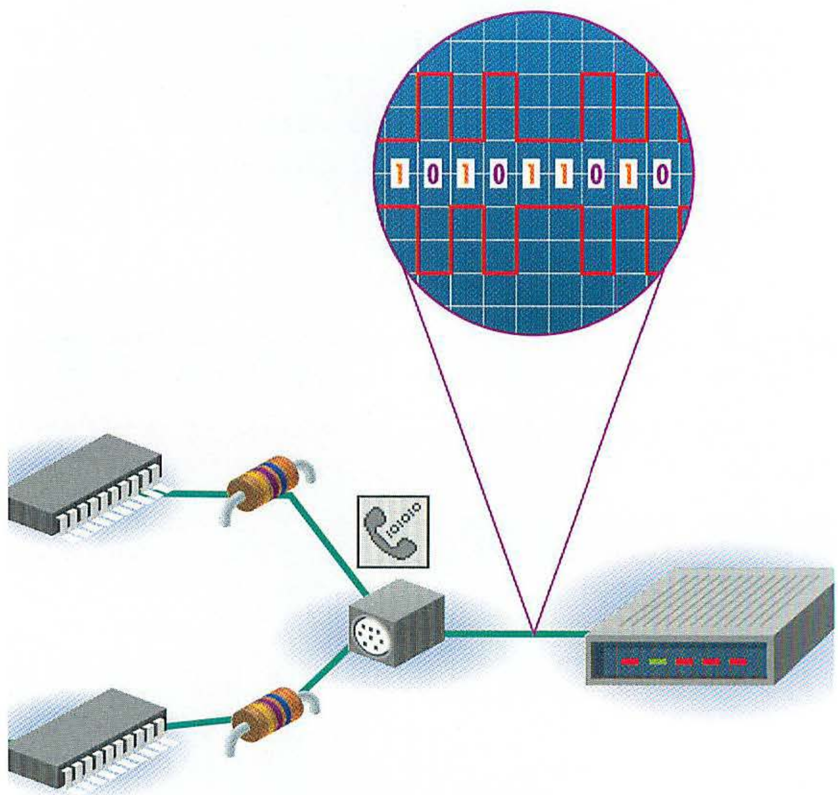
The bus grant line tells a requesting device that the bus will be available at the end of the next cycle.

Some PDSs can send a PDS.MASTER signal, which enables an add-in card to replace the CPU as the master of the system bus. Accelerator cards containing CPUs that are faster than the standard CPU can use this feature to take control of the Mac and speed up performance.



PDSs in Macs with 68030, 68040, and PowerPC microprocessors provide DMA (direct memory access) control signals, which can transfer data in and out of memory without using the CPU. This feature is useful for cards that move a lot of information quickly, such as graphics accelerators and coprocessor cards.

CHAPTER 22 How Serial Ports Work



SINCE before the first MAC, computers have used serial ports to add input/output devices. Serial ports transmit data the old-fashioned way, one bit at a time. Despite the proliferation of high-speed ports, serial ports are still fast enough to remain a mainstream method of moving data in and out of computers. Over the years, Apple has taught the old port some new tricks.

The two serial ports in every Mac, the printer and modem ports, enable the Mac to talk to the world outside your office through a modem or a local area network. They can also run desktop devices such as inexpensive printers, digital cameras, and MIDI-capable musical instruments. You can even make a direct connection to another Mac or a PC using serial ports.

The printer and modem ports are almost identical, and most modems, printers, and other devices can be run on either port. Typically, the port you use is set in the software you are using.

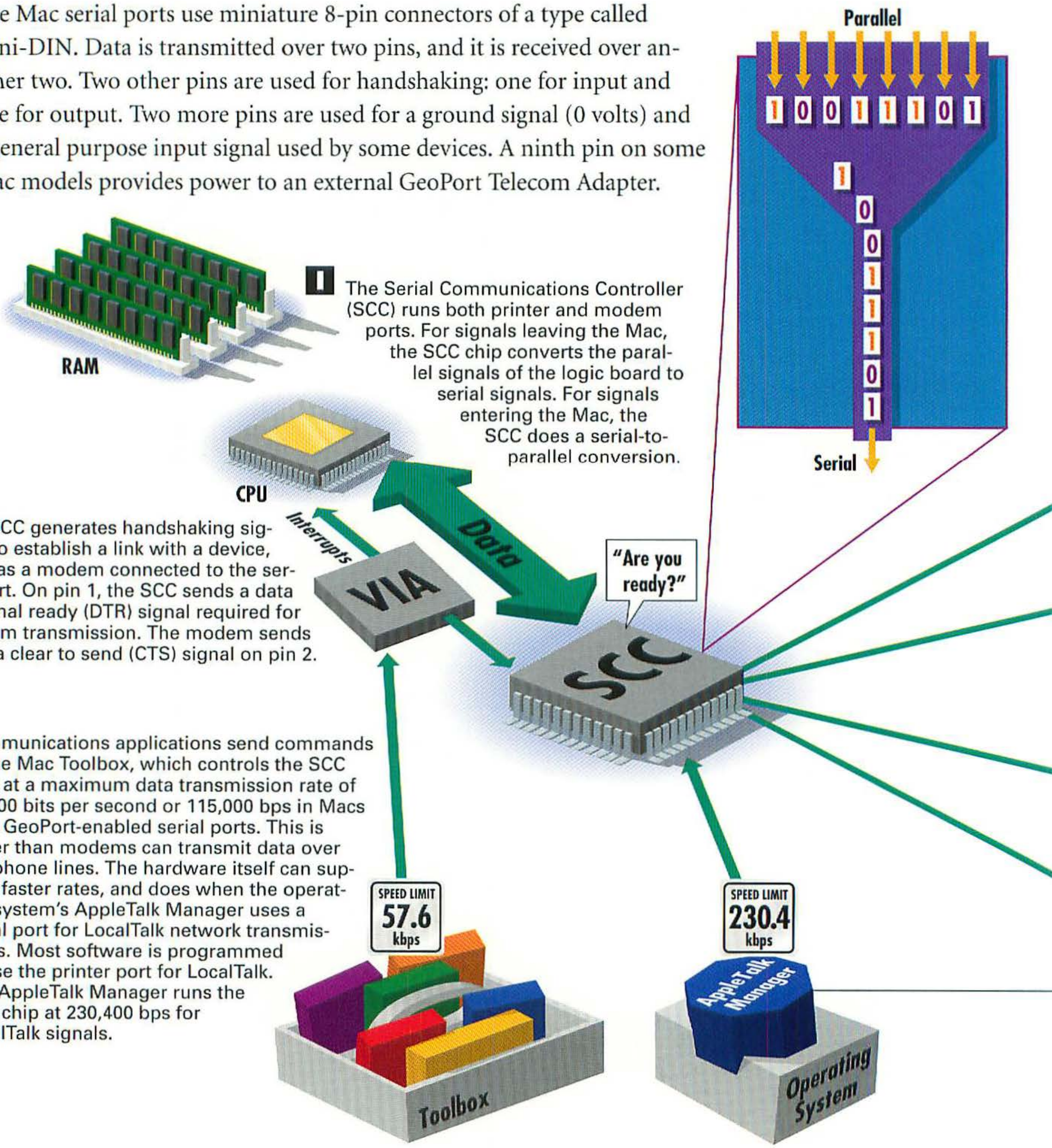
Both ports are controlled by the same chip, the Serial Communications Controller (SCC), and follow the RS-422 standard for data transmission. RS-422 is a newer and somewhat better transmission method than the RS-232 serial ports used in IBM-compatible PCs, in that signals can travel for longer distances and are less affected by line noise.

A serial port, usually the printer port, can also be used to connect to LocalTalk local area networks, which give you access to laser printers and other network communications. In LocalTalk mode, a serial port is taken over by the operating system's AppleTalk Manager, which speeds up the maximum transmission rate by a factor of four. This results in printer performance similar to that of parallel ports on IBM-compatible PCs, but at distances up to about a hundred times greater.

GeoPort is another trick Apple uses to speed up the serial port. GeoPort-enabled serial ports can operate at LocalTalk speeds and beyond because of a trick called direct memory access (DMA), which can bypass the CPU to directly access RAM. Some 68040 AV Macs and most Power Macs have GeoPort-enabled serial ports, though the Performa 5200 and 6200 do not. You can use GeoPort-enabled serial ports to connect ordinary serial devices and to use GeoPort devices, such as Apple's GeoPort Telecom Connector, Apple's software-only Express Modem, and Apple's QuickTake digital camera. In the future, GeoPort serial ports may be used for connecting to high-speed ISDN (integrated service digital network) lines.

Modem and Printer Ports

The Mac serial ports use miniature 8-pin connectors of a type called mini-DIN. Data is transmitted over two pins, and it is received over another two. Two other pins are used for handshaking: one for input and one for output. Two more pins are used for a ground signal (0 volts) and a general purpose input signal used by some devices. A ninth pin on some Mac models provides power to an external GeoPort Telecom Adapter.



1 The Serial Communications Controller (SCC) runs both printer and modem ports. For signals leaving the Mac, the SCC chip converts the parallel signals of the logic board to serial signals. For signals entering the Mac, the SCC does a serial-to-parallel conversion.

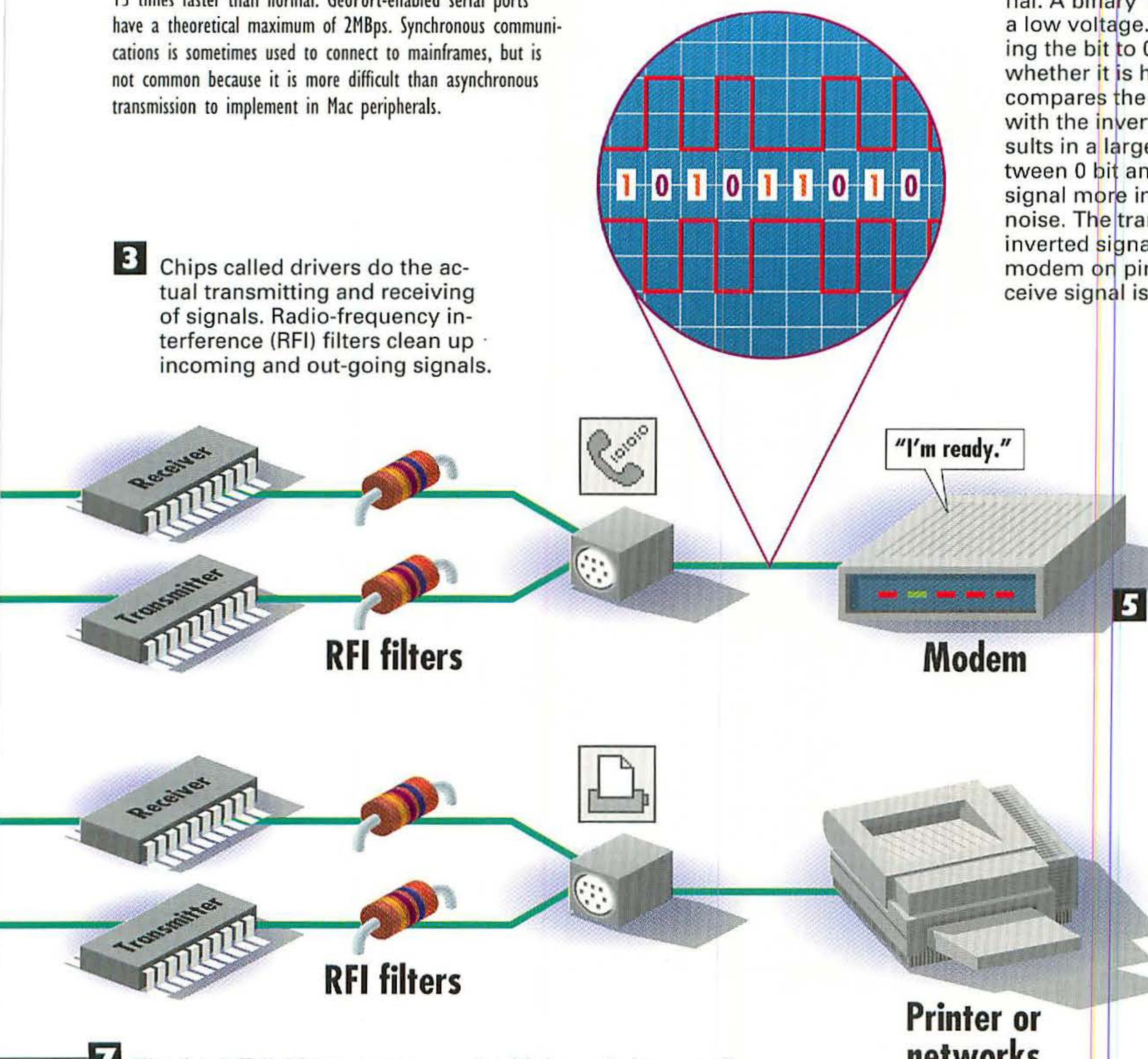
2 The SCC generates handshaking signals to establish a link with a device, such as a modem connected to the serial port. On pin 1, the SCC sends a data terminal ready (DTR) signal required for modem transmission. The modem sends back a clear to send (CTS) signal on pin 2.

6 Communications applications send commands to the Mac Toolbox, which controls the SCC chip at a maximum data transmission rate of 57,600 bits per second or 115,000 bps in Macs with GeoPort-enabled serial ports. This is faster than modems can transmit data over telephone lines. The hardware itself can support faster rates, and does when the operating system's AppleTalk Manager uses a serial port for LocalTalk network transmissions. Most software is programmed to use the printer port for LocalTalk. The AppleTalk Manager runs the SCC chip at 230,400 bps for LocalTalk signals.

MAC FACT Most of the time, your modem port runs in asynchronous mode, which requires the Mac and the peripheral to frequently exchange handshaking signals. The modem port also supports synchronous transmission of data, which doesn't use handshaking. Synchronous mode can transmit data at a theoretical maximum of 900,000 bits per second, some 15 times faster than normal. GeoPort-enabled serial ports have a theoretical maximum of 2MBps. Synchronous communications is sometimes used to connect to mainframes, but is not common because it is more difficult than asynchronous transmission to implement in Mac peripherals.

3 Chips called drivers do the actual transmitting and receiving of signals. Radio-frequency interference (RFI) filters clean up incoming and out-going signals.

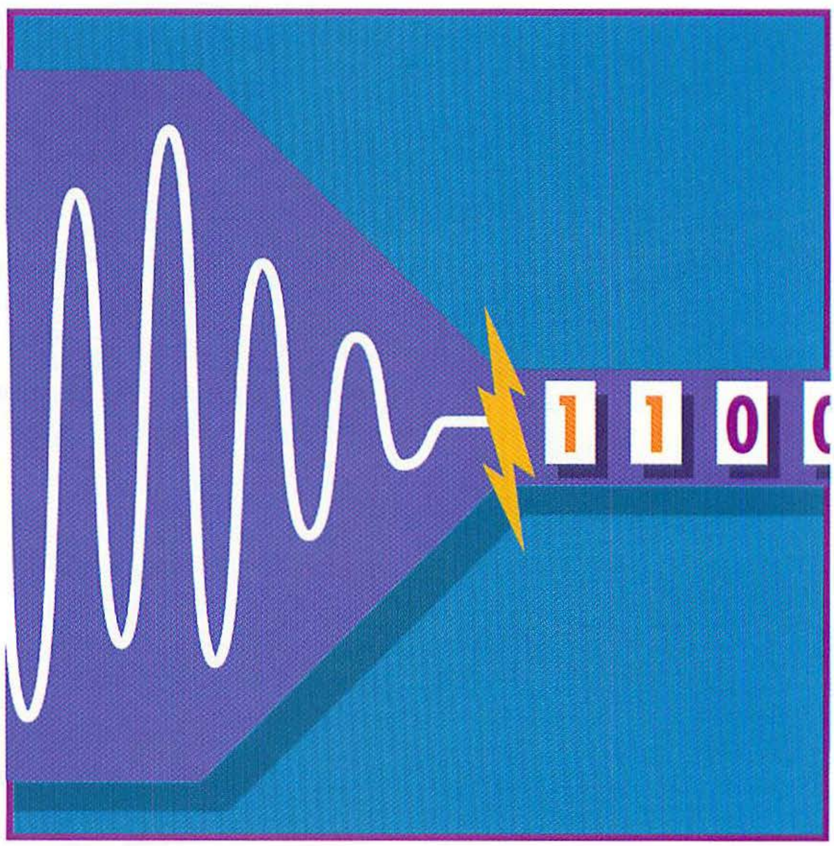
4 Digital data is sent to the modem with an inverted copy of the signal. A binary 1 is represented as a low voltage. Instead of comparing the bit to 0 volts to determine whether it is high or low, RS-422 compares the voltage of the bit with the inverted signal. This results in a large difference between 0 bit and 1 bit, making the signal more immune to line noise. The transmit signal and its inverted signal are sent to the modem on pins 6 and 3; the receive signal is on pins 8 and 5.



5 For incoming data, the modem sends the SCC an interrupt signal to get its attention. Inside the SCC, the modem port has a higher interrupt priority than the printer port, making it the better port for high-speed communications.

7 The AppleTalk Manager can run at a higher rate because it takes complete control of the serial circuitry, and it doesn't require interrupt signals to the SCC to receive and transmit data. For instance, if an AppleTalk message comes in when a floppy-disk drive is running, the VIA (versatile interface adapter) will interrupt the drive to send data to the CPU or RAM. You may also notice a slowdown in an application when a network file transfer is occurring in the background.

CHAPTER 23 How Multimedia Works



Of all the computer terms in this book, *multimedia* has to be one of the least descriptive, most overused, and most misunderstood of them all. It brings to mind people creating video games and CD-ROM encyclopedias on high-end, expensive computers. In actuality, multimedia is a fact of everyday computing.

Multimedia is all about bringing together different communications media: sound, video, still graphics, animation, 3D images, and text. The Mac has always been an integrated multimedia computer. Its built-in hardware and software work together to create and present multimedia in a manner that doesn't require your knowledge of every component. Even the earliest Macs let you easily combine three media—text, graphics, and sound—at a time when most computers were text-only machines.

Today, there are a lot more media in multimedia. These include audio CDs, stereo systems, musical instruments, CD-ROMs, analog VCRs and video cameras, digital cameras, microphones, cable television, digital files created on your Mac, and even the Internet. Of course, not every Mac model contains all the hardware needed to connect to every source of multimedia. But every Mac does have a core set of multimedia hardware as well as the software infrastructure to support any of the added hardware and multimedia applications you decide to use. The Mac incorporates multimedia throughout.

Consider that sound is everywhere in a Mac. Every Mac ever produced can play sophisticated sounds through built-in speakers. Many models have sound output and input ports, and some come with built-in microphones. On recent models, these sound ports can play or record 16-bit, 22 KHz per channel "CD-quality" sound. The SimpleText text editor can actually speak to you, reading aloud any text document, and the Mac's Movie Player utility can play sound files and MIDI files, as well as video movie clips.

The Mac works with sound data in both analog form (constantly fluctuating voltages) and digital form (1's and 0's). The sound ports move analog sound data, like the signals coming out the back of your home stereo receiver. When you capture a sound and turn it into a file with the Sound or Simple Sound control panel, you are *digitizing* it, turning the analog data into a digital Mac file.

You can import and export digital sound signals in a variety of ways, including through a MIDI device connected to a serial port. *MIDI* (short for musical instrument data interface) enables Mac software to control and record music from electronic instruments such as keyboards and synthesizers. Composers use MIDI to have Mac software automatically write a score as they play an instrument, or to lay down multiple tracks. Musicians often have Macs onstage with them during performances to play some of the things they've put together on the Mac.

Just as you can work with analog or digital sound, you can use Macs to create and view digital and analog video. You play digitized video when you view movie clips on CD-ROMs. You view analog video when you connect a VCR or video camera to a Mac's video input port and use the Apple Video Player desk accessory to watch it. You digitize this video when you capture it with the Apple Video Player desk accessory or other software. You can capture still images or video clips to use in a presentation, or for output to a VCR. Some Macs (such as the certain Quadra 630s and Performa 5200s) have a television tuner card: Plug in cable TV and watch the news in a window on your Mac screen.

Macs that have a built-in video digitizing port include the so-called AV Macs, such as the Quadra 840A; certain Power Macs, such as the 7500; and the 8500, which also has a video-out port. If you don't have a video input or output port, you can add one in the form of a card from Apple or other companies. Video professionals often add expansion cards that provide higher quality and speed than the Mac's built-in hardware. These people also use more sophisticated software than Simple Sound or Apple Video player. However, the most powerful Mac multimedia software is never seen by anyone.

QuickTime is the king of Mac multimedia system software. This behind-the-scenes system extension gives Mac applications the ability to use digitized audio, video, graphics, and animation files created by a variety of software. QuickTime provides a standard way for applications to handle all of these multimedia elements. You may think your software has great multimedia features, but it's all being handled by QuickTime.

For instance, it is QuickTime that enables you to view video movies in a word processor or to look at still PhotoCD images in SimpleText. QuickTime lets you double-click on PC sound files in .WAV and .AU formats, and automatically converts them to formats your software can read. QuickTime gives you the power to easily integrate different multimedia elements in a document by letting you cut, copy, and paste movies and sounds between documents of different applications.

QuickTime also provides a standard file format for sharing different types of multimedia. This is called the *Movie format*. Files written in the Movie format can contain video clips with synchronized sound, animation, or just sounds. An application supporting the Movie format can play each of these types of multimedia. Some applications, such as Adobe Premiere or DiVA VideoShop, can edit QuickTime Movie files. In these programs, you add special effects such as visual transitions, and set the start and stop points of sounds in a movie.

QuickTime also contains standard MIDI sounds, and can convert synthesized MIDI music files into Movie format. This enables QuickTime to automatically play sounds through electronic musical instruments connected to the Mac through a MIDI interface connector.

Part of what makes QuickTime successful is its ability to compress and decompress multimedia data. Video and sound files are huge, and would quickly fill a hard-disk drive if it weren't for compression. When a Movie file plays, QuickTime decompresses it on the fly. QuickTime uses several Apple and industry standard compression/decompression algorithms (CODECs), including JPEG (Joint Photographic Experts Group) for photos and MPEG (Motion Picture Experts Group) for video, which you can use if your Mac has an MPEG card.

QuickTime is designed to be able to use new CODECs as they come along, and has been adding them with each new version of QuickTime. You don't have to upgrade your application software to use new compression standards; QuickTime takes care of that as well.

Recently, Apple has been expanding on QuickTime technology to add new types of multimedia system software. QuickTime VR (the VR is short for virtual reality) lets you move around a photo-realistic or rendered scene at your own discretion. Instead of passively watching a video pan around a room, you can move 360 degrees around in a scene, gliding in and out of rooms and up and down stairs. A second QuickTime VR experience lets you pick up an object in a scene. You can examine an object closely and turn it completely around.

Unlike video clips, QuickTime VR scenes are actually panoramic still images that wrap completely around. As such, QuickTime VR files are very small, with a typical panoramic scene occupying several hundred kilobytes (not megabytes) of disk space. You don't need special goggles, wired gloves, or add-in cards to use QuickTime VR. You don't even need a particularly speedy Mac. All you need is a mouse.

Another QuickTime family member is QuickTime Conferencing, system software that lets you set up a video conference over an AppleTalk network or TCP/IP network, such as the Internet. You simply plug a video camera into a video port, connect the microphone that came with your Mac, and open a QuickTime Conference application such as Apple Media Conference. QuickTime Conferencing compresses and decompresses live video images and sound on the fly using a compression/decompression algorithm called H.261. You can also play QuickTime movies over a network.

QuickTime Conferencing has a smaller scope than the Mac's other multimedia technologies, and is not as important to everyday Mac users. However, it is concentrating on an area that may become very important soon—the Internet, the last big hurdle for multimedia to overcome.

Some analysts predict that the Internet will surpass CD-ROM as the main delivery mechanism for multimedia, but at this time, there is no clear idea of how that will happen. There are several competing technologies that promise to deliver audio, video, and animation over the Internet—you can even find QuickTime VR being used on some Internet sites—but none has a clear advantage. The biggest limitation of the Internet is that most people access it through modems. Even the fastest 28.8Kbps modems move data more slowly than floppy disks do, making slow CD-ROM drives seem like multimedia race cars.

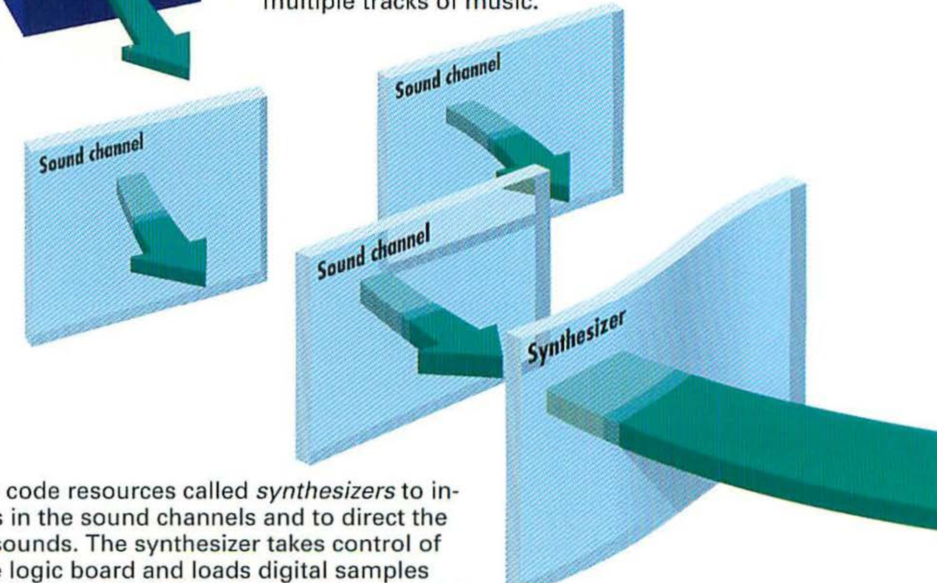
Whether on the Internet or on your desk, any new forms of multimedia probably won't seem like great technological breakthroughs to experienced Mac users. When video found its way onto the Mac desktop a few years ago, most everyday users weren't surprised. Multimedia has been an integral part of Mac computing for many years, and is likely to become more so.

Playing Sound

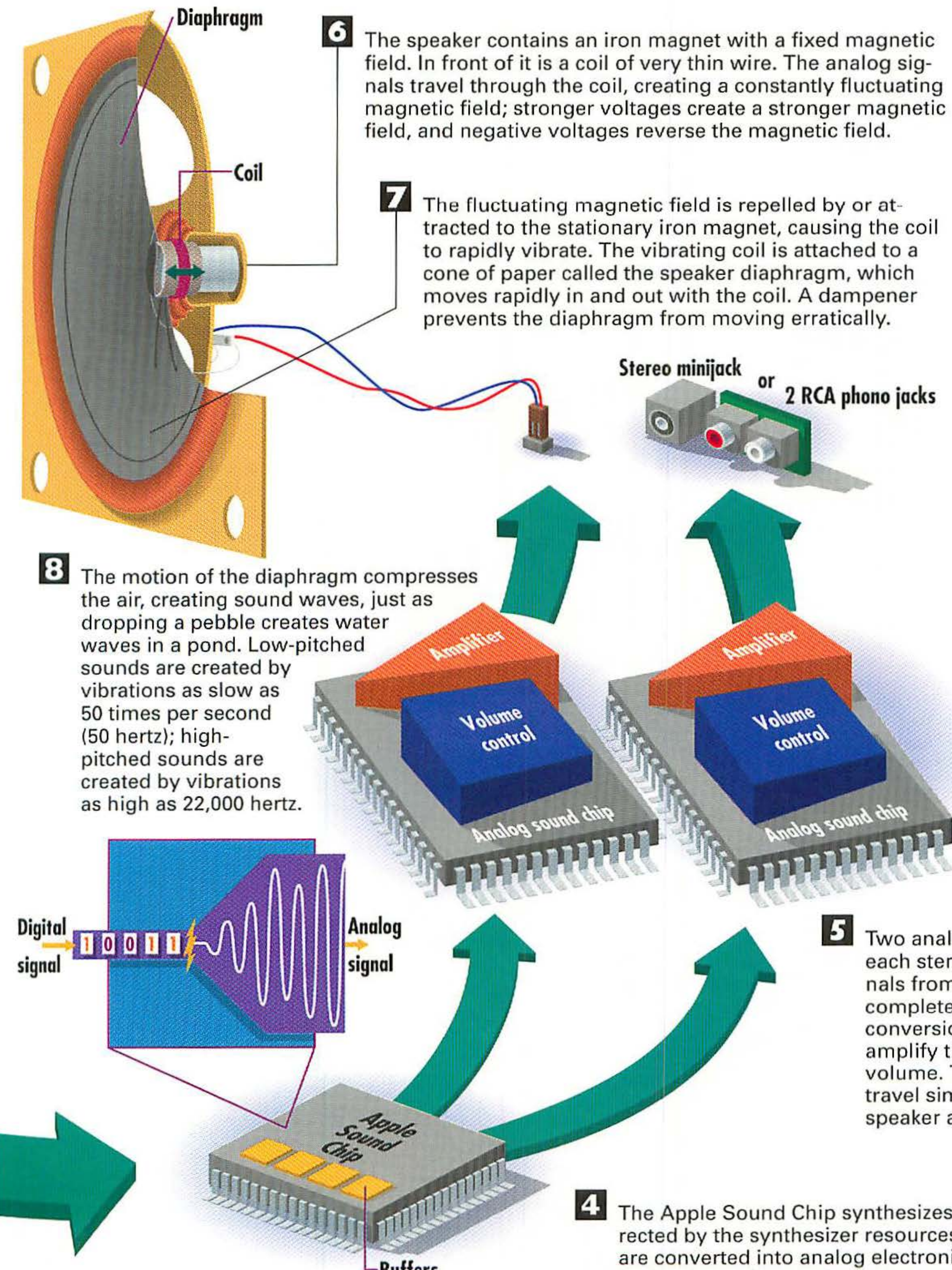
Sounds on the Mac can be stored as resources in the System folder (like fonts) or as sound files, which specify the sound commands and synthesizer routines to be used. The simplest sounds, such as the system alerts, are stored as sound resources. Most sounds can be stored as AIFF (audio interchange file format) files, which can be easily shared by different applications. The Mac can also play CD audio files, QuickTime Movie sound files, and PC .WAV files.

1 When an application needs to play a sound, it sends commands to the operating system's Sound Manager. The application can specify exactly what to do, or it can reference a resource or sound file that contains the sound commands and sound synthesizer. In the case of a musical CD, the commanding application is the Apple CD Audio Player.

2 The Sound Manager calls up instructions to play or modify a sound from ROM and either creates or calls up (depending on what the application needs) one or more *sound channels*, which are queues of sound commands required to produce a sound. The Sound Manager can create and run multiple sound channels at the same time. For instance, the system can play an alert beep (to tell you that you are having a printing problem, for example) at the same time an application is playing music for a multimedia presentation. Multiple sound channels can also be mixed to combine voice and music or multiple tracks of music.



3 The Sound Manager calls up code resources called *synthesizers* to interpret the sound commands in the sound channels and to direct the sound chips to produce the sounds. The synthesizer takes control of the Apple Sound Chip on the logic board and loads digital samples into the chip's buffers. There are several types of synthesizers stored in the System file: A square wave synthesizer generates simple sounds, a wave table synthesizer creates more complex sounds, and a sampled sound synthesizer plays digitally recorded sounds.



4 The Apple Sound Chip synthesizes complex sounds as directed by the synthesizer resources. The digital 1's and 0's are converted into analog electronic waves of constantly fluctuating voltages. A constant sound fluctuates at a constant rate—for instance, 440 times per minute for the musical note middle A.

5 Two analog processing chips (one for each stereo channel) smooth the signals from the Apple Sound Chip to complete the D/A (digital-to-analog) conversion. The analog chips then amplify the signals and regulate the volume. The resulting analog signals travel simultaneously to the Mac's speaker and stereo minijack.

6 The speaker contains an iron magnet with a fixed magnetic field. In front of it is a coil of very thin wire. The analog signals travel through the coil, creating a constantly fluctuating magnetic field; stronger voltages create a stronger magnetic field, and negative voltages reverse the magnetic field.

7 The fluctuating magnetic field is repelled by or attracted to the stationary iron magnet, causing the coil to rapidly vibrate. The vibrating coil is attached to a cone of paper called the speaker diaphragm, which moves rapidly in and out with the coil. A dampener prevents the diaphragm from moving erratically.

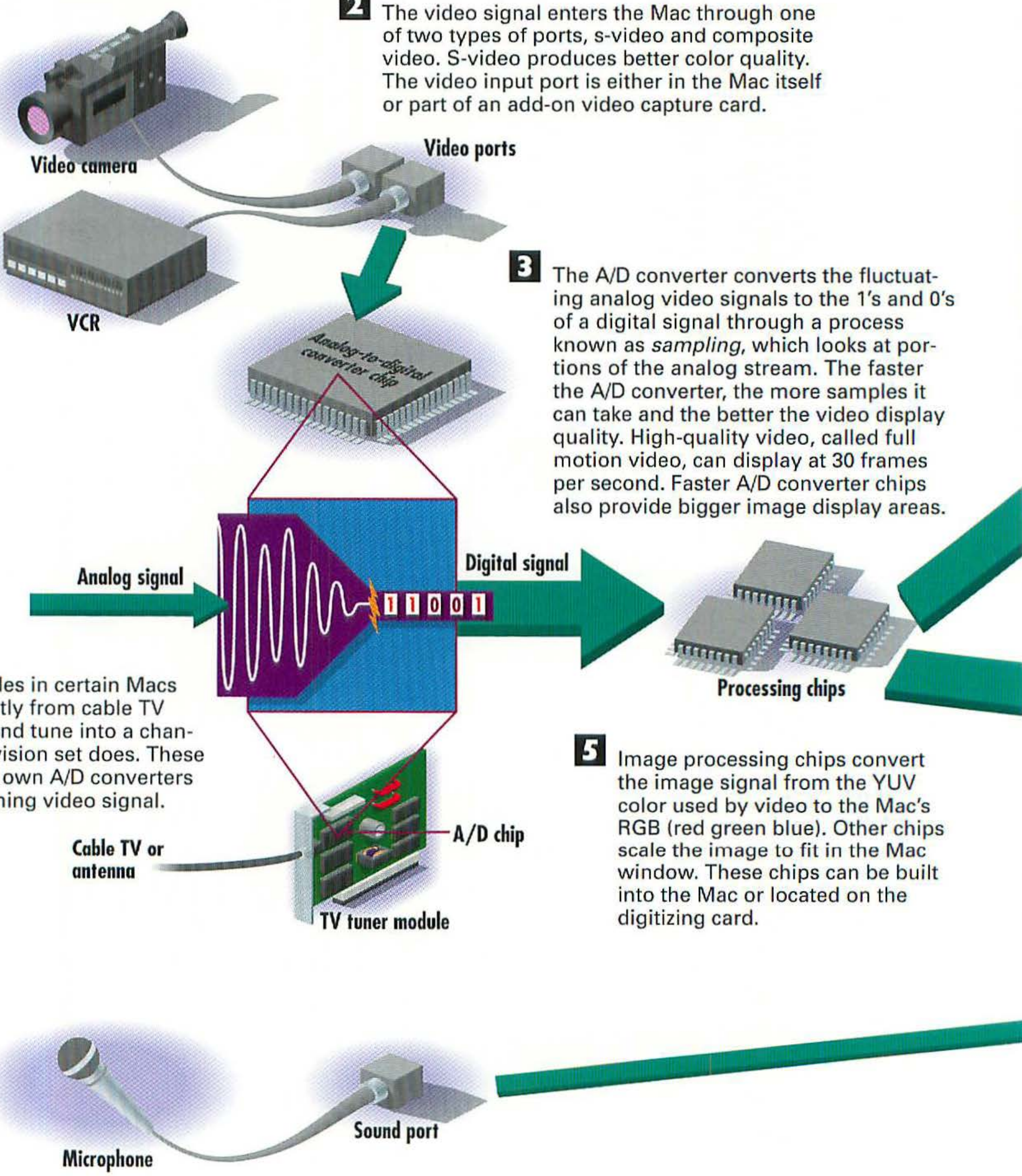
8 The motion of the diaphragm compresses the air, creating sound waves, just as dropping a pebble creates water waves in a pond. Low-pitched sounds are created by vibrations as slow as 50 times per second (50 hertz); high-pitched sounds are created by vibrations as high as 22,000 hertz.

Capturing and Displaying Live Video

1 The video source can be a video camera capturing an image, a VCR playing a tape, or some other video source.

2 The video signal enters the Mac through one of two types of ports, s-video and composite video. S-video produces better color quality. The video input port is either in the Mac itself or part of an add-on video capture card.

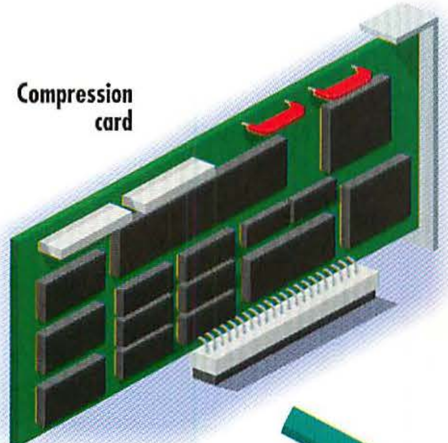
3 The A/D converter converts the fluctuating analog video signals to the 1's and 0's of a digital signal through a process known as *sampling*, which looks at portions of the analog stream. The faster the A/D converter, the more samples it can take and the better the video display quality. High-quality video, called full motion video, can display at 30 frames per second. Faster A/D converter chips also provide bigger image display areas.



4 The TV tuner modules in certain Macs accept signals directly from cable TV wires or antennas and tune into a channel, much as a television set does. These modules have their own A/D converters to digitize the incoming video signal.

5 Image processing chips convert the image signal from the YUV color used by video to the Mac's RGB (red green blue). Other chips scale the image to fit in the Mac window. These chips can be built into the Mac or located on the digitizing card.

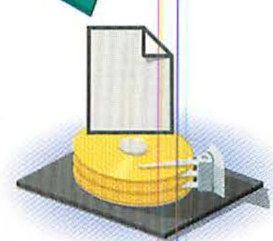
6 Signal compression sometimes occurs, either with a separate card, or with compression circuitry on a video digitizing add-in card. These cards use a compression algorithm, such as MPEG, to remove redundant data from the data stream. This codes the stream so it uses fewer 1's and 0's to represent the same images.



8 The image is displayed in the window of a video application, such as the Apple Video Player.

7 Macs don't have any built-in hardware compression, so if there is no add-in video card, compression does not occur at this stage.

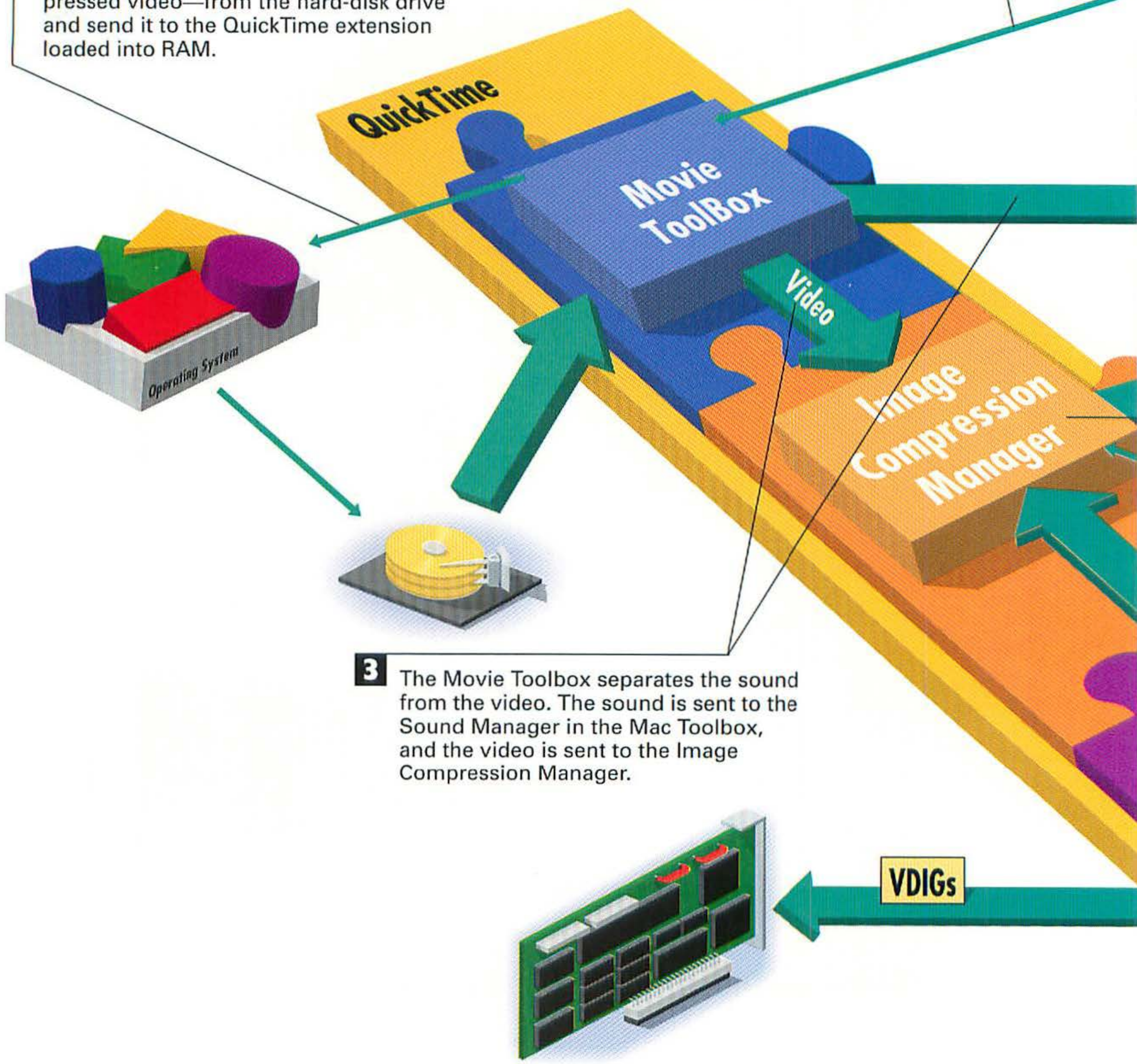
9 If you decide to save some of the video, the application calls QuickTime. QuickTime will compress the data using an algorithm that you have selected. QuickTime combines the video with sound from the sound input port in a Movie-format file, and saves the file to the hard disk.



QuickTime

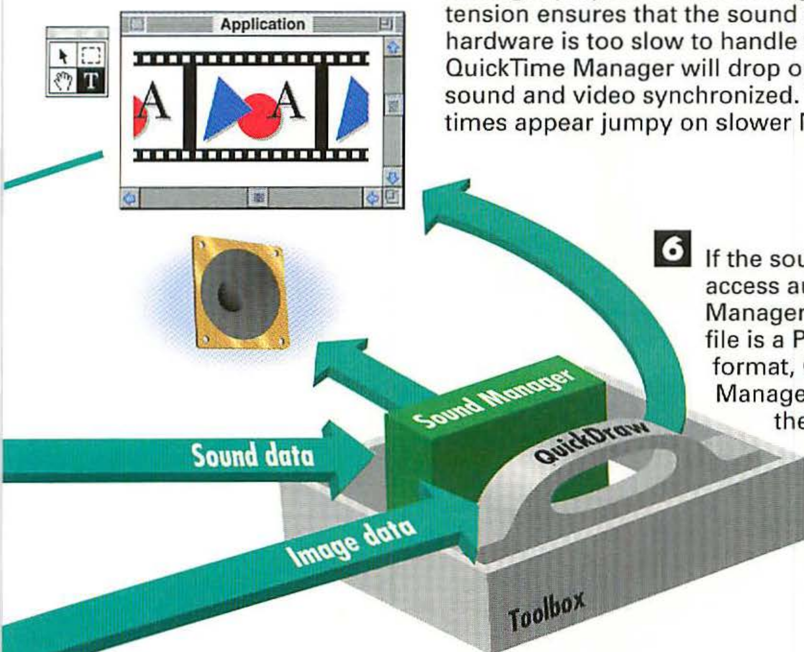
1 QuickTime allows any application that supports it to display video and sound movies. When a user pastes a QuickTime movie into a file and presses the start button, the application sends a call to the Movie Toolbox.

2 The Movie Toolbox asks the Mac operating system to retrieve the compressed movie data—which consists of sound and compressed video—from the hard-disk drive and send it to the QuickTime extension loaded into RAM.



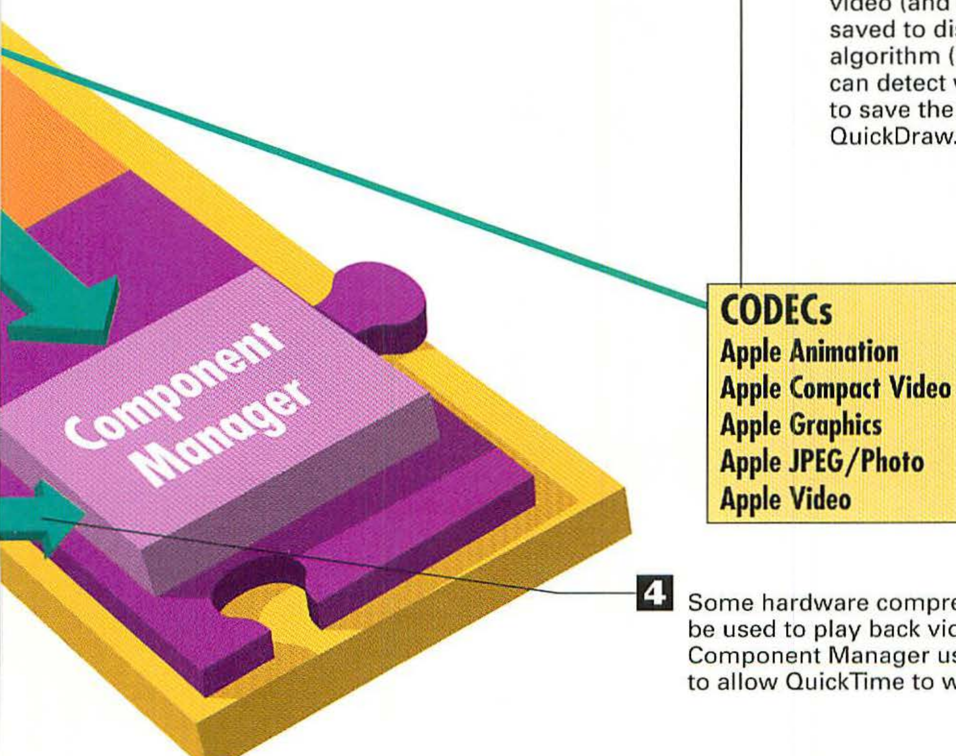
3 The Movie Toolbox separates the sound from the video. The sound is sent to the Sound Manager in the Mac Toolbox, and the video is sent to the Image Compression Manager.

7 QuickDraw displays the video in the application on screen while Sound Manager plays the audio through the Mac's speaker. The QuickTime extension ensures that the sound and video are played in sync. If the Mac hardware is too slow to handle the amount of data being displayed, the QuickTime Manager will drop out frames from the video to keep the sound and video synchronized. This is why QuickTime movies sometimes appear jumpy on slower Mac models.

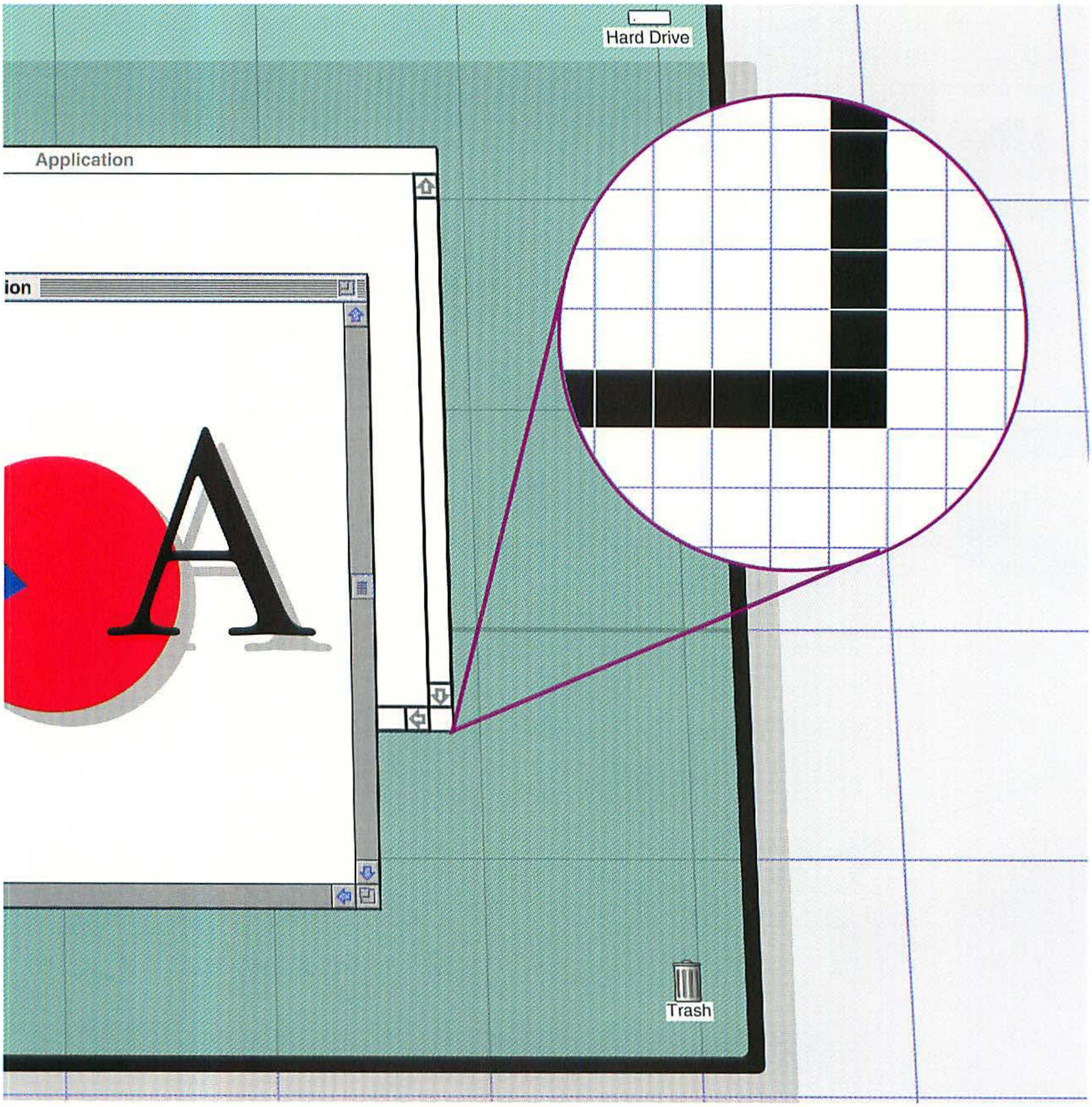


6 If the sound is compressed, QuickTime can access audio algorithms from the Sound Manager to decompress the sound. If the file is a PC sound file in the .WAV or .AU format, QuickTime will use the Sound Manager audio algorithms to convert them to the QuickTime format.

5 The Image Compression Manager decompresses the video (and compresses a video movie when it is being saved to disk) using a video compressor/decompressor algorithm (CODEC). The Image Compression Manager can detect which of half a dozen or so CODECs was used to save the file. It then sends the decompressed video to QuickDraw.



4 Some hardware compression add-in cards require that the card be used to play back video as well as capture it. QuickTime's Component Manager uses video digitizer components (VDIGs) to allow QuickTime to work with any card.


Hard Drive
Application
ion
A
Trash

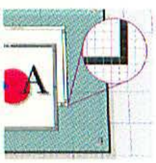
PART 6

DISPLAY

COMPUTERS are not TV sets. It's true that TV sets and computer monitors share some basic technology, but the similarity ends there. Computer displays give you better resolution, crisper colors, and sharper images than even the best television screens.

The main difference between TV and computer displays is that TVs are analog devices based on decades-old technology. Computer displays are digital affairs. Behind the screen that presents the Finder and your documents is a complex system of digital electronics that produces the images, as well as system software that tells the hardware what to do. The main piece of imaging system software is *QuickDraw*, a group of graphics routines in the Macintosh Toolbox (in ROM). QuickDraw is the director of the whole show.

Computer displays are also more versatile than TV displays. For instance, TV screens are locked into a 3:4 image aspect ratio at one size, often cutting off the left and right sides of movies. Many of today's Mac displays let you select the number of *pixels* (the basic dots that make up an image), so that if you want to see more of a document at one time, you can switch from 640-by-480 pixels to 1,024-by-768 pixels. Multiscan monitors can display different pixel resolutions, but all Mac displays let you choose the number of colors you want. Some applications will change the number of colors for you if they require a certain setting.



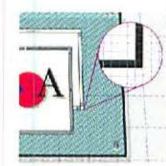
Video display monitors come in many shapes and sizes, from the 9-inch built-in screens of the classic Macs to 21-inch double-page displays. PowerBooks come with a different type of screen called a flat-panel LCD (liquid crystal display), which is more like the readout on a microwave oven than the display on a television set. If one display isn't big enough, most desktop Macs and PowerBooks let you add more, enabling you to use them together as if they were one. You can also connect desktop and laptop Macs to large projection display systems for giving talks in front of live audiences. And, with a special adapter cable, many Mac models allow you to connect monitors used with PC-compatible computers.

Monitors also vary in the number and type of shades they can display. The first Macs had *monochrome* displays, as do some PowerBook models. Monochrome displays use only two colors, black and white. *Grayscale* presents many shades of gray between black and white for a more realistic rendition of graphic images. Technically, so-called black-and-white photographs and television programs are actually grayscale images.

Color Macs have been available since 1987, and the first color PowerBooks were introduced in 1992. Color monitors can display in color, grayscale, and monochrome. The maximum number of gray values or colors that a particular Mac model can display usually varies between 256 and more than 16 million. Macs that can display millions of colors can produce images with photographic realism. However, all those colors take more processing power and can slow down actions such as scrolling. Fortunately, the Mac's system software contains the Monitors control panel, which lets you select fewer colors to improve screen performance when you don't need the colors. You can also choose to display grayscale or black and white to further improve performance.

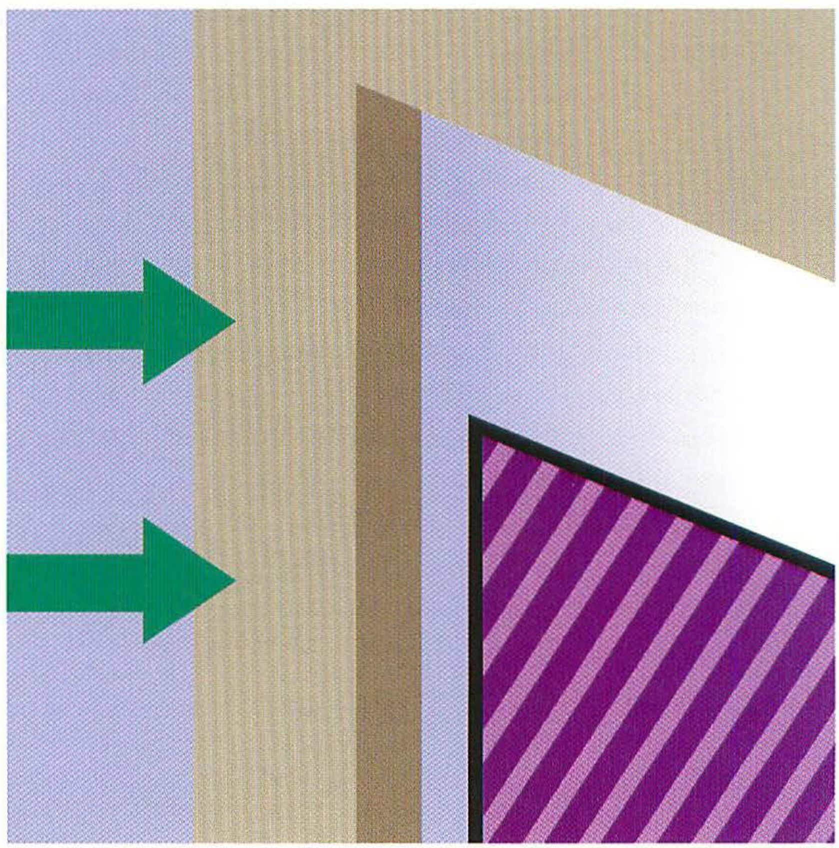
Displays also take a lot of energy—in fact, most of the energy your Mac consumes. You can always safely turn off a monitor while you leave a Mac turned on. Many Macs and monitors can do this for you. Current Macs come with the Energy Saver control panel that will turn off the monitor after a specified period of inactivity. You can turn the monitor back on again when you hit the keyboard. Some monitors have their own energy-saving features that put the monitor in various "sleep" modes when you are not using it.

The Mac uses a display technology called RGB video, short for red, green, and blue. RGB video creates every displayable color by mixing red, green, and blue light in different amounts. Software applications may use different color systems to enable users to set colors, but the Mac translates them into RGB values for displaying the colors on screen.



All Macs made today have built-in video circuitry and don't require add-in video cards. However, you can add video display cards and video RAM to accommodate bigger-sized screens, more colors, and faster drawing speed. Like other parts of the Mac, the display system is plug-and-play. Replacing a monitor or adding additional monitors is simply a matter of plugging in any required cards and cables—no switches need to be set, and no programming is required. Installing a Mac monitor is not quite as simple as plugging in a television set—but not as hard as programming your VCR.

CHAPTER 24 How QuickDraw Works



QUICKDRAW plays a key part in Macintosh graphics, acting as the supervisor for the other graphics managers in the System software and the video hardware to create the text and graphics you see on screen. QuickDraw provides consistent display for all applications and is responsible for the Mac's what-you-see-is-what-you-get, or WYSIWYG (wiz-ee-wig), display. With WYSIWIG, an inch on the screen is an inch on printed paper, and lines of text will end in the same place on screen as on paper.

Since the first Mac, QuickDraw has been evolving along with the Mac hardware and software. Part of the original Mac's ROM, QuickDraw was revamped to add color in 1987. When System 7 was released in 1991, QuickDraw was expanded to communicate with RAM in 32 bits. In 1993 Apple released QuickDraw GX, an enhancement to QuickDraw that offers more color, graphics, text, and printing abilities. QuickDraw GX is an option under System 7, but will be used extensively in Mac OS 8.

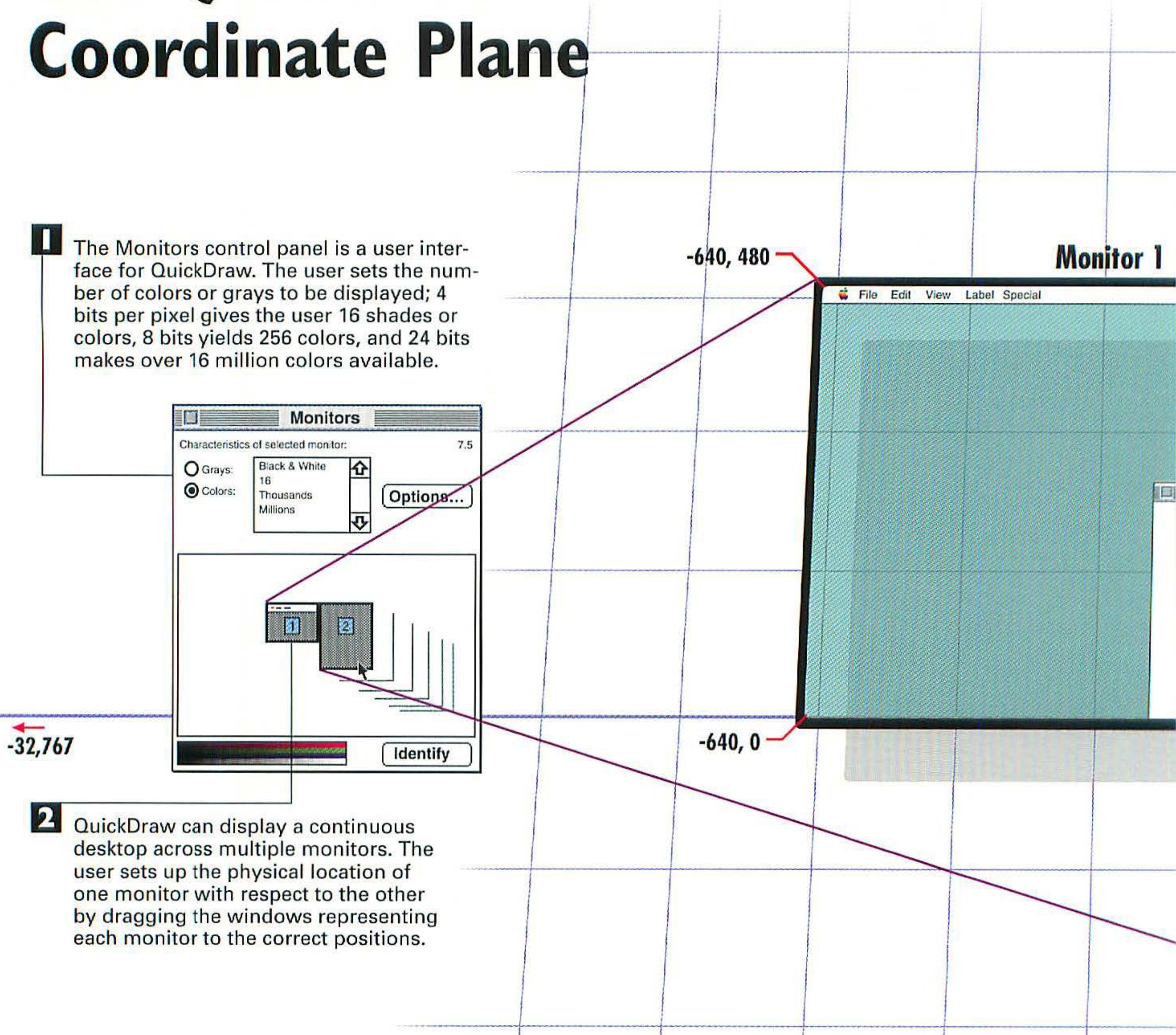
QuickDraw displays screen information at a resolution of 72 dots per inch. This may sound like a random number, but the size was chosen to match a unit of measure called a *point* in typography, which is 1/72 of an inch in size. QuickDraw always creates graphics at 72 dots per inch, though some monitors can compress or expand this. However, when this happens, you no longer have WYSIWYG.

Each dot on screen is called a pixel and is the smallest possible area that can be drawn. The number of colors that can be displayed is often given in bits per pixel. (See Chapter 10 for an explanation of the binary number system.) A pixel in a 1-bit monitor has only two numbers available, 0 and 1, to represent two shades, black and white. A display in 2-bit video has four binary numbers available: 00, 01, 10, and 11, to represent four colors. Similarly, 8-bit video can use 256 colors, and 32-bit video can provide over 16 million colors, more than the number of pixels on a screen. In 32-bit video, 24 bits are used for colors; the extra 8 bits are used to represent other information.

The extra bits needed to display more colors require more processing power. For graphic-intensive uses such as animation and image processing QuickDraw isn't so quick. QuickDraw GX not only improves drawing speed, making animation smoother, but also enables more sophisticated features, such as mixing the colors of overlapping graphic objects.

The QuickDraw Coordinate Plane

1 The Monitors control panel is a user interface for QuickDraw. The user sets the number of colors or grays to be displayed; 4 bits per pixel gives the user 16 shades or colors, 8 bits yields 256 colors, and 24 bits makes over 16 million colors available.

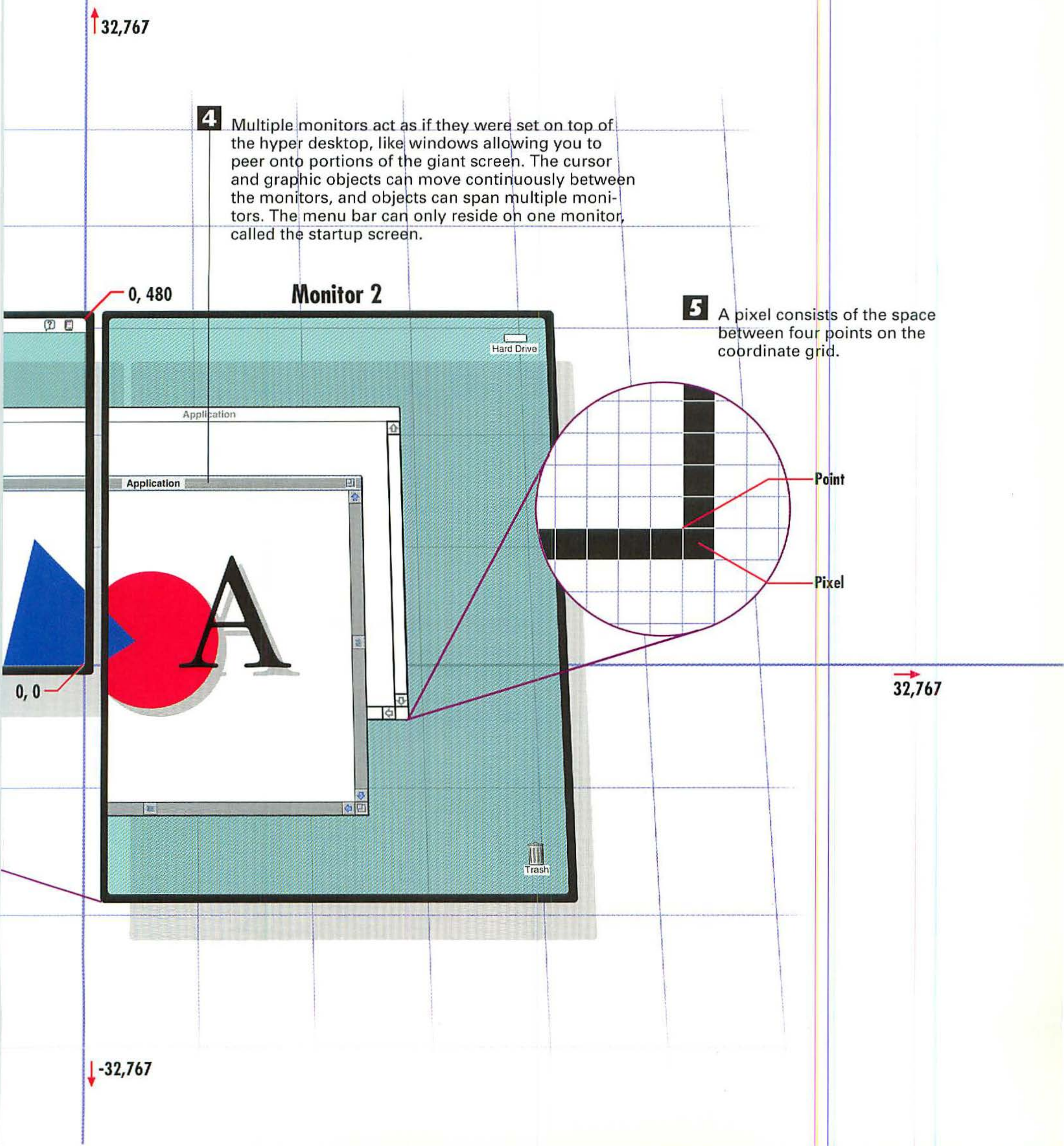


2 QuickDraw can display a continuous desktop across multiple monitors. The user sets up the physical location of one monitor with respect to the other by dragging the windows representing each monitor to the correct positions.

3 QuickDraw creates a giant hyper desktop area: a coordinate plane of numbered points. This plane is 65,535 points tall by 65,535 points wide—the binary number 11111111,11111111, takes up 16 bits. The points are numbered in both horizontal and vertical directions, so that a pair of numbers represents a point. QuickDraw measures the location of objects and their motion in terms of this coordinate system.

4 Multiple monitors act as if they were set on top of the hyper desktop, like windows allowing you to peer onto portions of the giant screen. The cursor and graphic objects can move continuously between the monitors, and objects can span multiple monitors. The menu bar can only reside on one monitor, called the startup screen.

5 A pixel consists of the space between four points on the coordinate grid.



The RGB Color Model

QuickDraw and Macintosh video hardware and monitors use the *RGB* (red, green, and blue) *color model* to specify colors; the model is based on the way colored light behaves. Some software may use other color models. CMYK (cyan, magenta, yellow, and black), used in color publishing, is based on the way colored inks behave, and HLS (hue, lightness, and saturation) is sometimes used in graphic arts. Although software can use these other models, QuickDraw converts them to RGB values for display.

3 The values for red, green, and blue components correspond to an imaginary three-dimensional plot called a color space. There are three axes, one red, one green, and one blue.

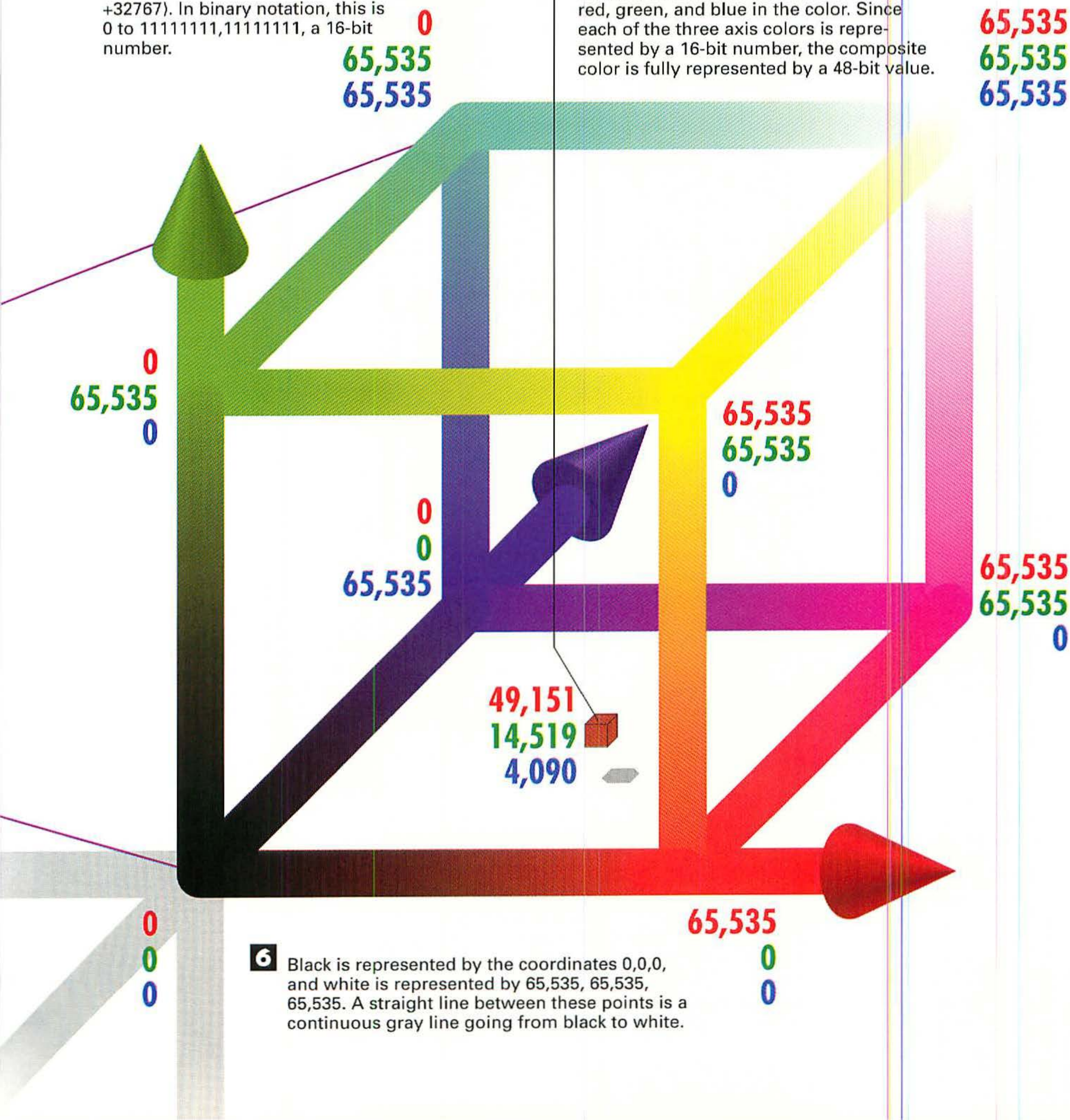
1 The *Color Picker* is the QuickDraw user interface for color selection. It lets users specify new colors for objects within applications. The user can select a color by clicking on a color on the color wheel or typing in number values for the red, green, and blue components of the color. The color picker also gives the equivalent values for hue; saturation; and brightness, an alternative method for describing a color in the RGB model.

2 In the RGB model, any color can be created by mixing red, green, and blue in different amounts. The color is additive: The more of each color you add, the closer you get to white. With no amount of any of the colors, you get black, like a room with the lights turned off.

MAC FACT With 48 bits worth of numbers to play with, the Color Picker can specify over 280 *trillion* colors—far more than the Mac can display or the human eye can see. QuickDraw allows this many possible colors to best match any conceivable color that can be created by video hardware.

4 Like the coordinate plane in QuickDraw's hyper desktop, each axis can have 65,535 values, going from 0 to 65,535 (instead of –32767 to +32767). In binary notation, this is 0 to 11111111,11111111, a 16-bit number.

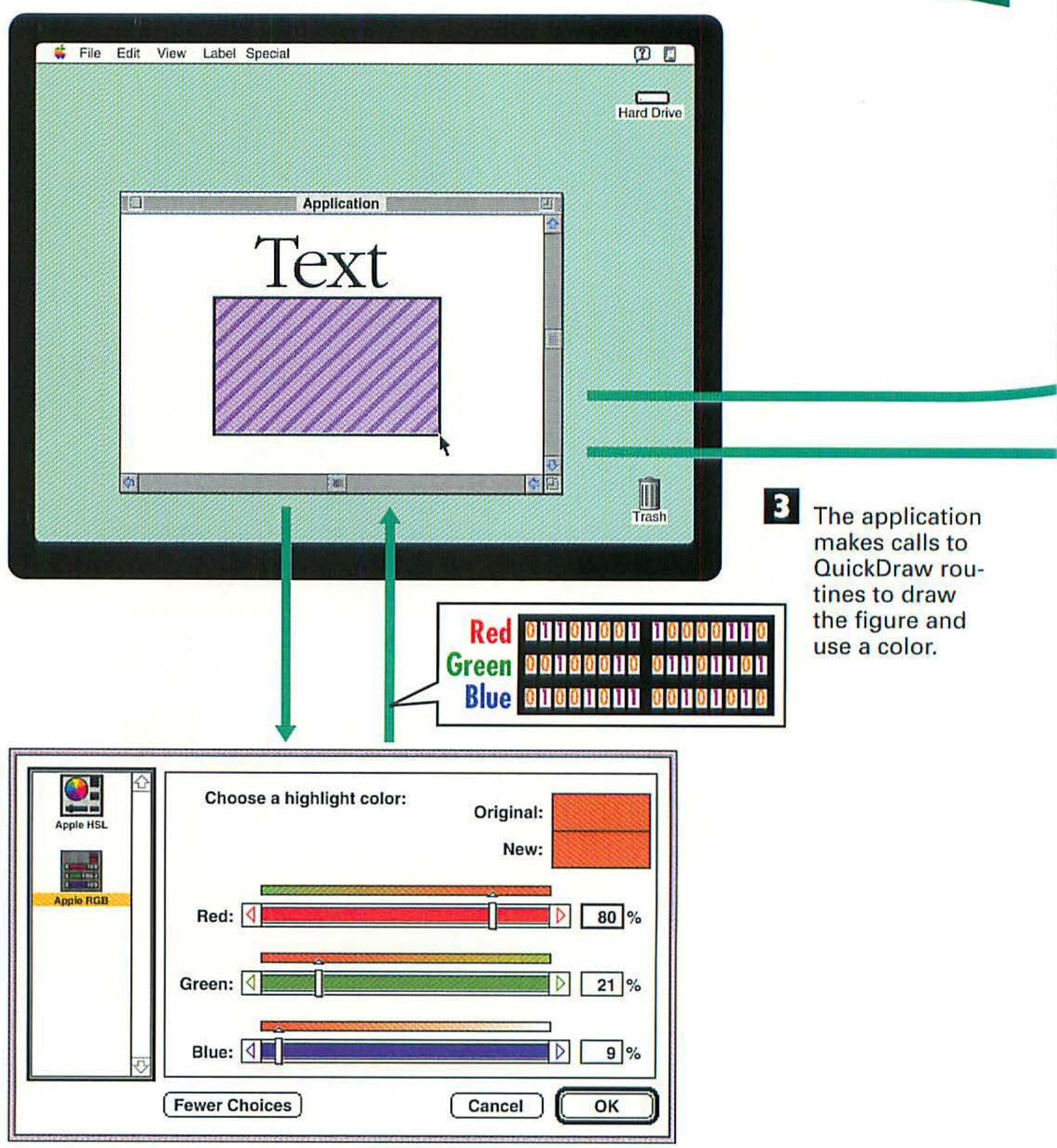
5 A point in the interior of the space represents a color. This color is described in the three-dimensional graph by its three coordinates, which represent the amount of red, green, and blue in the color. Since each of the three axis colors is represented by a 16-bit number, the composite color is fully represented by a 48-bit value.



6 Black is represented by the coordinates 0,0,0, and white is represented by 65,535, 65,535, 65,535. A straight line between these points is a continuous gray line going from black to white.

QuickDraw at Work

Most of what QuickDraw draws is created from mathematical descriptions of objects. QuickDraw can also use a technique called direct pixel, or bit image, which is faster but requires more RAM. QuickDraw manages the translation of a color chosen by the user into a color the hardware can create.



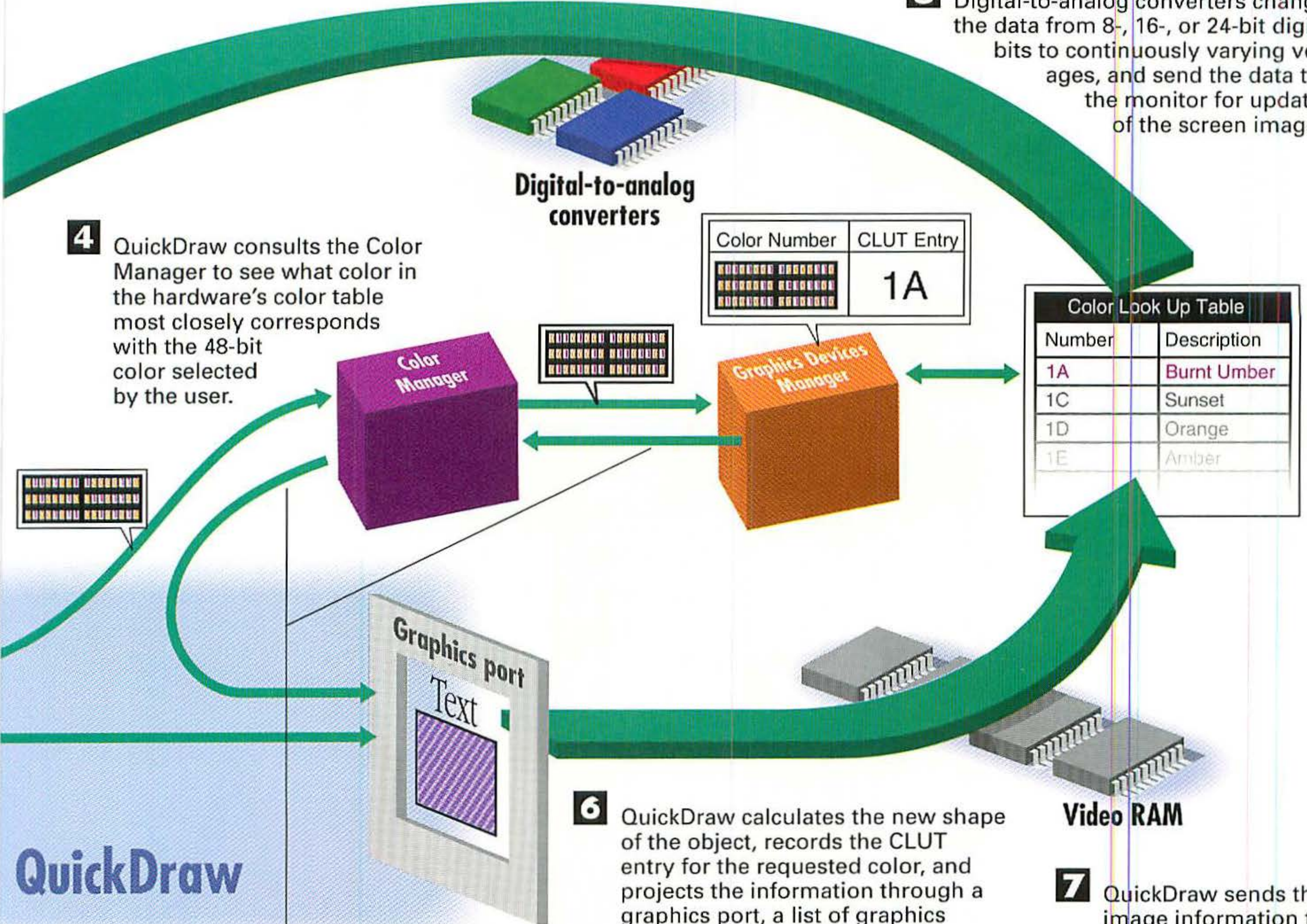
1 A user draws or types in an application window. In this example, the user draws a box with a filled pattern and chooses a color for it.

2 The application consults the Color Picker for the color the user has chosen. The Color Picker returns with a 48-bit RGB description of the color.

3 The application makes calls to QuickDraw routines to draw the figure and use a color.

8 Digital-to-analog converters change the data from 8-, 16-, or 24-bit digital bits to continuously varying voltages, and send the data to the monitor for updating of the screen image.

4 QuickDraw consults the Color Manager to see what color in the hardware's color table most closely corresponds with the 48-bit color selected by the user.



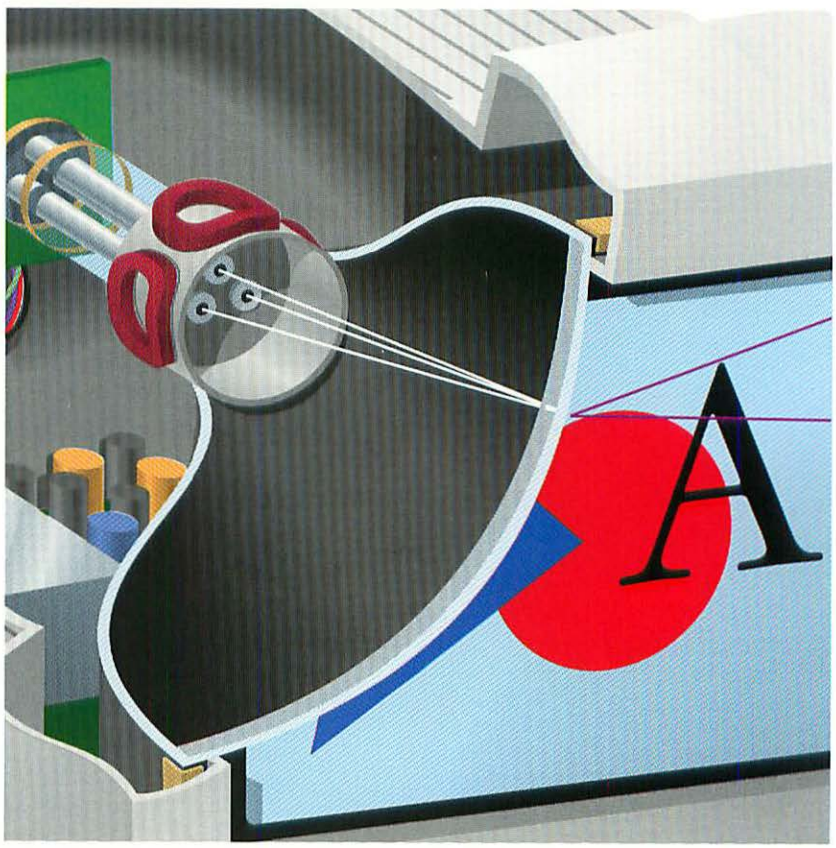
5 The Color Manager asks the Graphics Devices Manager which color comes closest to the color requested by the user. The Graphics Devices Manager keeps a record of the video hardware's *color look-up table* (*CLUT*) to see what colors are available. The CLUT entry for that color is returned to QuickDraw. The CLUT entry is described by 8, 16, or 24 bits, depending on what the user has set in the Monitors control panel.

6 QuickDraw calculates the new shape of the object, records the CLUT entry for the requested color, and projects the information through a graphics port, a list of graphics parameters associated with the particular window being used. The graphics port records the size and shape of the window, as well as the location of objects, the thickness of the pen being used, fill patterns, fonts, and color attributes. Graphics ports enable QuickDraw to be device independent: A different graphics port is used for each window and output device.

7 QuickDraw sends the image information to video RAM in video hardware. Video RAM passes the color entry to the CLUT, which looks up the color requested. (If a bit image is used, this step is skipped, and the specified color is sent directly to the display.)

MAC FACT The old black-and-white Macs can connect to color monitors because they have color QuickDraw in ROM. The original QuickDraw in early Macs was designed for black-and-white display, but actually had the ability to display eight colors. Although the Macs themselves had no built-in way to connect to a color monitor, at least one company made an add-in card for the SE that connected it to an eight-color monitor. In fact, applications that only require eight colors can access these simple precolor color routines, which still exist in modern QuickDraw.

CHAPTER 25 How a Video Monitor Works



A video monitor is a producer of optical illusions. To the eye, a color screen appears as a steady image of solid colors, like a movie being projected on a screen. You may think you're looking at a photographic image of a basket of flowers, but the color monitor actually displays tiny dots of red, green, and blue, in varying intensities. The colored dots are so close that they blend in different color strengths, only seeming to produce all the colors of the rainbow. And although the image is projected from within, it is not flashed on the screen all at once, as from a movie projector, but rather is drawn one $\frac{1}{72}$-of-an-inch picture element (or pixel) at a time. The pixels are redrawn so rapidly that the screen appears to display a solid image. Monochrome and grayscale monitors also draw their images one pixel at a time, but use dots of white or gray.

The dots that make up the screen image, 72 of them per inch, are created by the Mac's QuickDraw routines and are passed on to the monitor. This resolution gives you WYSIWYG display, where an inch on screen equals an inch on the printed page. Some monitors deviate from the standard 72 dots per inch: Large screen monitors sometimes have slightly more pixels per inch to shrink the image and fit more on screen. 12-inch monitors display less than 72 pixels per inch (68), which enlarges the image slightly.

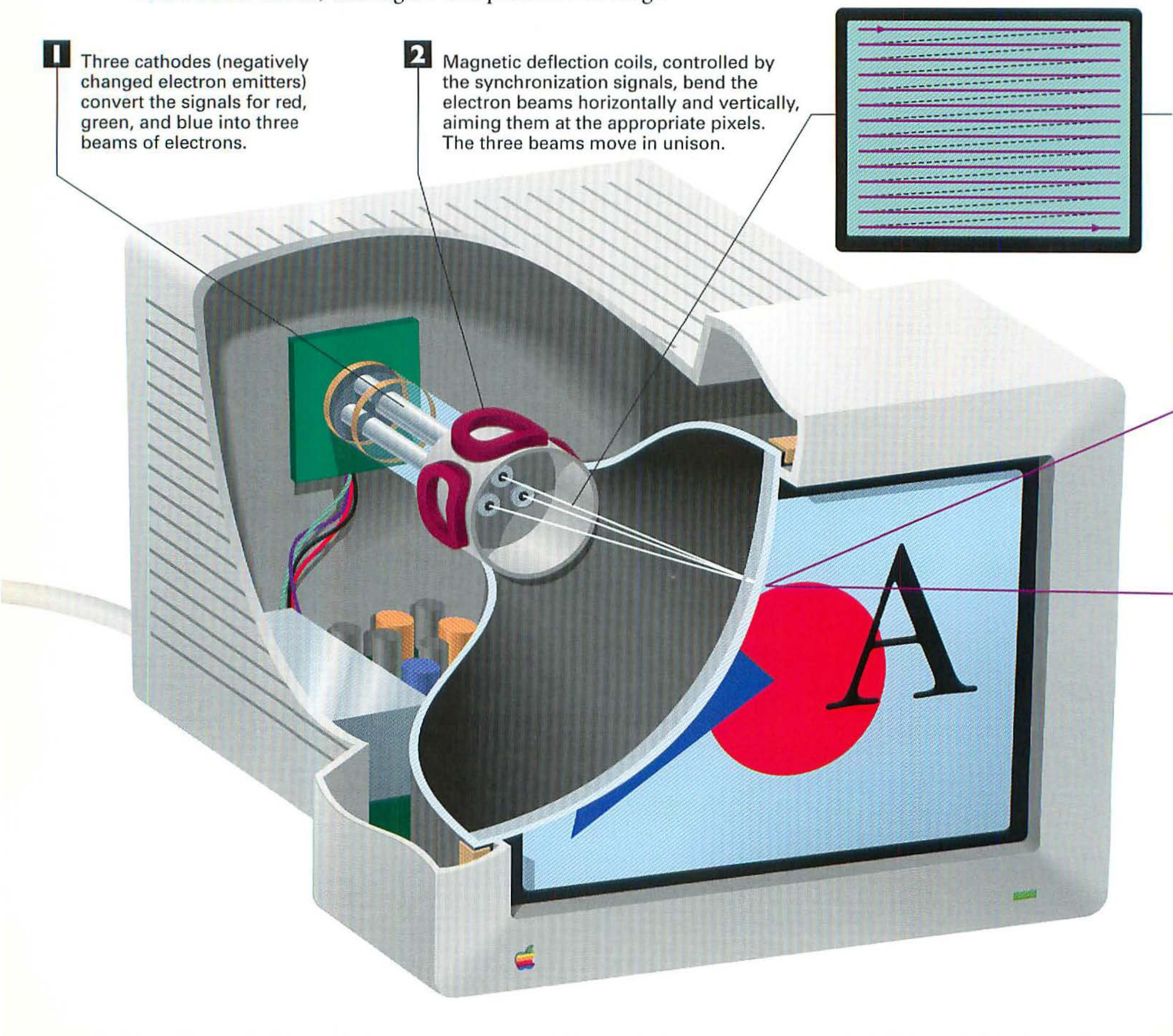
The number of pixels that fit vertically and horizontally on the screen is a measure of the screen size. For instance, the Mac standard for 13-inch displays is 640 pixels across the screen by 480 down. A monitor that can display a full 8½-by-11-inch page at 72 dots per inch usually measures 870 by 640 pixels. That's 556,800 pixels, each being redrawn at least 60 times a second—enough to fool any eye.

Color Video Monitor

A monitor is controlled by the video circuitry on the Mac's logic board or on a separate video board in an expansion slot. Most of the monitor consists of a cathode-ray tube, the interior of which is a vacuum. Inside the tube, beams of electrons are projected against *phosphors* on the inside of the screen, which glow and produce the image.

1 Three cathodes (negatively changed electron emitters) convert the signals for red, green, and blue into three beams of electrons.

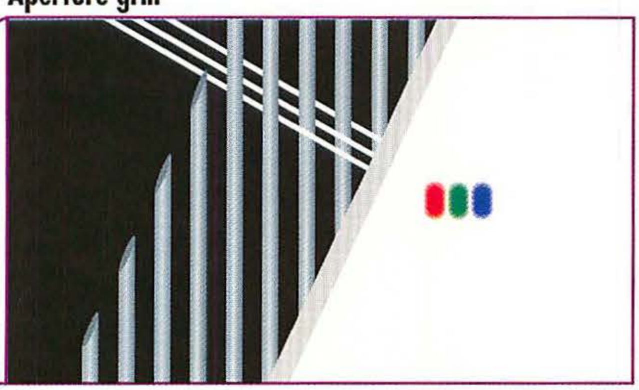
2 Magnetic deflection coils, controlled by the synchronization signals, bend the electron beams horizontally and vertically, aiming them at the appropriate pixels. The three beams move in unison.



Various types of signals come to the display through an internal or external cable, which typically ends in a 15-pin connector. Three of the signals control the levels of red, green, and blue arriving from the digital-to-analog converters on the Mac's video circuitry. The horizontal and vertical synchronization signals tell the monitor when to hit each pixel with what signal. There are also several ground signals, which provide 0 volts as a reference voltage.

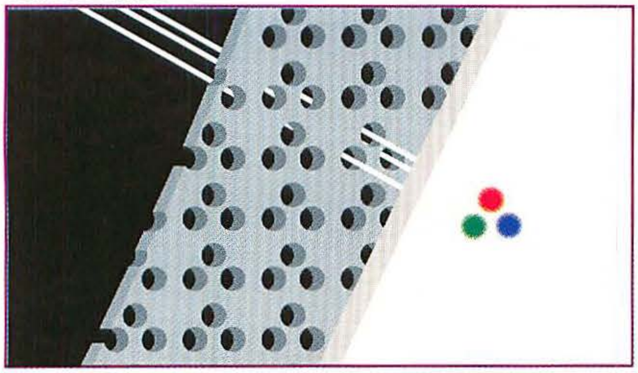
3 The electron beams are aimed at the screen, one pixel at a time. The beams start at the top-left corner of the screen, and they scan horizontally from left to right. The beams then shut off, aim at the left of the next line, and start shooting at the pixels from left to right. This continues until the beams reach the bottom right, when the beams shut off, return to the top left, and begin scanning again. A complete image is painted on screen between 60 and 75 times per second. This is called the *vertical scan rate*. At this rate, the eye cannot detect the individual scans. The higher the scan rate, the less chance you will perceive the flicker on the screen.

Aperture grill



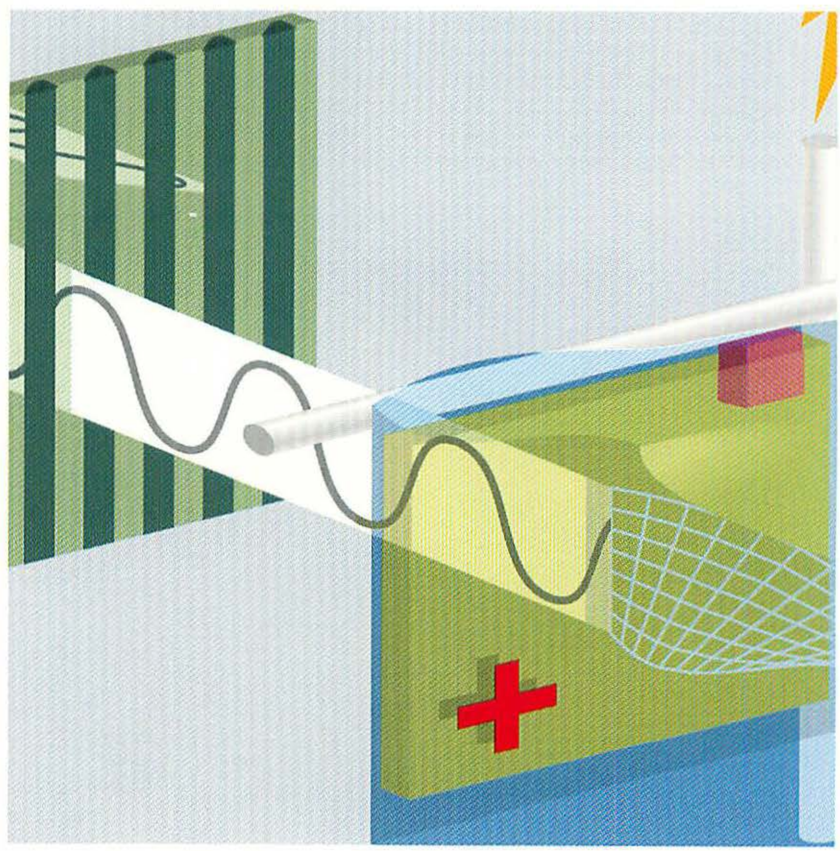
4 To help focus the electron beams on the spots they are aimed at, and to separate one pixel from the next, the beams pass through some sort of grating. Many Mac monitors contain Sony Trinitron tubes, which use an aperture grill made of thin parallel wires, each about 195 microns thick. Aperture grills tend to produce sharper and brighter images than monitors that employ a *shadow mask*, a thin metal plate with holes in it. The *dot pitch*, the distance between the holes, is typically 0.2 to 0.3 millimeters. The closer the holes, the sharper the image.

Shadow mask



5 The electron beams each hit a phosphor. (Phosphors coat the inside of cathode-ray tubes and glow when activated by electron beams.) One of the phosphors glows red when excited by electrons, another glows green, and a third glows blue. A pixel contains one of each type of phosphor. After the beam moves on to the next pixel, the phosphors continue to glow until the beams strike them again on the next pass. The phosphors are so close to each other that they trick the eye into seeing the blended colors. The intensity of each beam determines how bright the phosphor will glow. White is produced when all three beams hit at maximum intensity, black when all three are shut off.

CHAPTER 26 How PowerBook Displays Work



CATHODE-RAY tubes, which had been used for over a half a century in television monitors, adapted well to desktop computers in the 1980s. However, cathode-ray tubes are much too big and require too much power for today's battery-operated laptop computers. PowerBooks and PowerBook Duos use *flat-panel displays*, which measure less than a half an inch thick.

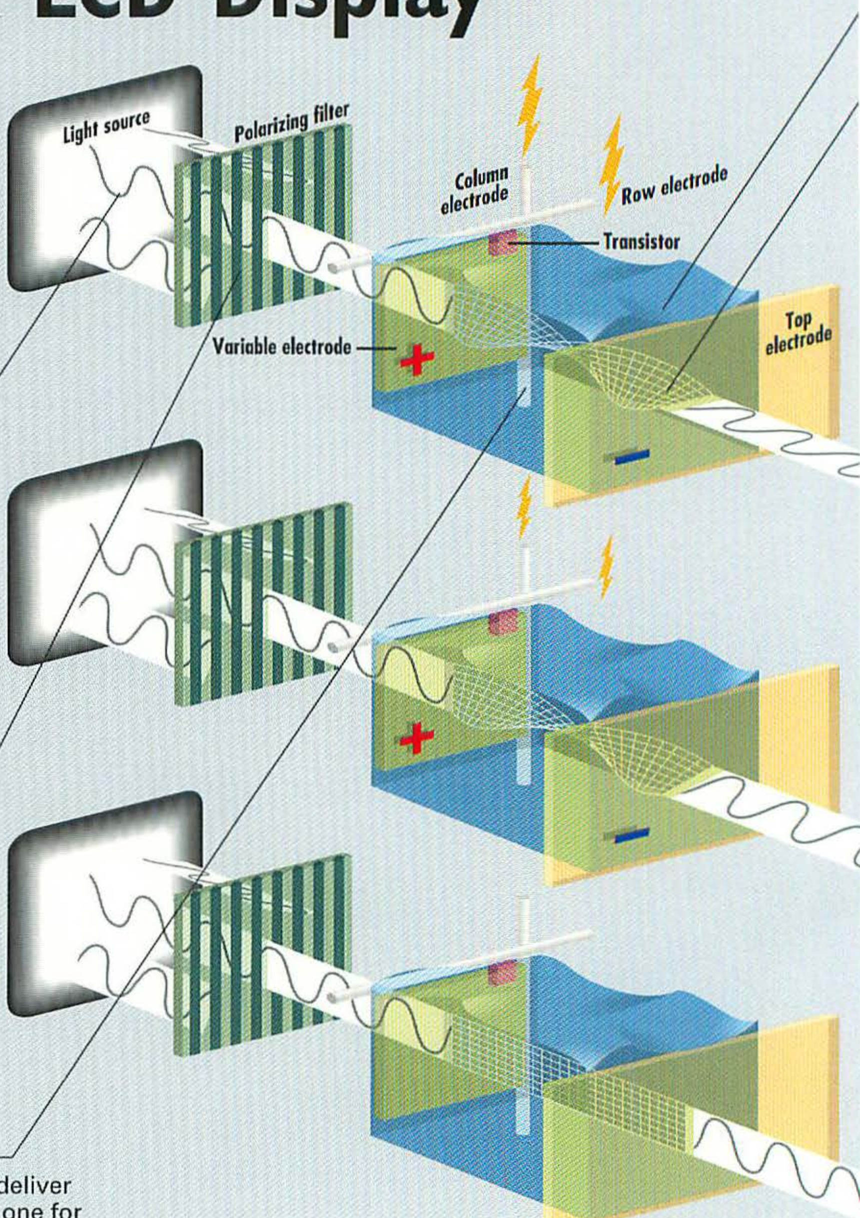
Flat-panel displays use the liquid crystal display (LCD) technology first popularized in digital watches in the 1970s, and they bear little resemblance to the cathode-ray monitors found on desktop Macs. For instance, it is easy to see individual square pixels on the PowerBook displays. Each pixel of a flat-panel display is filled with transparent material called *liquid crystal*, an odd substance that becomes opaque when an electric charge is applied to it.

Grayscale LCDs are slightly more complicated than monochrome displays, in that they vary the electric charge applied to the liquid crystal to produce several levels of translucence, which in turn appear as several layers of gray. Color is even more complicated, requiring three rays of light for each pixel—each ray travels through either a red, green, or blue filter. Color therefore requires more power, and drastically shortens PowerBook battery life.

Several types of flat-panel displays are used in PowerBooks and PowerBook Duos. There are monochrome, grayscale, and color LCD panels, each of which uses one of two types of design: *active matrix* or *passive matrix* (also called supertwist). Active matrix technology—used in the monochrome PowerBook 170, grayscale 180, and color S300c—is superior to passive matrix. Active matrix displays produce sharper images, have purer colors and grays, have better contrast, and can be viewed from wider angles than passive matrix displays. Active matrix screens don't draw images one pixel at a time, as do cathode-ray tubes and passive matrix displays. Instead, active matrix screens light up all at once. Active matrix is more expensive than passive matrix, but it has also caught up in quality with cathode-ray tubes. In some ways, active matrix displays are superior to cathode-ray tubes—they have no distortion at the edges and corners, as is often found in cathode-ray monitors.

Active Matrix LCD Display

Flat-panel displays make use of two rather odd properties of physics: the polarization of light, a phenomenon used in sunglasses to partially block out light; and the interaction of light with liquid crystal, a liquid material that shows some properties of solid crystals.



1 The backlighting panel produces white light at the back of the display. Light consists of electromagnetic waves that vibrate in a single plane. White light produces many light waves, which vibrate in every different direction. White light also contains light of every color.

2 The light passes through a *polarizing filter*—a material with embedded crystals—which acts like a grating, allowing only light vibrating in one direction to pass.

3 Thin wires (column and row electrodes) deliver the video signals to thin-film transistors: one for each pixel on monochrome and grayscale screens; three for each pixel for color displays. In monochrome screens, the transistors are either on or off. However, with grayscale and color displays, the transistor puts out several levels of current, which will eventually represent different levels of gray or color.

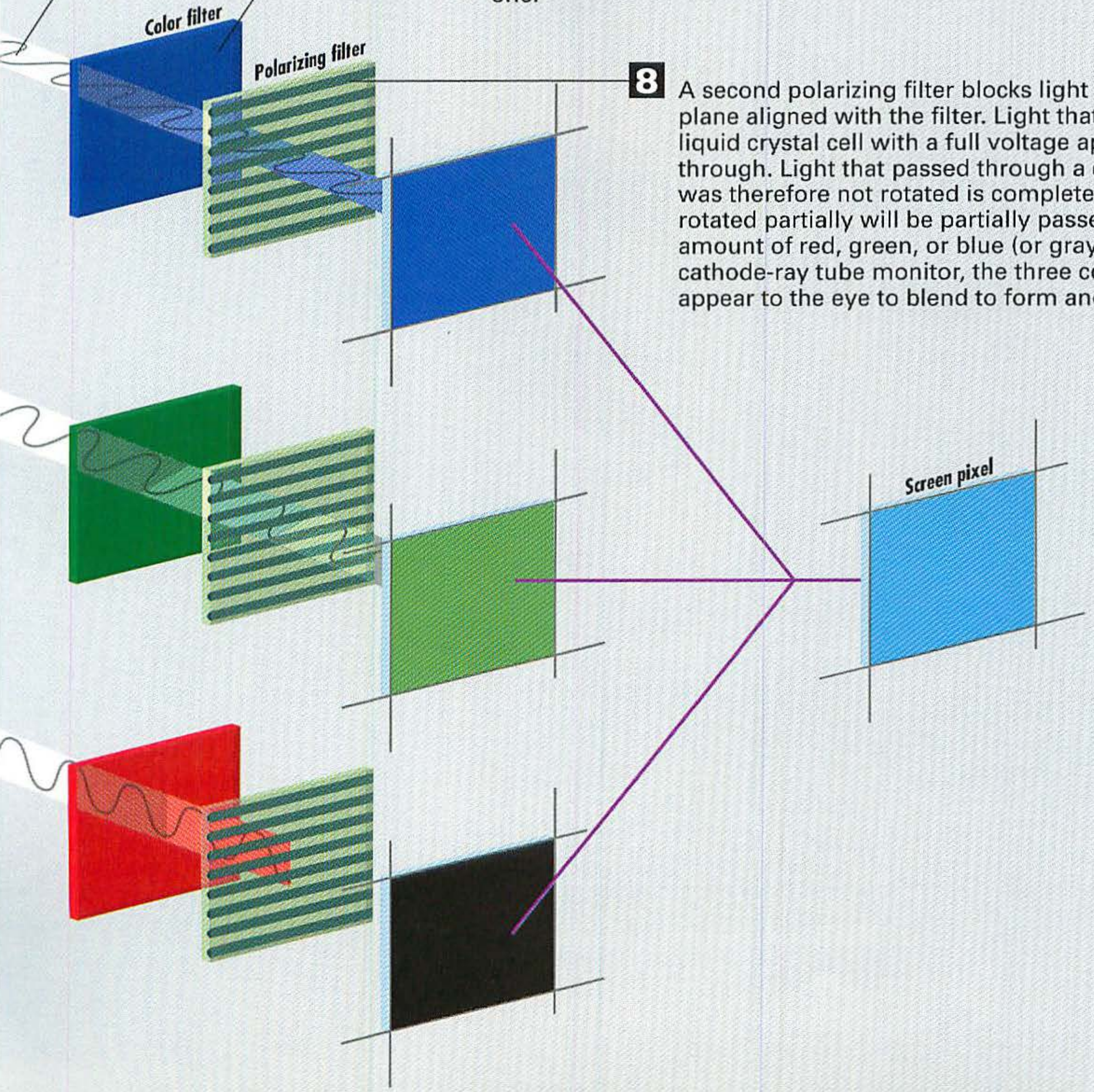
4 Each pixel contains a cell filled with liquid crystal. The transistor drives a transparent variable electrode on one face of the liquid crystal cell, applying a voltage to it.

5 Molecules within the liquid crystal have a fixed orientation with respect to one another. When an electric voltage is applied to the transparent electrode, the orientation of the molecules begins to twist through the cell. Different voltages twist the molecules to different degrees.

6 The twisting of the molecules affects the light passing through the liquid crystal material, rotating the plane of vibration of the light waves.

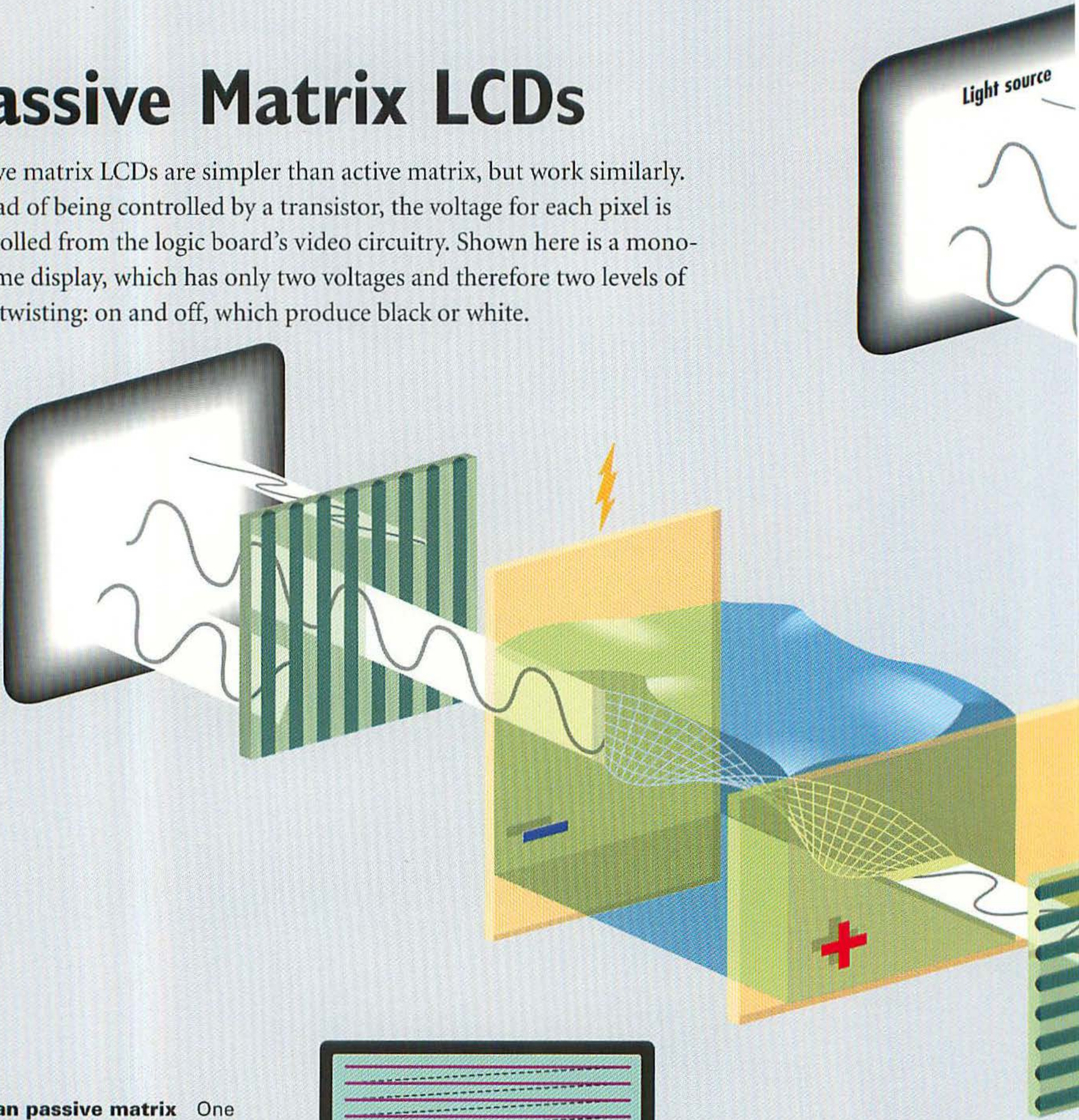
7 In color displays, the three rays of light for each pixel each pass through a color filter, which is either red, green, or blue. Filters block all colors except one.

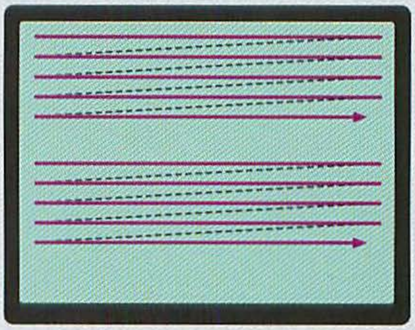
8 A second polarizing filter blocks light that is not vibrating in a plane aligned with the filter. Light that was rotated through a liquid crystal cell with a full voltage applied passes completely through. Light that passed through a cell with no charge and was therefore not rotated is completely blocked. Light that was rotated partially will be partially passed, providing a lesser amount of red, green, or blue (or gray if grayscale). As with a cathode-ray tube monitor, the three colors in close proximity appear to the eye to blend to form another color.



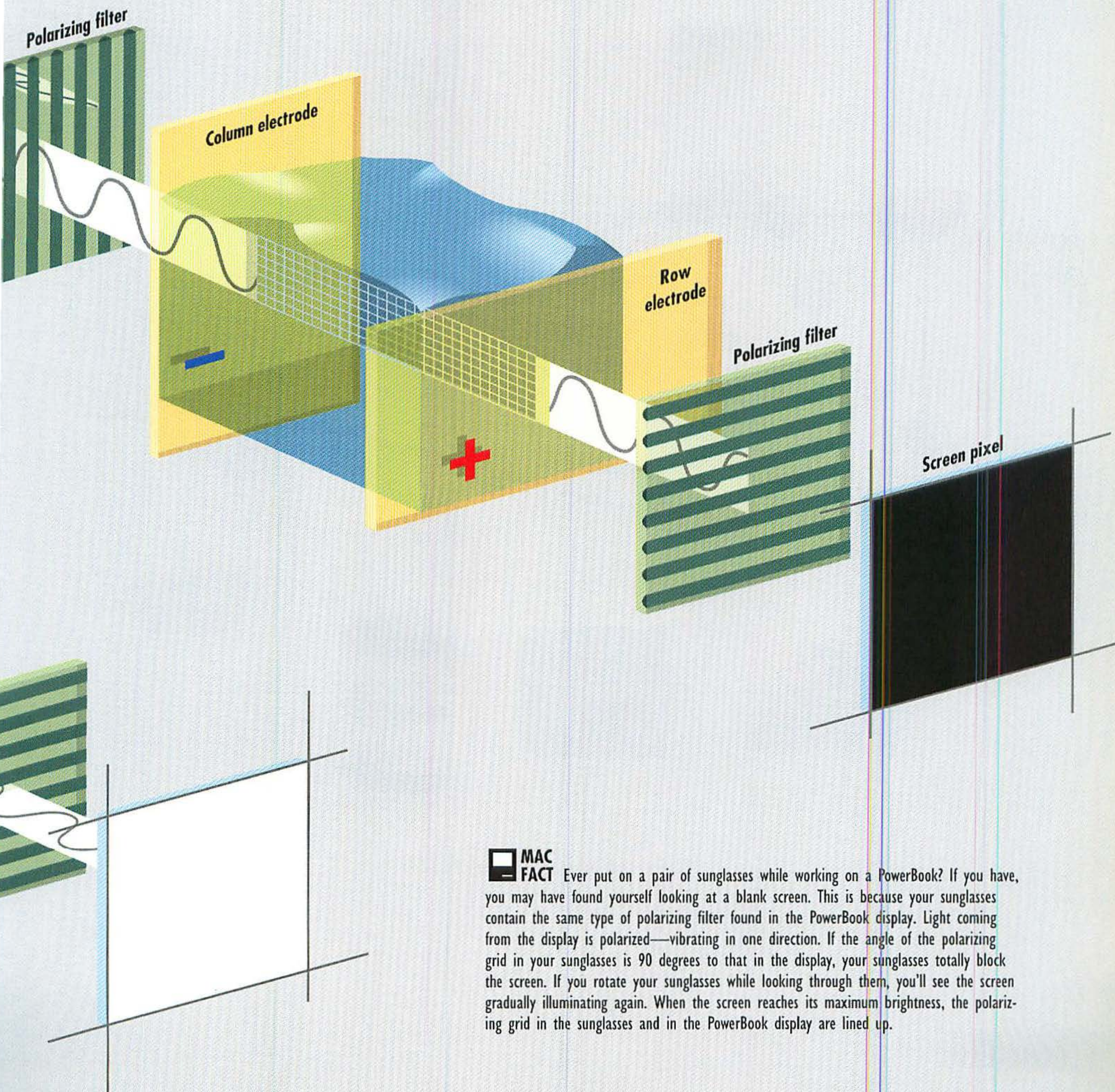
Passive Matrix LCDs

Passive matrix LCDs are simpler than active matrix, but work similarly. Instead of being controlled by a transistor, the voltage for each pixel is controlled from the logic board's video circuitry. Shown here is a monochrome display, which has only two voltages and therefore two levels of light twisting: on and off, which produce black or white.

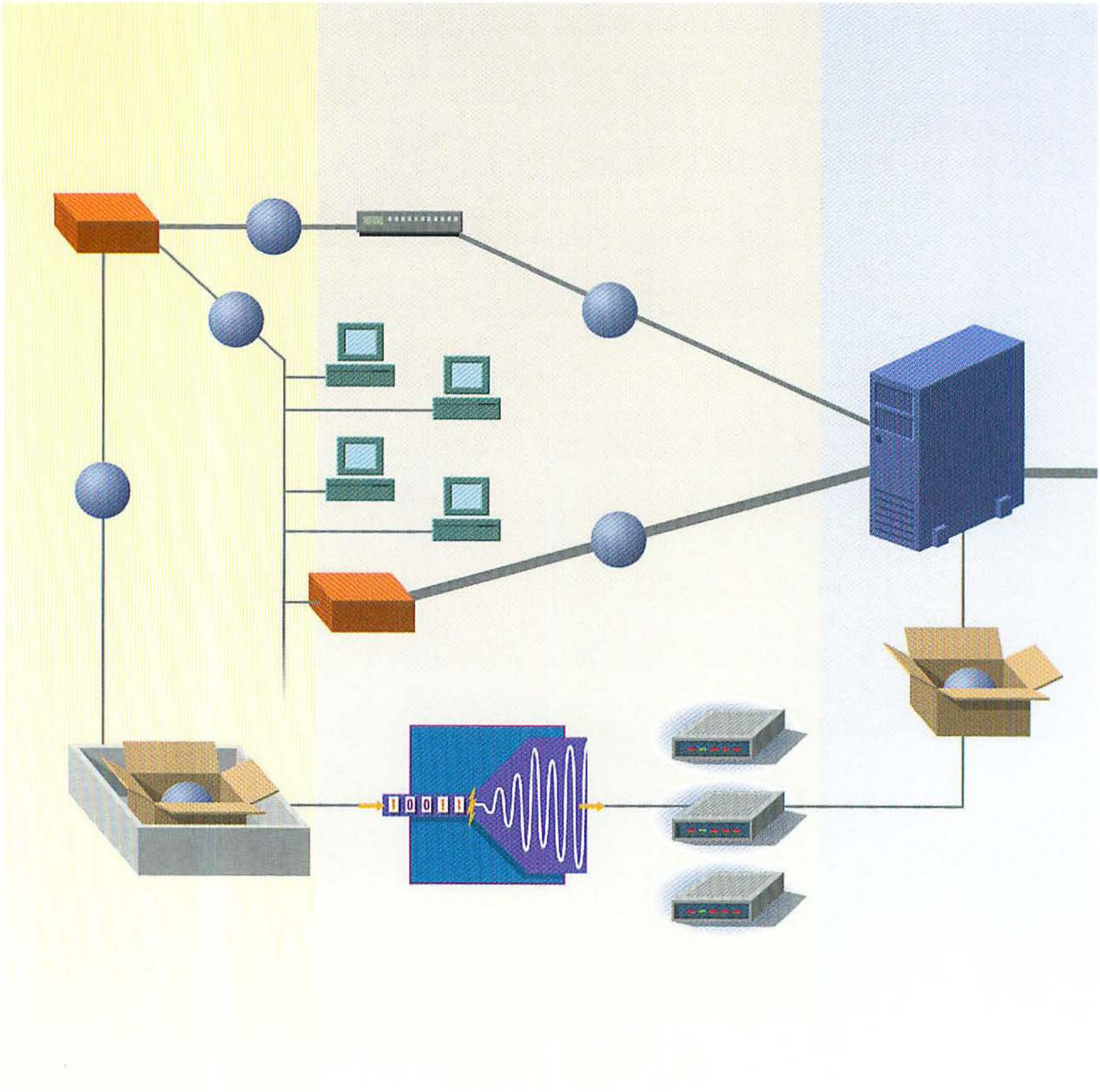




Dual-scan passive matrix One type of passive matrix display called a *dual-scan screen* produces faster performance and brighter colors. It does this by scanning the top and bottom halves of the screen simultaneously.



MAC FACT Ever put on a pair of sunglasses while working on a PowerBook? If you have, you may have found yourself looking at a blank screen. This is because your sunglasses contain the same type of polarizing filter found in the PowerBook display. Light coming from the display is polarized—vibrating in one direction. If the angle of the polarizing grid in your sunglasses is 90 degrees to that in the display, your sunglasses totally block the screen. If you rotate your sunglasses while looking through them, you'll see the screen gradually illuminating again. When the screen reaches its maximum brightness, the polarizing grid in the sunglasses and in the PowerBook display are lined up.



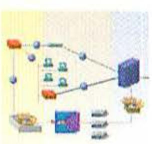
P A R T

7

NETWORKS

WITH computer technology being such a precise field, it is surprising how ambiguously the term *network* is used. Sometimes it's used to describe the type of cable connecting network devices, and other times it's used to describe the hardware circuits generating signals to run over the cable. A network may also be described in terms of the protocols that enable different devices to understand one another. Actually, a network is a system containing all these things, as well as computers, printers, servers, and other devices acting together to improve the usefulness of all the devices.

Networks bring to your Mac the benefits of shared resources. For instance, electronic mail and file servers enable you to send messages and files to people when they aren't at their desks, and electronic calendars let you set up meetings and appointments with people you haven't seen for weeks. These and other network resources, called network services, consist of software running on dedicated machines called *servers* or on other users' Macs. Servers can be Macs, PCs, or mainframe computers in the next room or the next state. When you directly access another user's Mac (or a PC or a network printer), this is called *peer-to-peer networking*. Mac OS comes with software to let you access both dedicated servers and peer-to-peer network devices, as well as to let other users access your Mac on a peer-to-peer basis.



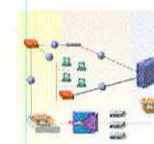
Running underneath all the file, e-mail, and other network services is the mechanism that moves the bits between computers. These are the network protocols that make up the language that computers on a network use to speak to each other.

AppleTalk was the first networking protocol built into Macs and most network printers. In fact, AppleTalk began as a method of sharing laser printers among multiple Mac users. Prior to the LaserWriter, a shared printer had to be directly connected to an individual computer on a network or to a central control box, which was connected to personal computers via slow serial links. Making a printer an independent network device gave it the flexibility to be located anywhere, with communciation speeds several times faster than with serial connections.

AppleTalk is designed to be as easy to set up and use as the Mac. Everything you need to set up an AppleTalk network is built into the Mac: the hardware that transmits and receives the signals, the protocols that enable communications, and the software that enables you to print and share files. You use the Chooser utility to access printers and file servers. System 7's file sharing feature lets other network users access designated files and folders on your hard disk.

AppleTalk is a versatile network, running on a variety of cabling schemes and network interface hardwares running at various speeds. It can run at 230.4 kilobits per second over LocalTalk hardware, or at 100 megabits per second over Fast Ethernet hardware. You can also establish AppleTalk links over a telephone.

The network originally designed for printers can now connect Macs to computers and networks all over the world. Gateways can connect AppleTalk networks to networks running other types of protocols, such as *TCP/IP* (Transmission Control Protocol/Internet Protocol), which is popular in government, academic, and large corporate network circles and is the protocol used on the Internet. Macs can also be put directly on networks running TCP/IP and the *IPX* (Internetwork Packet Exchange) protocol used in Novell NetWare networks, which are popular at sites with PCs. When you use your Mac to log onto the Internet or an IPX server, AppleTalk sits idle while you use TCP/IP or IPX protocol software to give you access to services running on PCs, UNIX machines, and mainframe computers.

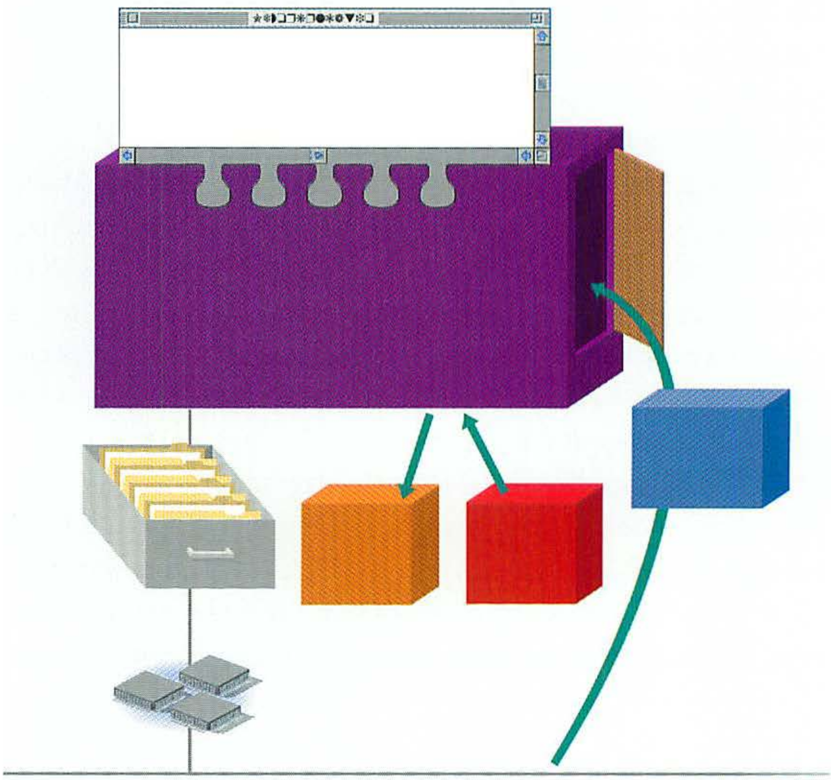


Because of the importance of the Internet and rise of Macs in PC environments, AppleTalk is no longer the core of Mac OS networking, and Mac users have other options for networking. The TCP/IP protocol software now ships with every Mac, and IPX protocol software comes with several Mac networking applications. The Mac became even more of a multiprotocol machine with the arrival of *Open Transport*, a completely new version of Mac OS network system software. Open Transport removed the Mac networking focus from AppleTalk and gave Mac OS equivalent access to TCP/IP and IPX networks. This speeds up the Mac when on non-AppleTalk networks. Open Transport also better conforms to industry networking standards and is designed to let you plug in software to enable the Mac to access other network systems as well.

Despite the decentralization of AppleTalk in Open Transport, AppleTalk isn't going away. It is still the easiest, cheapest, most "plug-and-play" network you can put together.

CHAPTER 27

How Open Transport Works



NETWORKING is as intrinsic a part of the Mac as any of the other input/output systems. As with the Mac's other data transport systems, the mechanics of networking are run by system software that takes its orders from your applications. The most recent form of network-enabling system software that connects network applications to networks is called *Open Transport* and was introduced in 1995. There are still Macs that run previous versions of networking software, including all Macs with 680x0 processors. However, Open Transport will be the standard Mac OS networking software in Mac OS 8.

Before Open Transport, AppleTalk was the Mac's native networking language. At its core was the AppleTalk Manager in ROM and AppleTalk protocol software in the System file and folder. There's nothing wrong with AppleTalk, but it isn't the only network protocol in the world. For instance, the Internet doesn't use it. You used to be able teach the Mac to speak other networking languages, such as those used by the Internet (TCP/IP) and Novell Netware networks (IPX), but the Mac spoke them like a first-year student—slowly and with limited ability.

Open Transport enables the Mac to speak non-AppleTalk network languages like a native. It also speeds up access to all networks, including AppleTalk; adds more features; and makes it easier for developers to create non-AppleTalk networking software.

Open Transport's speed improvements over earlier system software are mostly due to the fact that it is Power Mac native. Prior to Open Transport, the Mac's networking system software ran in the slow 68040 emulation mode. For AppleTalk, applications written to take advantage of Open Transport receive the speed benefits, but older applications don't. However, TCP/IP network access is speeded up for both new and old applications.

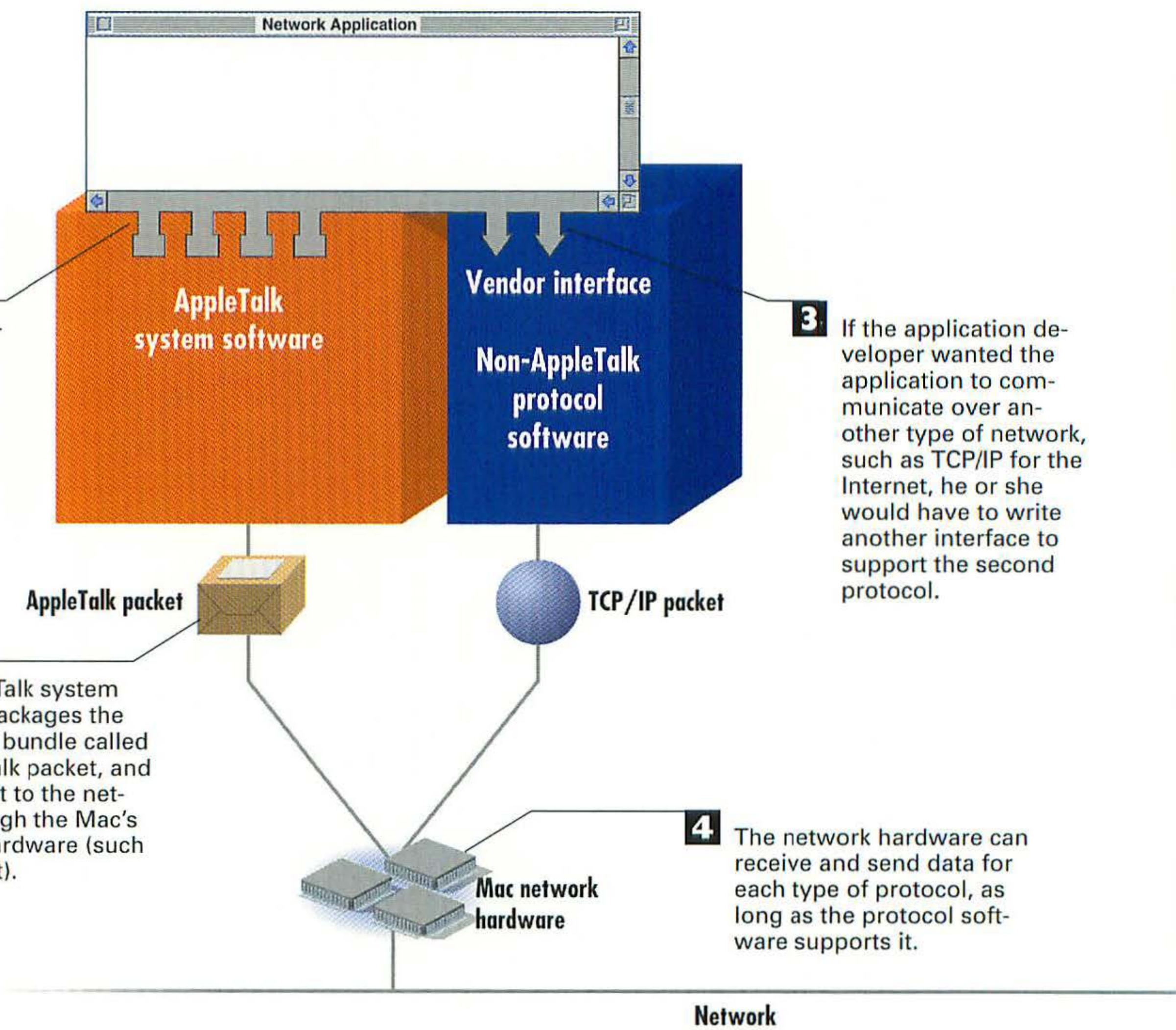
Open Transport's ability to speak multiple network languages fluently is called *transport independence.* Network applications used to be designed specifically for AppleTalk. Any additional networks had to be added. Applications written to support Open Transport don't have to know anything about protocols. They can automatically use AppleTalk, TCP/IP, and IPX, as well as different types of links, including serial, dial-up networks, local area networks, and wide area networks. This expands the usefulness of your network applications.

Open Transport also makes it easy to switch between different network setups. A PowerBook user can store Internet settings for several different locations and change them without having to restart the computer. And instead of loading all of the network protocol software on the fly, Open Transport loads them as you need them, conserving RAM.

In spite of all of these benefits, Open Transport does have some blemishes on its record. When it first shipped with the Power Mac 7200, 7500, 8500, and 9500, the software had the flavor of a wine shipped before its time. The first versions were quirky and had some bugs, particularly with existing versions of some TCP/IP Internet software. However, 1996's version 1.1 cleaned up most of these problems.

Network System Software

Before Open Transport



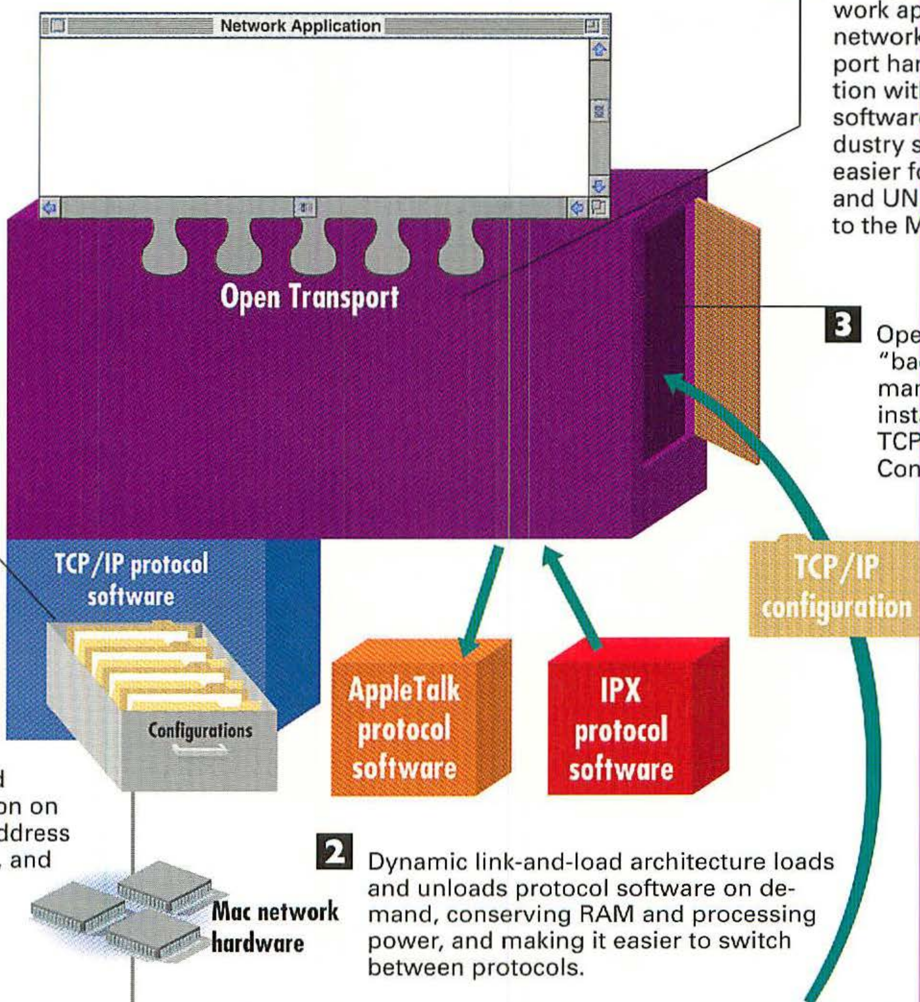
1 A networking application communicates with the Mac's networking system software through an interface that is specifically designed for AppleTalk.

2 The AppleTalk system software packages the data into a bundle called an AppleTalk packet, and sends it out to the network through the Mac's network hardware (such as Ethernet).

3 If the application developer wanted the application to communicate over another type of network, such as TCP/IP for the Internet, he or she would have to write another interface to support the second protocol.

4 The network hardware can receive and send data for each type of protocol, as long as the protocol software supports it.

With Open Transport



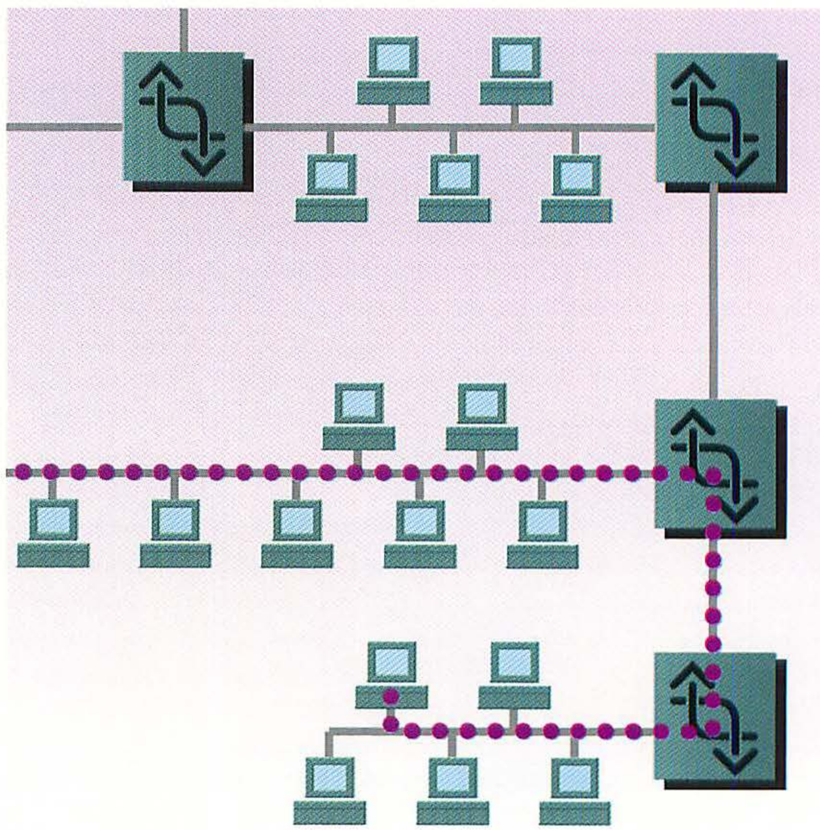
1 A single interface gives the network application access to all network protocols. Open transport handles the communication with the network protocol software. The interface is an industry standard, which makes it easier for developers to port PC and UNIX networking software to the Mac.

2 Dynamic link-and-load architecture loads and unloads protocol software on demand, conserving RAM and processing power, and making it easier to switch between protocols.

3 Open Transport has a "back door" for network management setup. For instance, it supports TCP/IP's Dynamic Host Configuration, which allows administrators to set up a user's TCP/IP information from a server.

4 Open Transport allows you to keep multiple configurations, which you can switch between on the fly. For an Internet setup, these could contain information on different TCP/IP address numbers, servers, and domains.

CHAPTER 28 How Networks Work



WIRES are wires and bits are bits. Some people may find it surprising to access a modem over a network or to see a Mac and PC on the same Ethernet segment, but the wires and bits don't really care. It's the same with *protocols*, the network languages that computers speak. If the networking system software on the computers can handle the data, the networking hardware usually can, too.

Network hardware starts with the transceiver chips in the computers, servers, printers, and other network devices. The most basic rule is that all the devices on the network use the same type of network interface hardware so that they all understand the signals that are being bounced through the wires. The different types of network interfaces, sometimes called *data links*, operate at different speeds and transmit data in different formats. LocalTalk, the original network interface built into every Mac and most network printers, works by running AppleTalk protocols exclusively. *Ethernet* can be used with any network protocol, as can other data links, including token ring, Fast Ethernet, and the long-distance ISDN.

The other pieces of network hardware that bring computers together include *hubs*, or *concentrators*, which boost the network signals traveling along the wires. *Bridges* and *routers* connect different network segments, even if they are using different data links, such as LocalTalk and Ethernet. Networks containing multiple segments are called *internetworks*. This is where the global Internet, the biggest network of networks, gets its name.

Most computer networks, whether they are running AppleTalk, TCP/IP, or other protocols, transmit information in a similar manner. A computer broadcasts a message over the wires in the form of data bundles called *packets*, which contain the addresses of the intended recipients. All of the computers on the network receive the packets, but only the intended recipients choose to read the message.

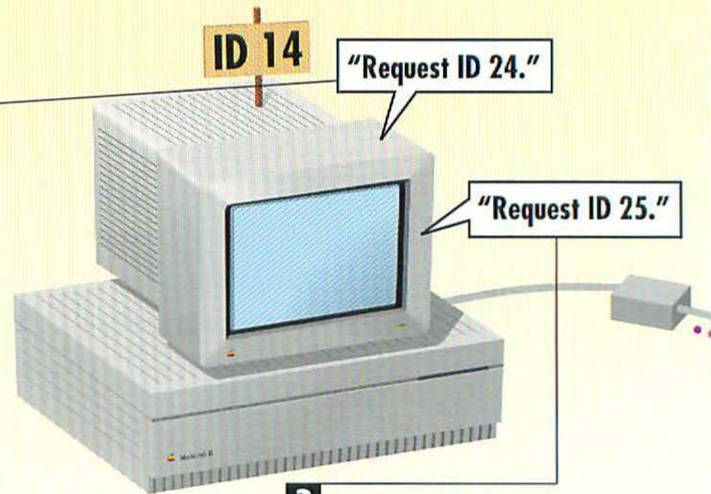
AppleTalk is still the most popular network protocol among Mac users because it is the easiest type of network to set up and use. For instance, a feature called *dynamic addressing* causes the Macs on the network to set their own addresses. You never see this happening—in fact, you never see an AppleTalk address at all.

Other AppleTalk ease-of-use features are designed to hide the more technical aspects of networking from the user. When you use the Chooser to pick a printer type and the specific printer to print to, you're actually installing a device driver. The vast majority of network laser printers use the LaserWriter driver, though you can access special features of a printer by using a specific driver for a particular printer.

System 7's file sharing enables you to pass files between Macs and PCs that have software that is compatible with the *AppleTalk Filing Protocol* (AFP). Networked hard disks and folders sitting on other users' Macs or on AFP-compatible servers all appear as desktop volumes, much like the hard disk in your Mac.

AppleTalk at Work

1 When a Mac starts up, it randomly chooses a *node number*, which is like a street address. The Mac sends out a message asking if other nodes have already chosen the same node number.

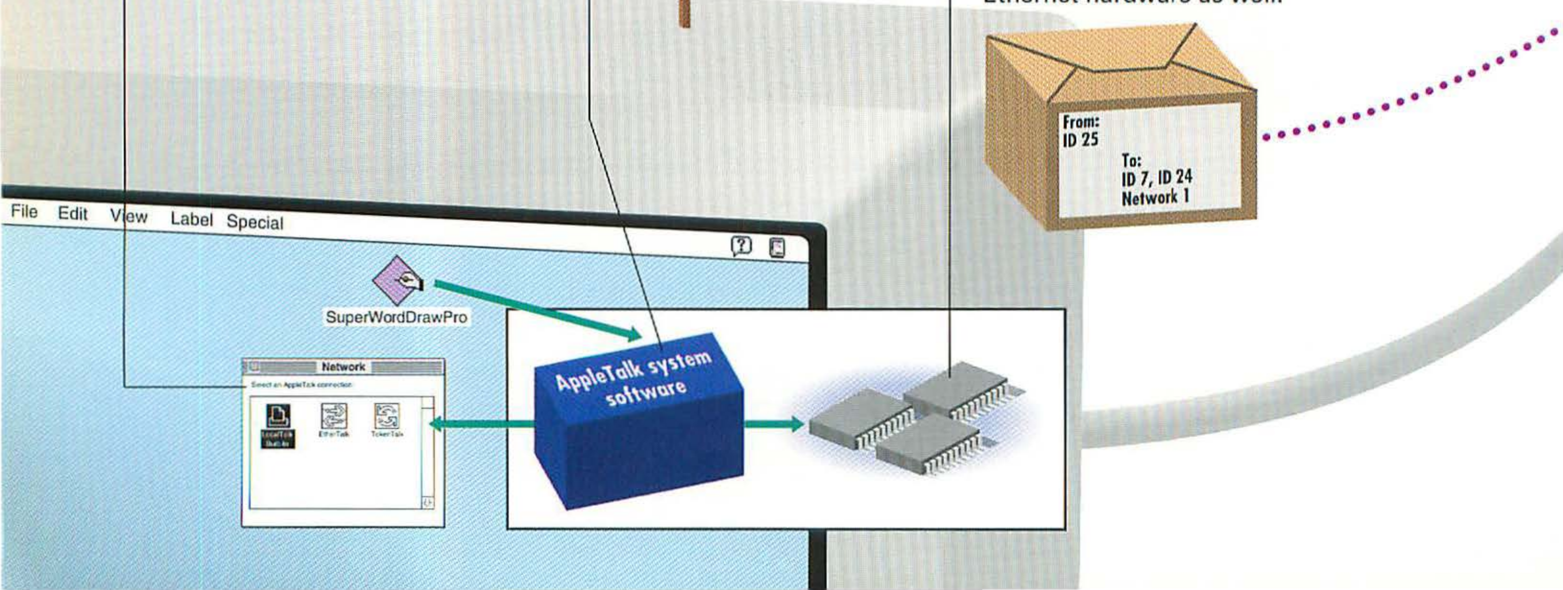


3 The Mac joining the network chooses another number, and repeats the process until no other device reports back that the number is in use.

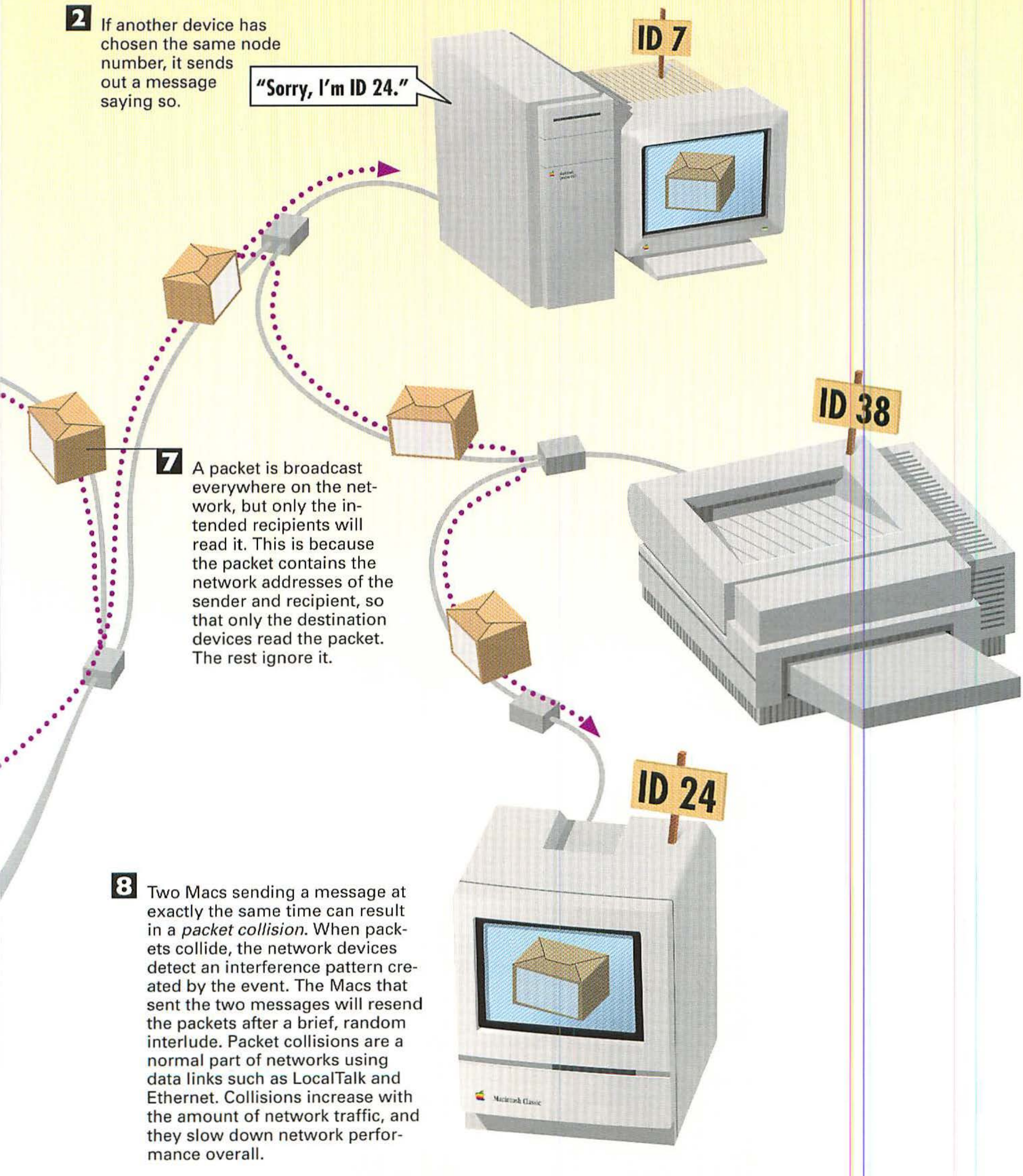
4 When you make contact with another device on the network, the application you are using makes calls to the AppleTalk system software.

5 The AppleTalk system software checks to see which data-link software has been chosen by the user in the Network control panel or the AppleTalk control panel in Open Transport. You can select LocalTalk to use the Mac's built-in LocalTalk hardware, or you can select EtherTalk for built-in or add-in Ethernet hardware. You can also choose the TokenTalk data link for token-ring hardware, as well as other data links you might have installed. However, all devices on a network must use the same type of network interface hardware.

6 The network interface hardware, either built-in or on an add-in card, sends and receives data over the network in small chunks called *packets* (sometimes referred to as frames). All Macs have LocalTalk (which uses the printer port), and some models have built-in Ethernet hardware as well.



2 If another device has chosen the same node number, it sends out a message saying so.

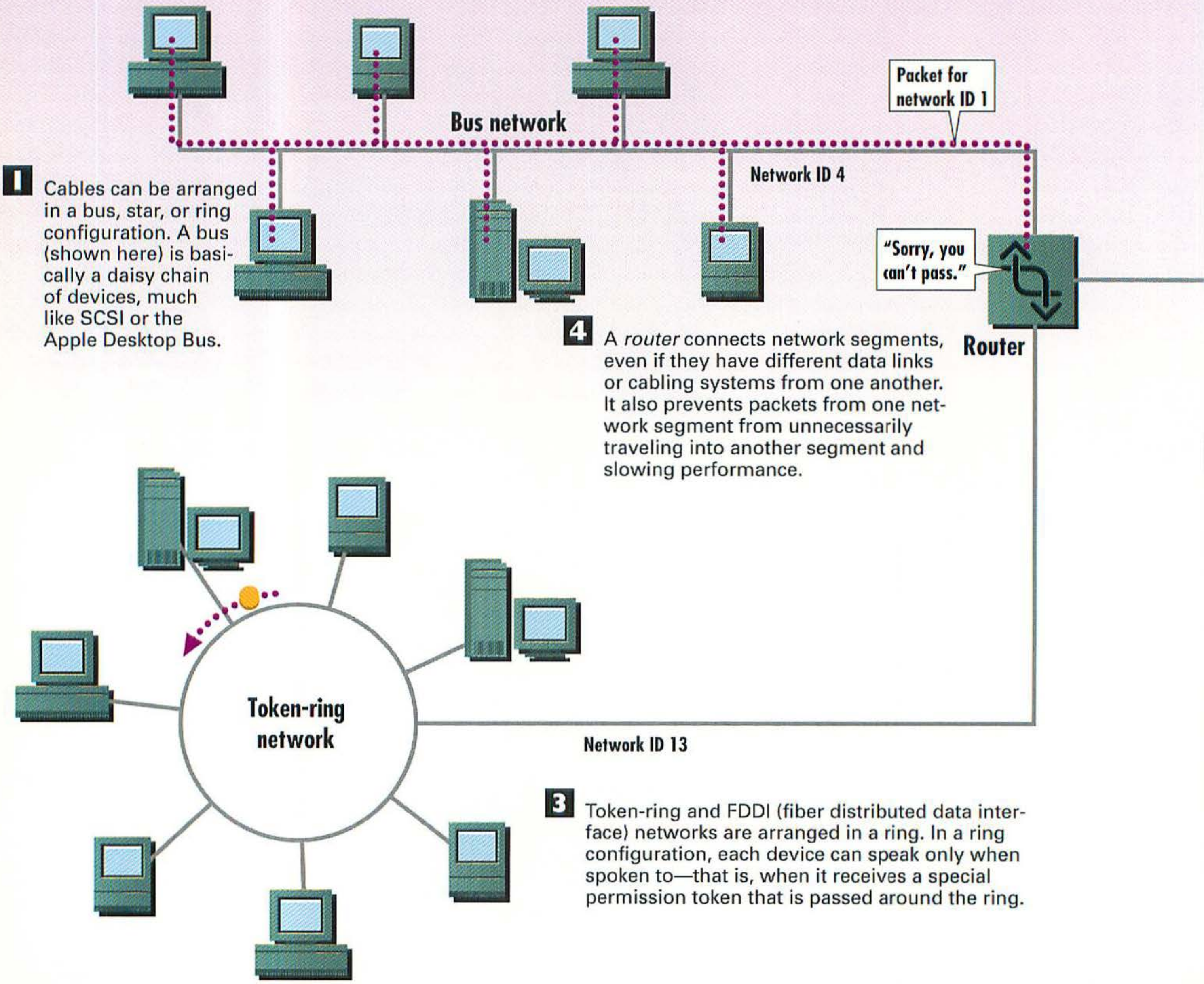


7 A packet is broadcast everywhere on the network, but only the intended recipients will read it. This is because the packet contains the network addresses of the sender and recipient, so that only the destination devices read the packet. The rest ignore it.

8 Two Macs sending a message at exactly the same time can result in a *packet collision*. When packets collide, the network devices detect an interference pattern created by the event. The Macs that sent the two messages will resend the packets after a brief, random interlude. Packet collisions are a normal part of networks using data links such as LocalTalk and Ethernet. Collisions increase with the amount of network traffic, and they slow down network performance overall.

AppleTalk Network Hardware

Small networks of a few Macs don't require anything other than cables and inexpensive connector boxes. When networks grow, several types of equipment are required to keep the network running. One of the most important is a router, which connects two or more networks together to form an internetwork, or divides a network into two network segments. To each segment, a network administrator must assign a network number, which is sort of like a postal zip code; the network number enables the delivery of messages to the proper network segment.

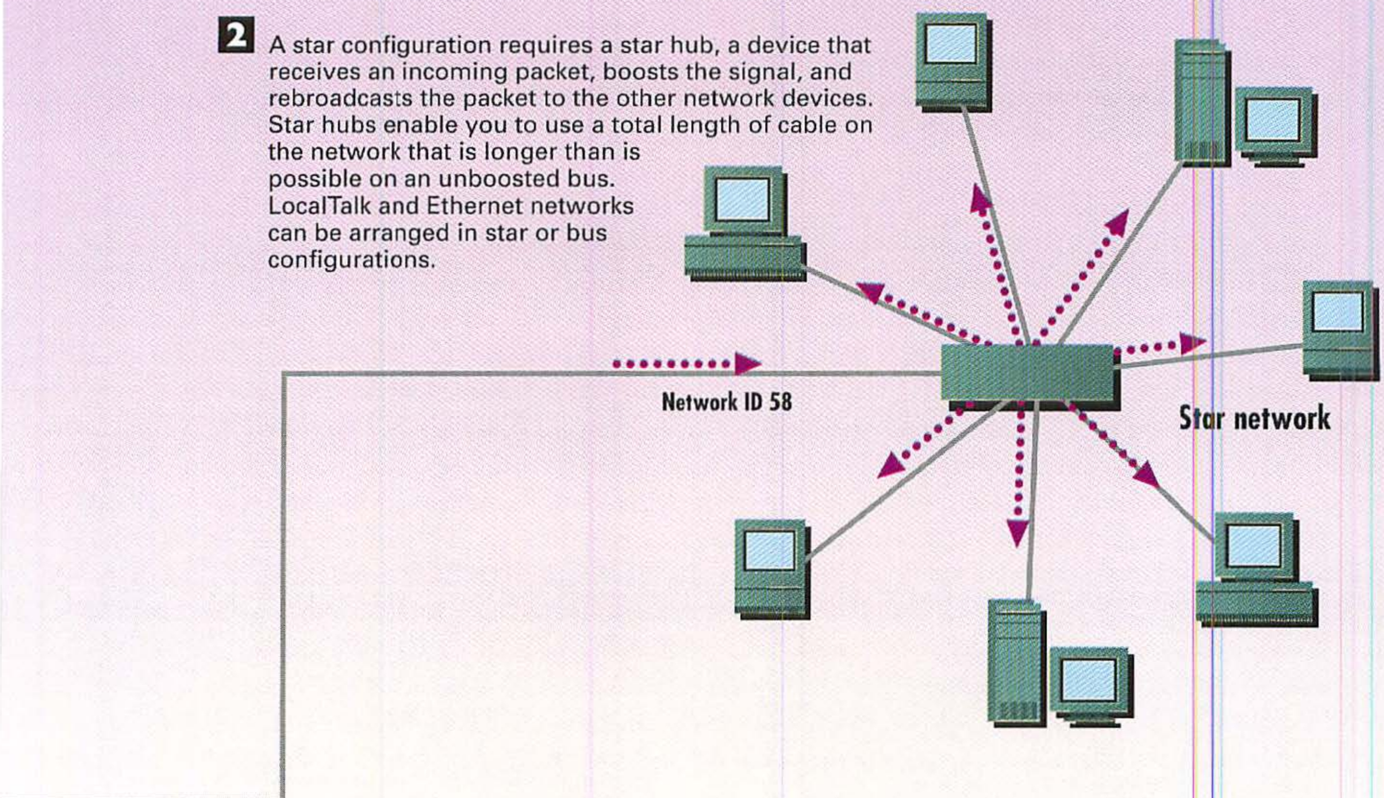


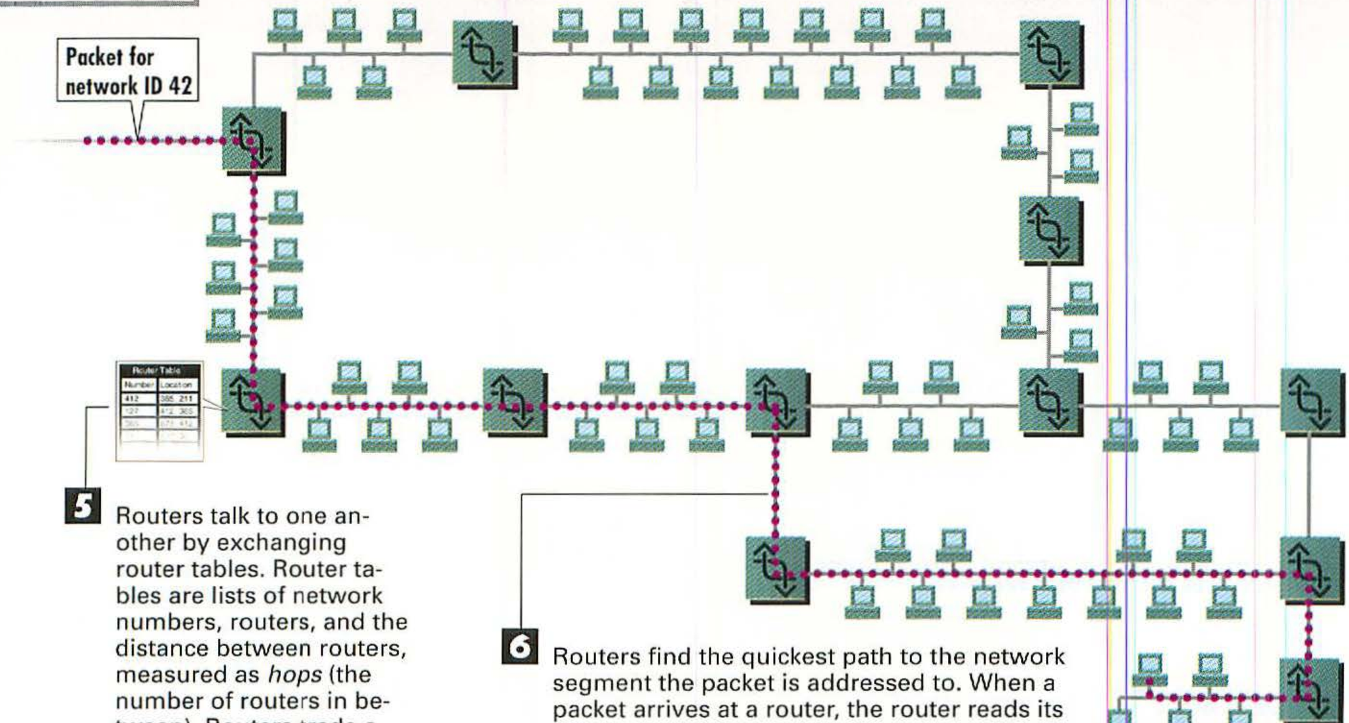
1 Cables can be arranged in a bus, star, or ring configuration. A bus (shown here) is basically a daisy chain of devices, much like SCSI or the Apple Desktop Bus.

4 A *router* connects network segments, even if they have different data links or cabling systems from one another. It also prevents packets from one network segment from unnecessarily traveling into another segment and slowing performance.

3 Token-ring and FDDI (fiber distributed data interface) networks are arranged in a ring. In a ring configuration, each device can speak only when spoken to—that is, when it receives a special permission token that is passed around the ring.

2 A star configuration requires a star hub, a device that receives an incoming packet, boosts the signal, and rebroadcasts the packet to the other network devices. Star hubs enable you to use a total length of cable on the network that is longer than is possible on an unboosted bus. LocalTalk and Ethernet networks can be arranged in star or bus configurations.



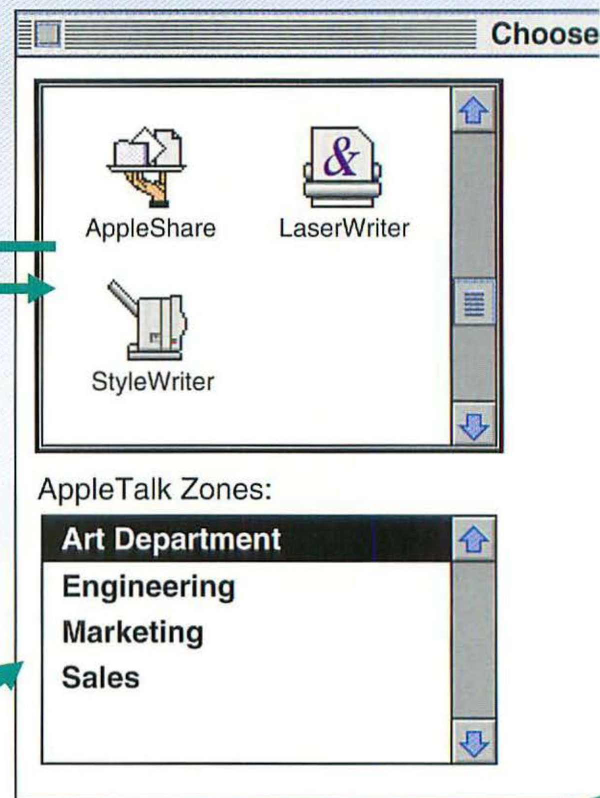


5 Routers talk to one another by exchanging router tables. Router tables are lists of network numbers, routers, and the distance between routers, measured as *hops* (the number of routers in between). Routers trade a zone information table, which lists network zone names—the names that show up in the Chooser.

6 Routers find the quickest path to the network segment the packet is addressed to. When a packet arrives at a router, the router reads its address to find the network number of the packet's destination. The router looks at its routing table to find the path with the fewest hops. The router sends the packet on its way to the next router along the shortest route.

The Chooser

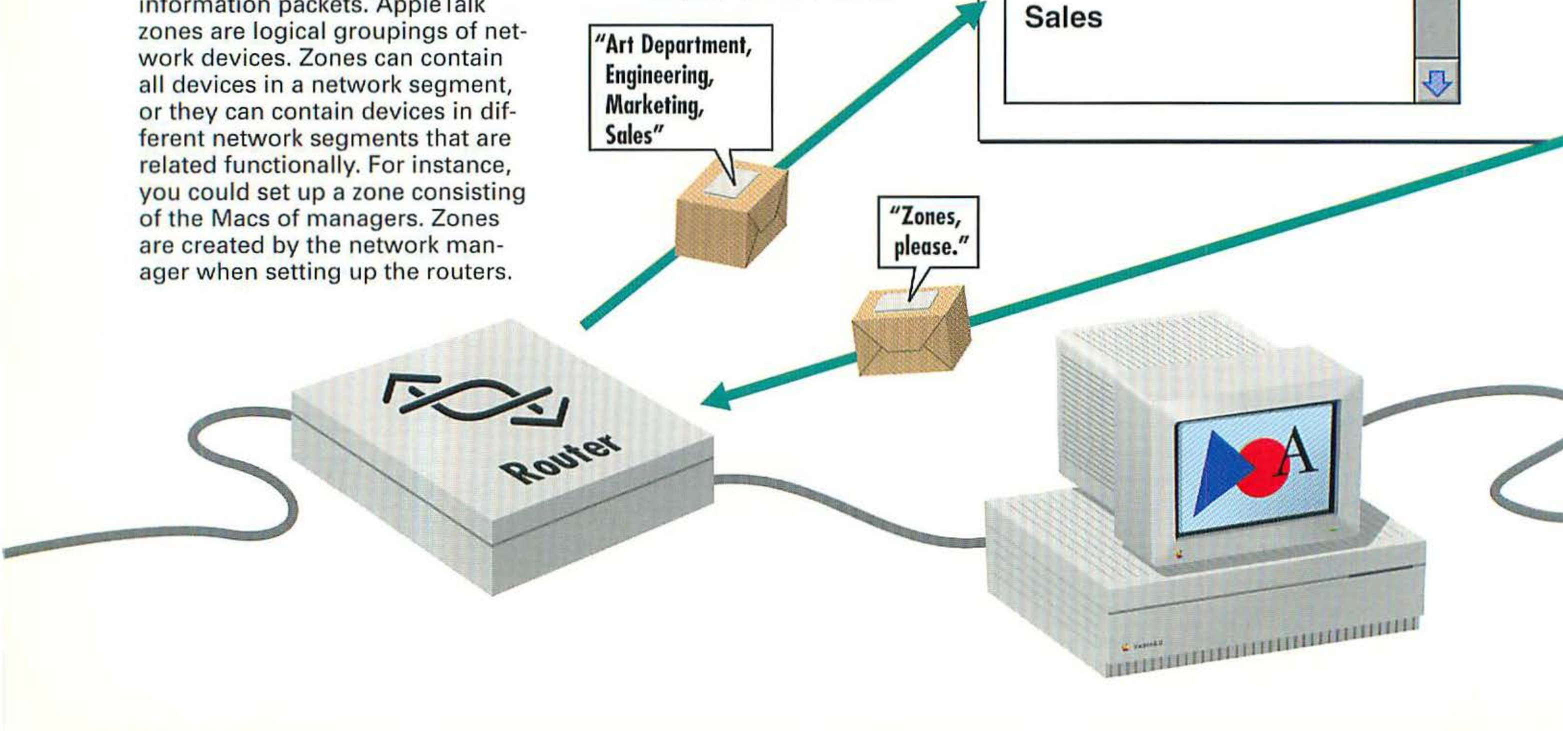
When Apple introduced the LaserWriter in 1985, it needed an easy way to select from among the multiple printers available on an AppleTalk network and those connected directly to your Mac. The result was an Apple Menu utility called Choose Printer. This simple utility eventually moved beyond printing to become today's Chooser, which is a view into an AppleTalk network and is where you make connections to AppleTalk devices.



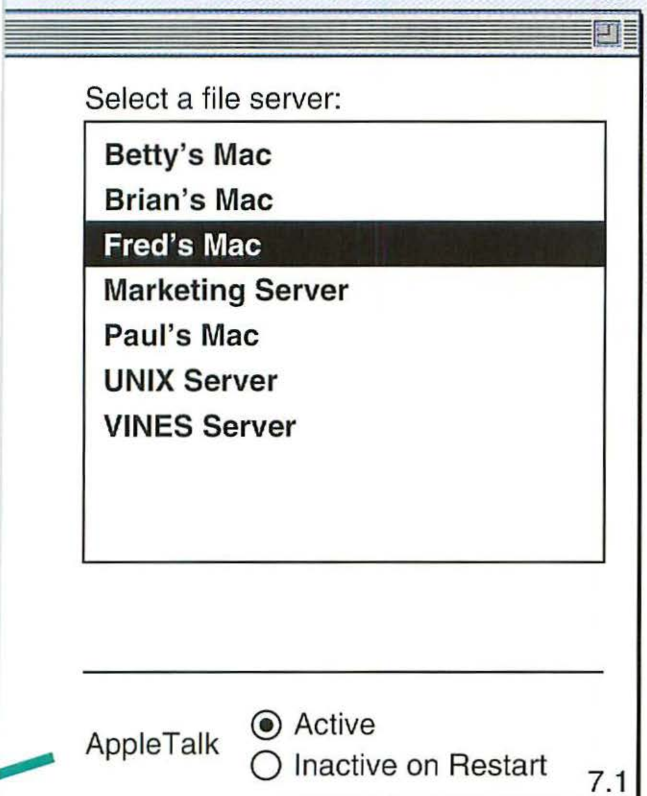
1 When you open the Chooser, it reads the network device drivers, also called Chooser extensions, in the System folder. There is one device driver for each type of printer: laser printer, dot-matrix printer, ink-jet printer, and so on. There is a single extension—the AppleShare driver—for AppleShare-compatible file servers and shared folders. The data in these files tell the Chooser what to display when you click on an icon in the device-driver field and when you choose a device.

2 If AppleTalk is *active* (turned on), the Chooser sends out packets requesting routers to send zone information packets. AppleTalk zones are logical groupings of network devices. Zones can contain all devices in a network segment, or they can contain devices in different network segments that are related functionally. For instance, you could set up a zone consisting of the Macs of managers. Zones are created by the network manager when setting up the routers.

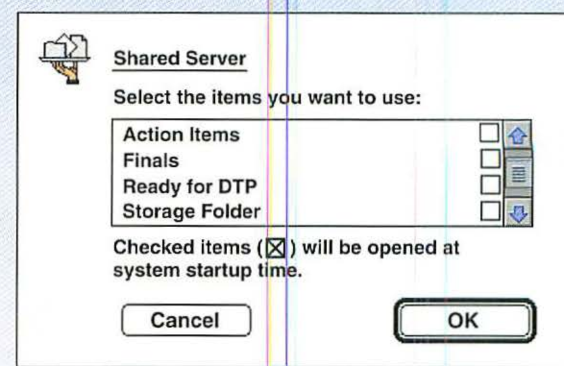
3 The Chooser reads the zone information packets that are sent and displays a list of zones. If no zones are detected, it does not display the zone field.



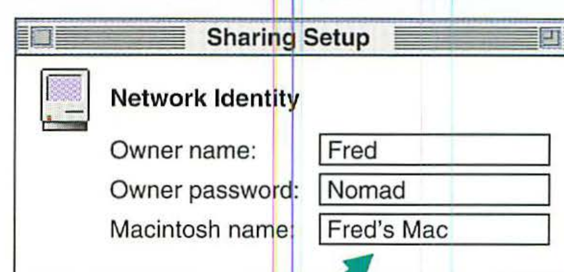
5 The file servers that show up in the device field are any devices using the AppleTalk Filing Protocol (AFP). This includes AppleShare file servers, Macs using System 7's file sharing, or AFP-compatible servers running on PCs or UNIX machines.

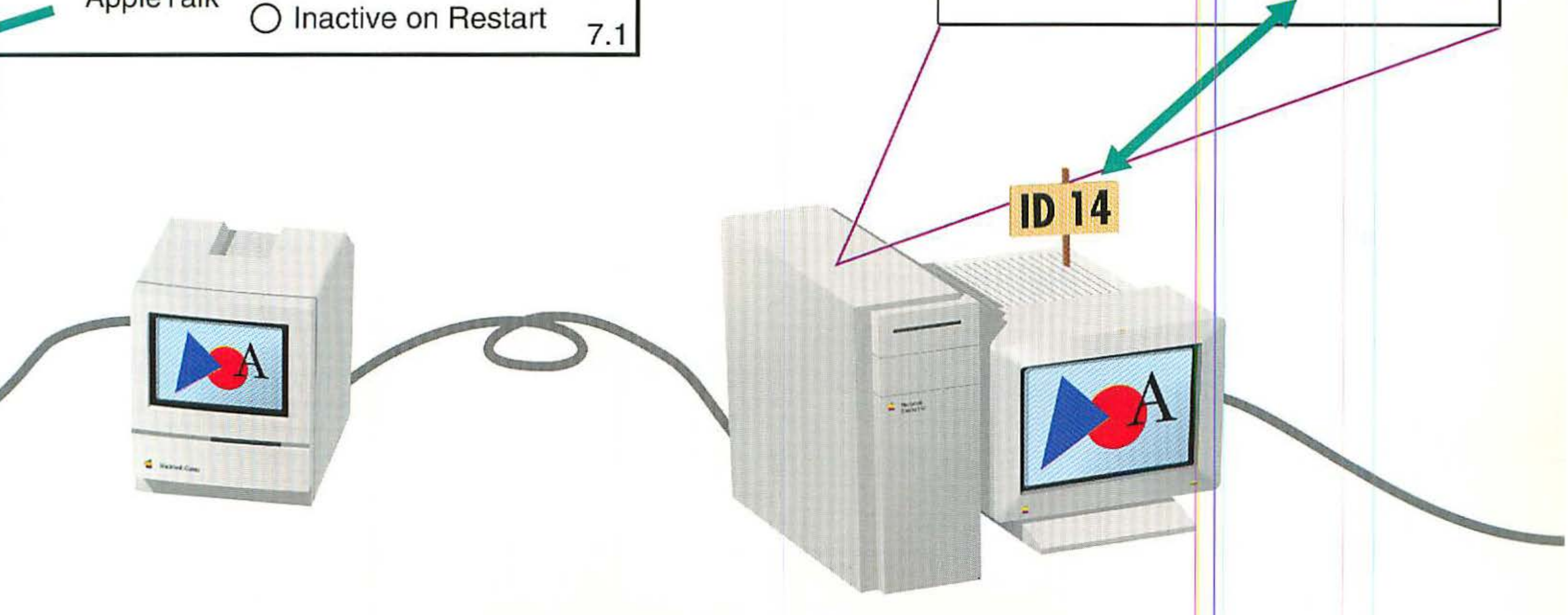


6 When you double-click on a particular file server, the Chooser presents you with a list of volumes (including hard-disk drives and shared folders) connected to that machine. Double-clicking on the volume name opens a link between your Mac and the volume and mounts an icon for the volume on your desktop.



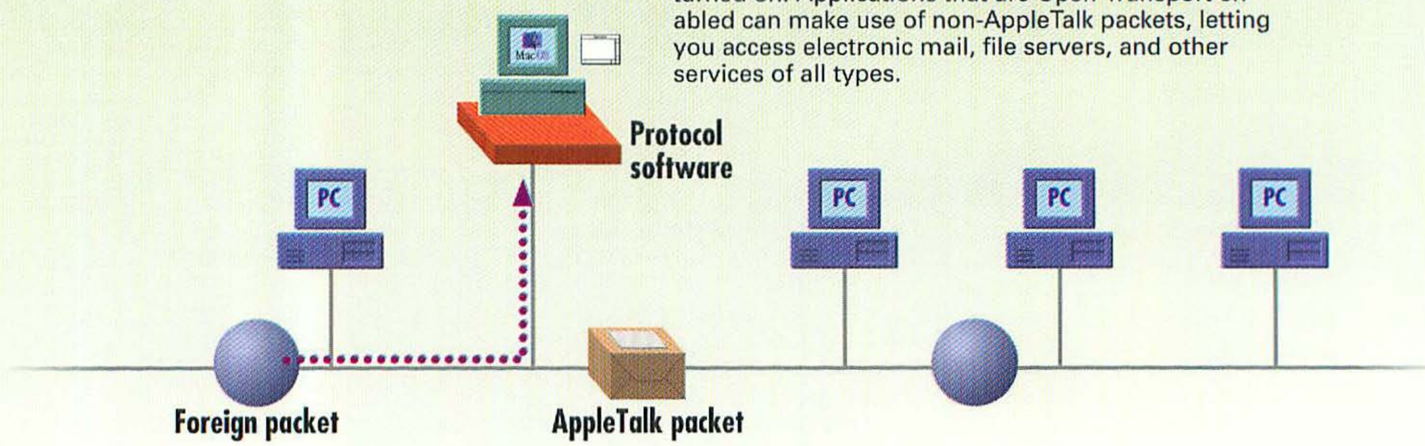
4 When you select the type of network service and a zone to which you are trying to connect, you get list of devices in that zone. In this case, the AppleShare icon has been selected. The Chooser uses AppleTalk's Name Binding Protocol (NBP) to read the node number of the file servers and translate it to the Mac and server names that users have typed in the Macintosh Name field of the Sharing Setup control panel.





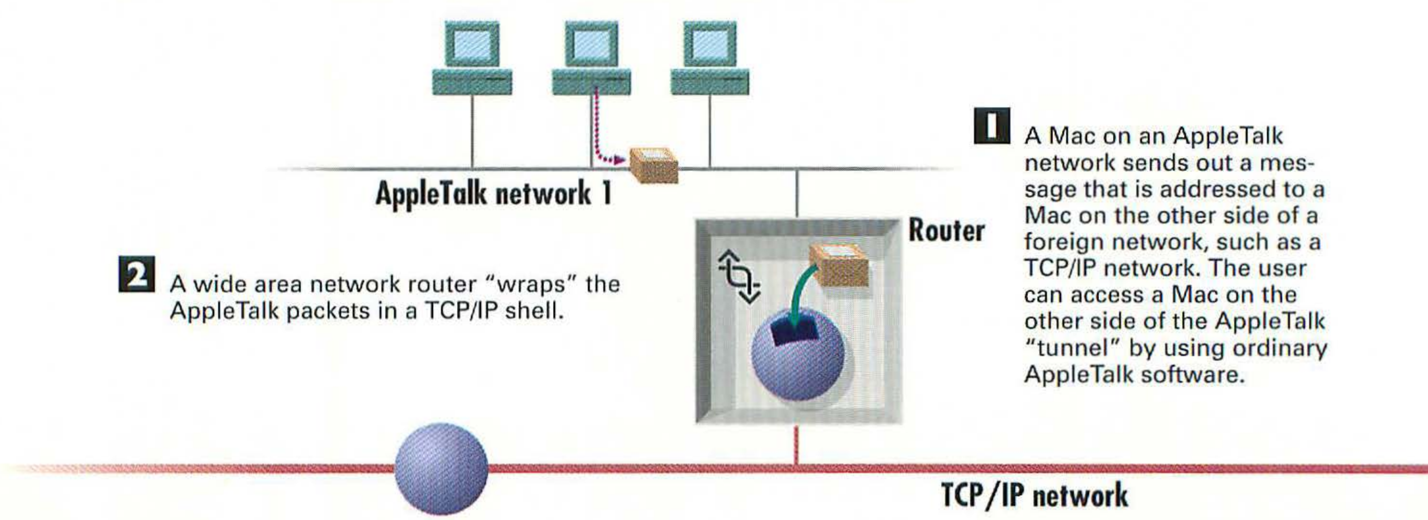
Macs and Non-AppleTalk Networks

Using non-AppleTalk packets Packets of different protocols can travel over the same network wire. A Mac can recognize non-AppleTalk packets if that Mac has the correct protocol software installed and turned on. Applications that are Open Transport enabled can make use of non-AppleTalk packets, letting you access electronic mail, file servers, and other services of all types.



Tunneling

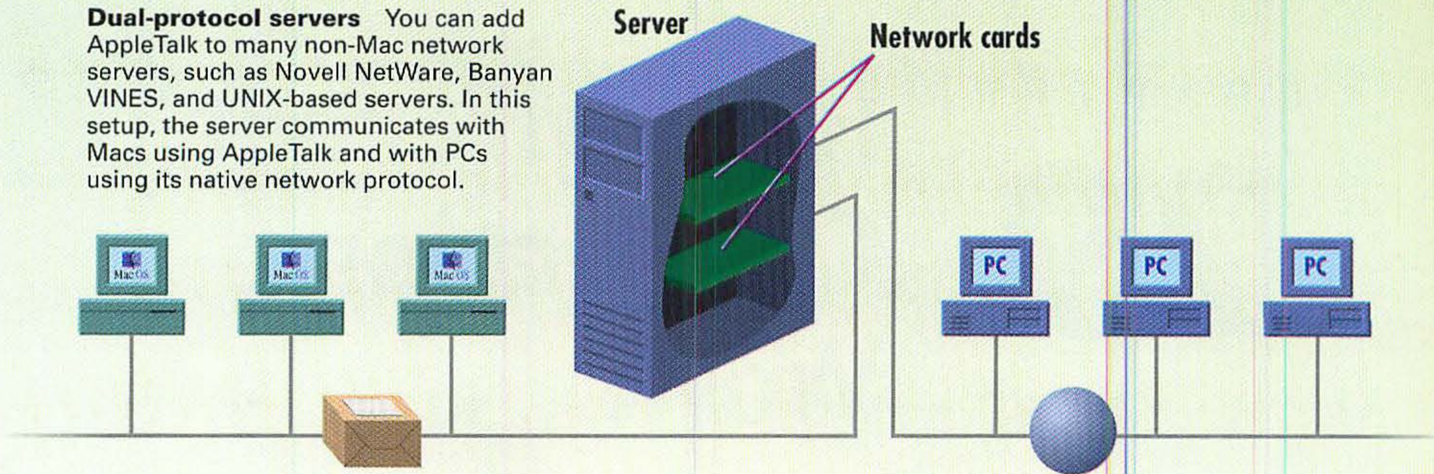
Tunneling (also called encapsulation) is a method that lets two AppleTalk networks connected by one or more non-AppleTalk networks (such as the Internet) communicate with each other. The two networks can be in adjacent buildings or on opposite coasts.



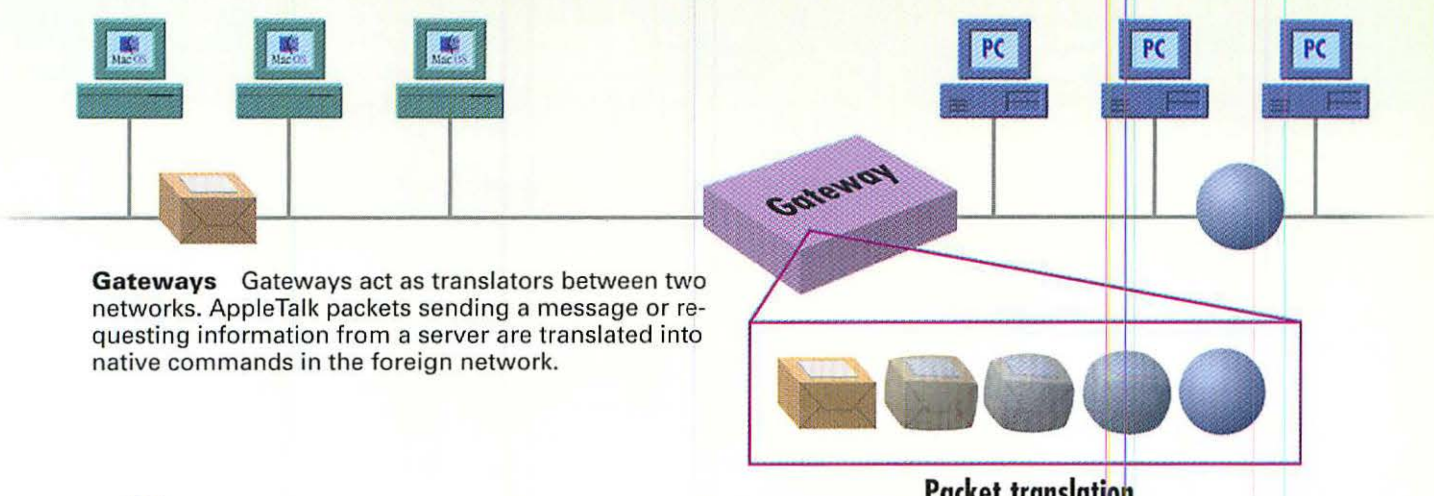
1 A Mac on an AppleTalk network sends out a message that is addressed to a Mac on the other side of a foreign network, such as a TCP/IP network. The user can access a Mac on the other side of the AppleTalk "tunnel" by using ordinary AppleTalk software.

2 A wide area network router "wraps" the AppleTalk packets in a TCP/IP shell.

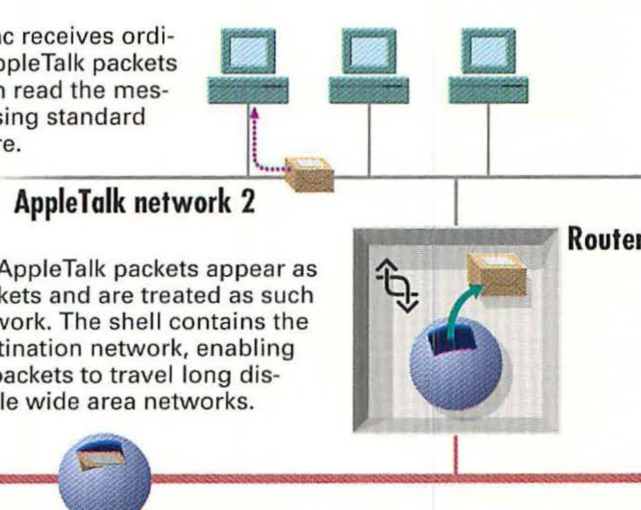
Dual-protocol servers You can add AppleTalk to many non-Mac network servers, such as Novell NetWare, Banyan VINES, and UNIX-based servers. In this setup, the server communicates with Macs using AppleTalk and with PCs using its native network protocol.



Gateways Gateways act as translators between two networks. AppleTalk packets sending a message or requesting information from a server are translated into native commands in the foreign network.



5 The Mac receives ordinary AppleTalk packets and can read the message using standard software.

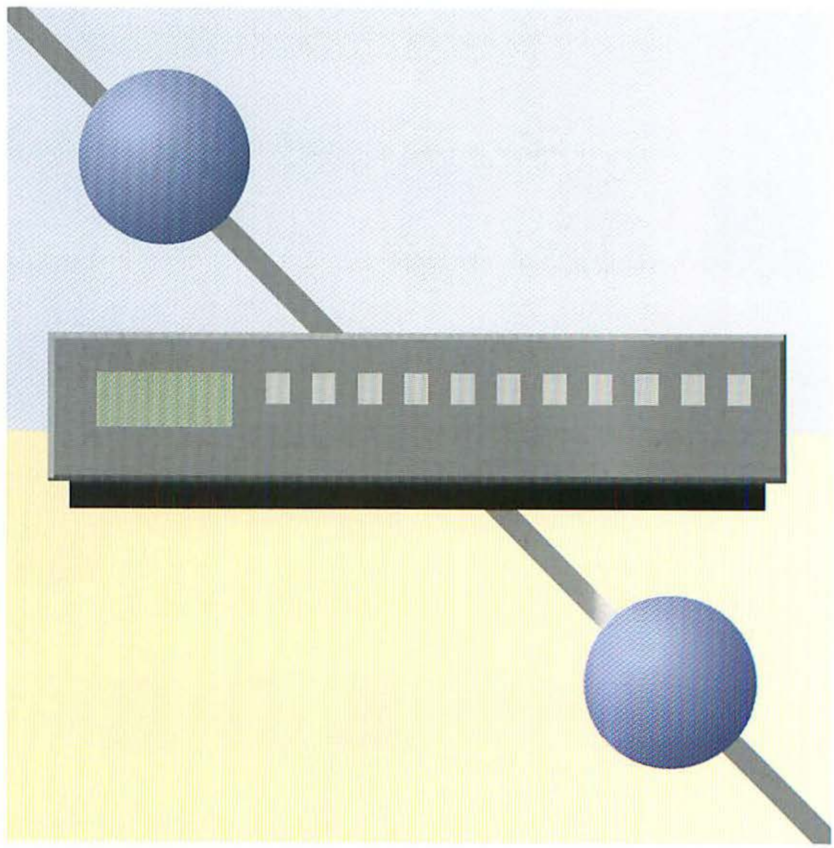


3 The encapsulated AppleTalk packets appear as normal TCP/IP packets and are treated as such by the foreign network. The shell contains the address of the destination network, enabling the encapsulated packets to travel long distances over multiple wide area networks.

4 A router connected to the AppleTalk network on the other side "unwraps" the encapsulated AppleTalk packets and passes them on to the AppleTalk network.

CHAPTER 29

How Internet Connections Work



IT'S hard to pick up a newspaper or magazine without seeing a reference to the Internet, the ever-expanding global network of networks. Millions of people keep in touch using electronic mail over the Internet, and there are an increasing number of services available to you from thousands of Internet servers. Among the most popular types of Internet resources are World Wide Web servers, which offer you a graphical interactive interface to data. The Mac's built-in networking facilities and easy-to-use software have made it a popular machine for accessing the Internet, as well as for serving World Wide Web information and for creating electronic publishing content.

Browsing a World Wide Web site on your Mac is quite a different experience from accessing a database from an AppleShare server, but the underlying mechanism isn't that different. In both situations, your Mac sends and receives commands and data in the form of bundles of bits called network packets. The language your Mac uses to exchange packets with server machines on the Internet is called TCP/IP, short for Transport Control Protocol/Internet Protocol. Like AppleTalk, the network system software needed to enable the Mac to speak TCP/IP comes with every Mac.

There are several different methods of creating Internet links, each with varying costs and connection speeds. Most individuals use an inexpensive modem dial-up connection to the Internet. For an additional monthly fee paid to your telephone company, you can increase the amount of data you can transmit fivefold or more by installing a digital modem over an ISDN line. Businesses often find it economical to connect a network full of computer users to the Internet through a dedicated high-speed line leased from a telephone company.

What all of these methods have in common is a connection to an Internet service provider, which is a company that links one or more of your Macs or PCs to the Internet. A service provider can be a traditional online service such as America Online or CompuServe, or a dedicated Internet access company. Every service provider handles the technical details of making your Mac a TCP/IP node on the Internet. Unlike AppleTalk, TCP/IP is not a plug-and-play network system. Your service provider supplies the network information you need to type into the Mac's TCP control panel. This information identifies your Mac as an entity on the network, and helps guide e-mail, file transfers, and requests to log onto Web sites between your Mac and the millions of other computers on the Internet.

The dramatic increase in popularity of the Internet in recent years has made TCP/IP networking as important to Mac users as AppleTalk. Because of this, we will see more Internet access features added to Mac OS as time goes on, particularly at the application level. As this happens, you can use the Internet to keep up with the changes in Mac OS technology at Apple's World Wide Web site: http://www.apple.com.

Connecting to the Internet

Your Mac

Internet Link

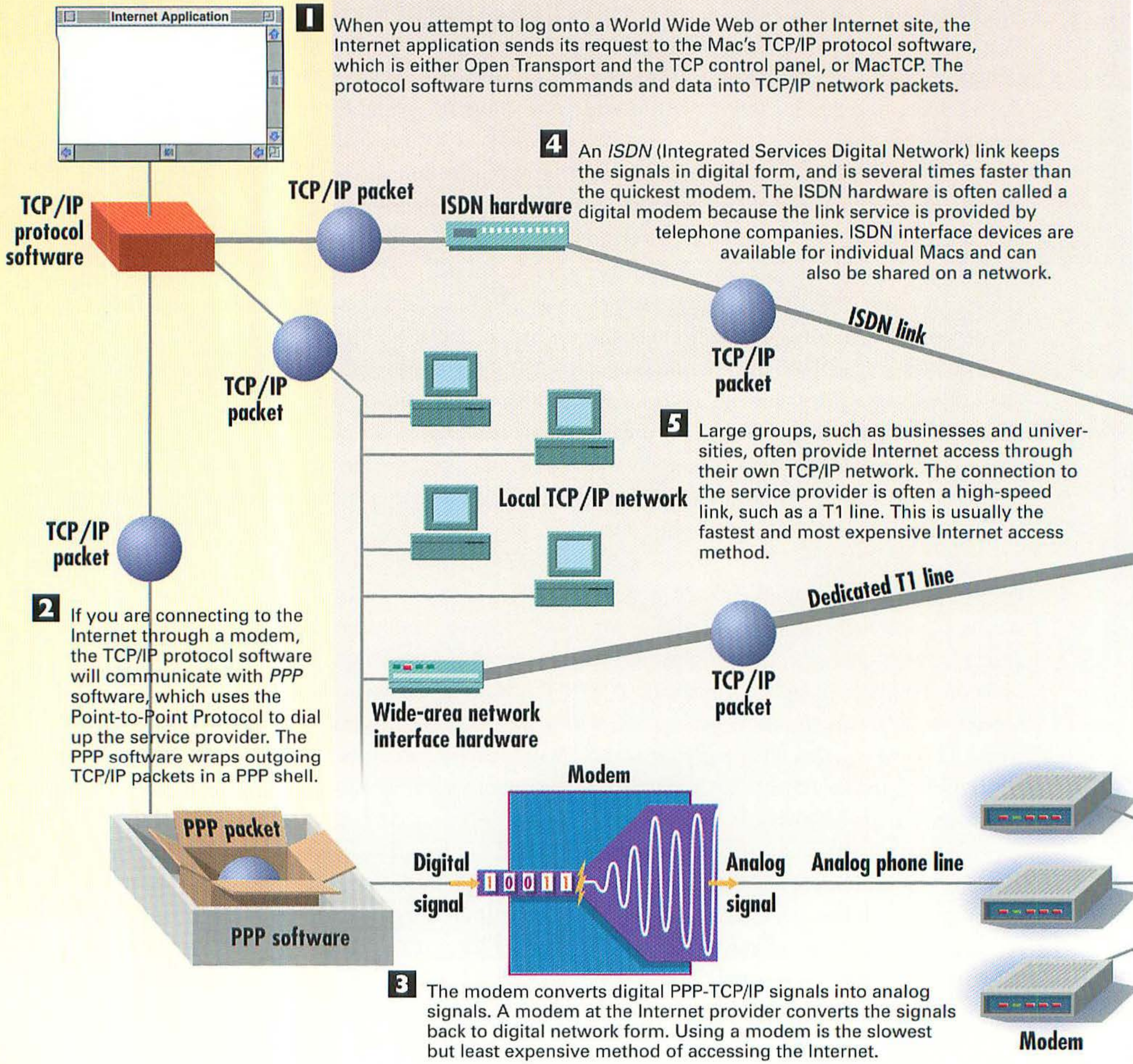
1 When you attempt to log onto a World Wide Web or other Internet site, the Internet application sends its request to the Mac's TCP/IP protocol software, which is either Open Transport and the TCP control panel, or MacTCP. The protocol software turns commands and data into TCP/IP network packets.

2 If you are connecting to the Internet through a modem, the TCP/IP protocol software will communicate with *PPP* software, which uses the Point-to-Point Protocol to dial up the service provider. The PPP software wraps outgoing TCP/IP packets in a PPP shell.

3 The modem converts digital PPP-TCP/IP signals into analog signals. A modem at the Internet provider converts the signals back to digital network form. Using a modem is the slowest but least expensive method of accessing the Internet.

4 An *ISDN* (Integrated Services Digital Network) link keeps the signals in digital form, and is several times faster than the quickest modem. The ISDN hardware is often called a digital modem because the link service is provided by telephone companies. ISDN interface devices are available for individual Macs and can also be shared on a network.

5 Large groups, such as businesses and universities, often provide Internet access through their own TCP/IP network. The connection to the service provider is often a high-speed link, such as a T1 line. This is usually the fastest and most expensive Internet access method.

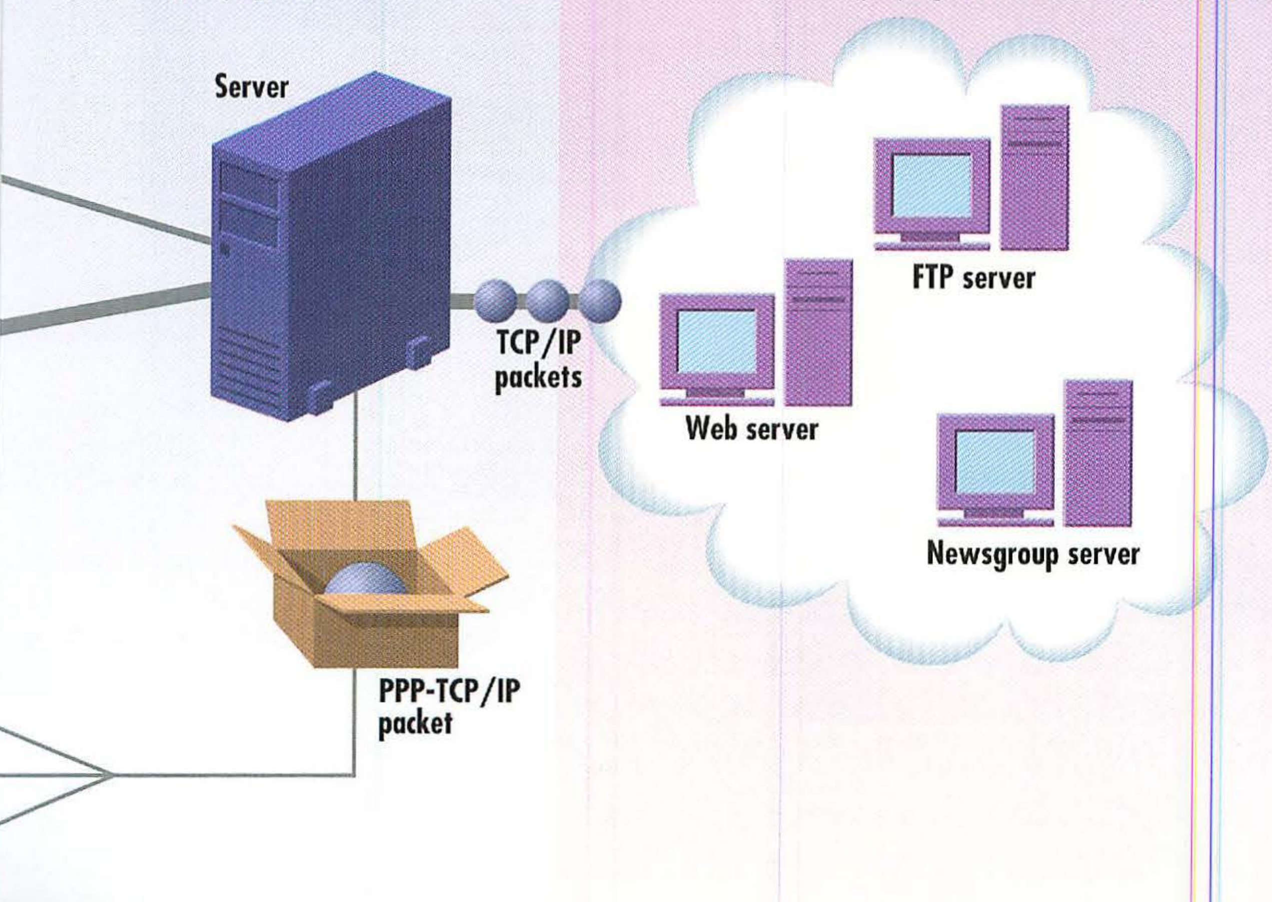


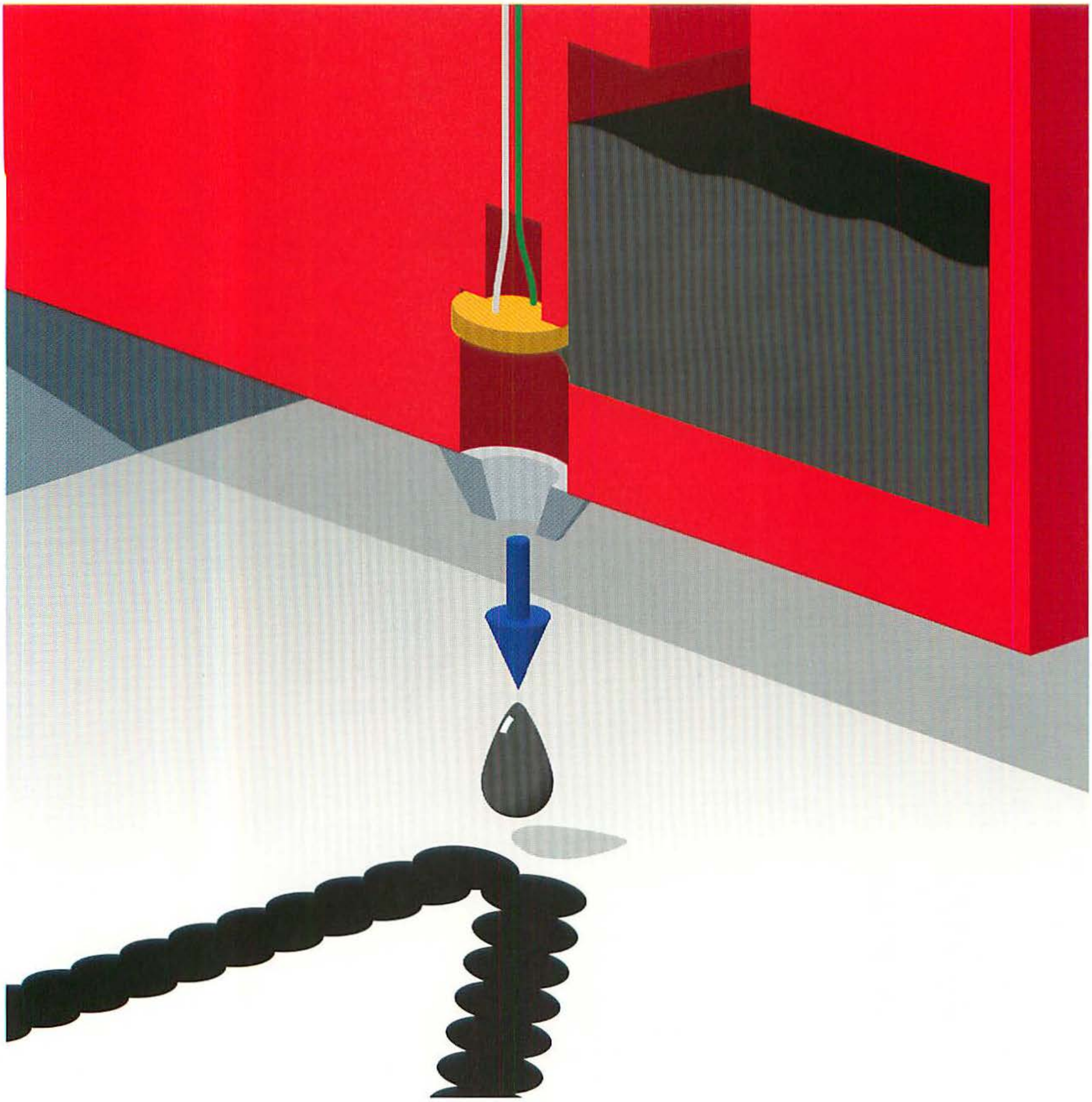
Internet Service Provider

6 The Internet service provider links you or your organization with the Internet by offering various services. The service provider usually gives you the unique TCP/IP network addresses for your organization's computers and supplies you with an electronic mail domain name, which is the portion of your e-mail address after the "@." Service providers often offer to run your Web pages on their servers.

Internet

7 The service provider links your Internet application (and those of many other users) to the Internet through a high-speed connection. Your TCP/IP packets can access thousands of Internet servers of varying types located around the world. The time taken from when you hit the Enter key to when the Internet server receives your request for access is a matter of seconds. The Web server sends TCP/IP packets back to your Internet application, displaying the Web site on your screen.





P A R T

8

PRINTING AND PUBLISHING

ACCORDING to *The American Heritage Dictionary,* the word *publish* is defined as the act of preparing and issuing printed material for public distribution. In light of this definition, many of us are publishers in one way or another. The final result of our work on a Mac is often on paper, whether it's the 8½-by-11-inch memo to be distributed among our coworkers or a four-color illustrated brochure to be sent to clients. Increasingly, the final result of our work is augmented by electronic publishing, including the creation of World Wide Web pages on the Internet. With the need to present our work in a manner that is informative and attractive, the phrase "publish or perish" applies to more people than just college professors.

In 1984, Apple introduced two new pieces of hardware, the Macintosh and a printer. The ImageWriter I was the first Mac peripheral, preceding even the first hard-disk drive. This dot-matrix printer was noisy and slow, and it produced low-resolution printouts. Still, it and the new Macintosh computer had two big strong points—they made printing easy, and they could print graphics as easily as text. The installation procedure consisted of plugging in a cable between the two devices. It also used a single software printer driver for all applications; you could select it once and then print anything from any program.

As convenient as the ImageWriter was, the introduction of the LaserWriter the following year was a much bigger advance in printing technology, and helped start the desktop publishing revolution. The LaserWriter was the first network printer and one of the first laser printers for desktop computers. It was quiet, could be accessed by multiple Mac users, and put quality text and graphics printing in the hands of ordinary users. The LaserWriter was also the first printer to use a page description language called PostScript, which made possible the scaling of type fonts to different sizes with acceptable results.

Desktop publishing and graphics are what made the Mac a successful machine in the mid-1980s. It was called desktop publishing because activities that were traditionally done by a roomful of people and thousands of dollars of equipment could now be done at your desk. Page layout software replaced the scissors and glue used to design a page of a newsletter or magazine and made revisions quick and easy. Electronic files replaced expensive and time-consuming photographic techniques, and the ability to include graphics right in the document file replaced the use of expensive stripping equipment.

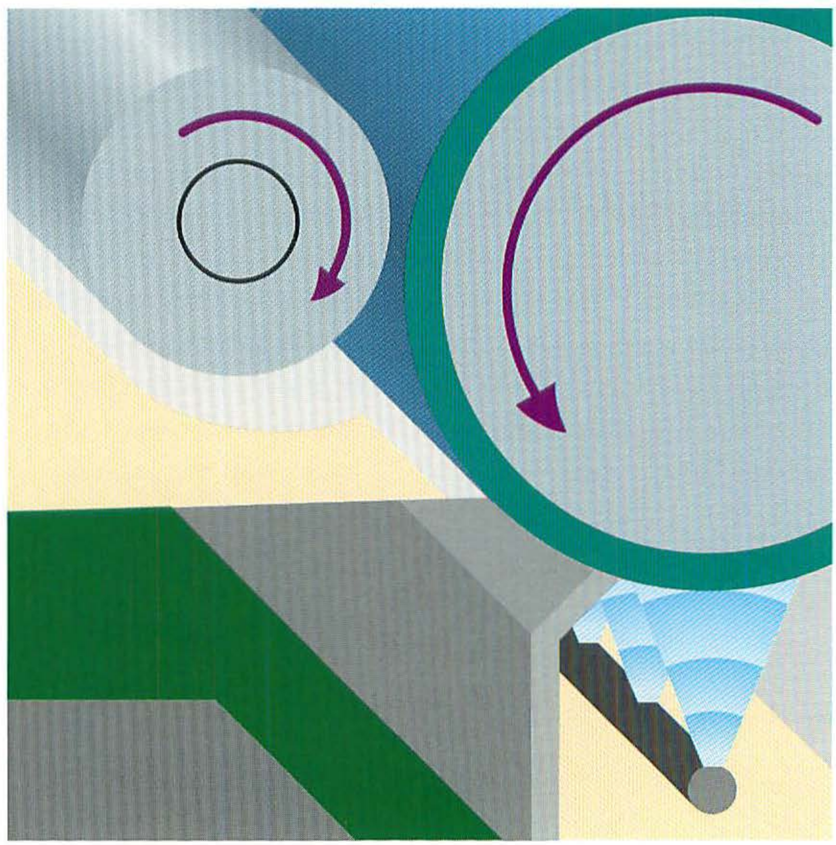
Today, the Mac's built-in graphics and printing technologies and ease of setup and use make it a favorite among professional desktop publishers. The Mac is widely used by magazines and newspaper publishers, and it was used to create the graphics, text, and page layout of this book.

Macs are also heavily used in the second phase of the desktop publishing revolution, electronic publishing. Many Mac publishing professionals have taken their skills over to the Internet, applying some of the same basic design principles to the new medium. This new area was inspired by the creation and rapid growth of the Internet's World Wide Web. The popularity of the Web, though spurred on by easy user access, was driven primarily by the ease with which publishers can create and post electronic documents online. The tools for electronic publishing are almost as varied as those for print media. Key developers of desktop publishing and graphic software, including Adobe, offer powerful electronic publishing software and are including features for electronic publishing in traditional desktop publishing products.

The advances in print publishing technology over the last decade have put more demands on hardware and software, particularly with the use of color. One problem area is still the lack of calibration of color among different peripherals: What looks like burnt umber on one monitor may look more like traffic-cone orange on another monitor or on a color printer. Support for color calibration was included with the addition of ColorSync system software in 1992, which became part of QuickDraw GX the following year.

Of course, printing and publishing on the Mac is not all high-end color production. Laser printers still form the majority of printed output devices in use today. Dot-matrix printers like the ImageWriter have been mostly replaced by ink-jet printers, which are quieter and produce better results than their predecessors. We'll start our discussion of how desktop publishing and printing work with these two workhorses—ink-jet and laser printers.

CHAPTER 30 How a Printer Works



LASER printers are ubiquitous office fixtures, now as common as copy machines. Though they vary in quality, speed, and cost, laser printers set the standards for these three printing parameters. Laser printers are available to fit the high-volume printing needs of big organizations, as well as the more modest requirements of the small workgroup and some individuals.

Ink-jet printers are popular because they are a great buy, offering low-cost printing with some of the print quality of laser printers. Ink-jet printers make great home printers, as they are almost silent in operation, small in size, and typically lightweight. Some are even portable and battery operated.

Most printers of all types create an image by drawing dots on paper. Standard printer resolution is 300 to 600 dots per inch, quite a bit higher than a monitor's 72 dots per inch, and higher resolutions are commonly available. One of the oldest types of printer technology, *dot matrix*, draws dots the same way typewriters create whole letters—by striking the paper through an inked ribbon. This makes dot-matrix printers useful for printing on forms that use carbon copies. Laser printers, on the other hand, are similar in design to copy machines. They use a dry powdered toner for the ink, which is applied electrostatically to the paper and bonded by heat. Ink-jet printers work the way the name implies, by squirting tiny jets of ink onto the paper.

Both laser and ink-jet printers come in color versions. Color printers work the same way their black-and-white counterparts do, except they print each page four times, one time each in cyan, magenta, yellow, and black ink. These colors combine to form all the other colors you need.

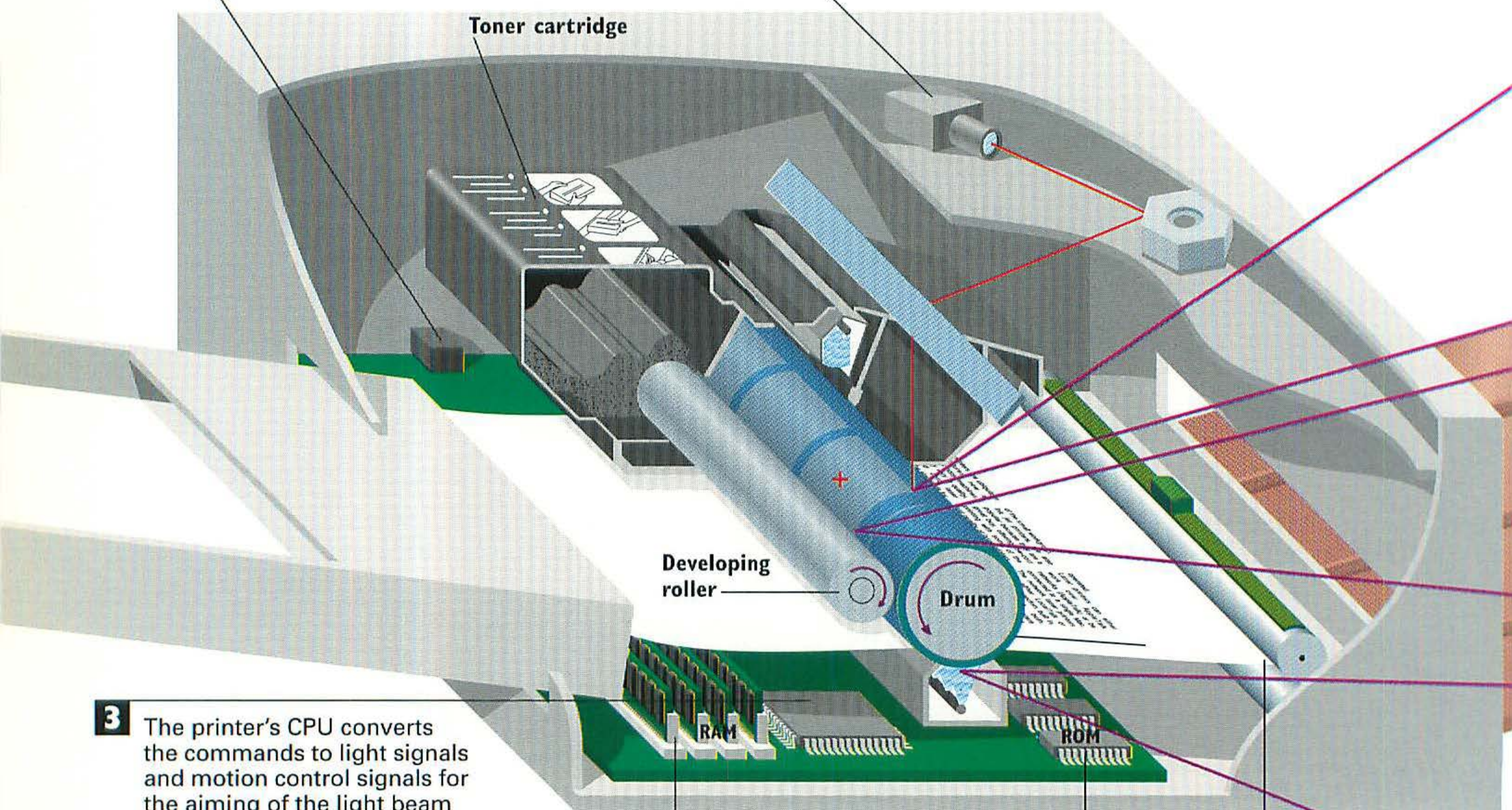
Color laser and ink-jet printers represent the lower end of color printing technology. Other types, such as wax-thermal and dye-sublimation printers, produce more realistic color images, but the costs of the printers and the printouts are much higher. The most expensive of these can produce color images indistinguishable from photographic prints.

Printers can either work over AppleTalk networks or be connected to a single Mac. Network laser printers have either LocalTalk or Ethernet connectors built in, and sometimes both. Typically, network laser printers use the PostScript page description language to print graphics and text. Nonnetwork laser or ink-jet printers are usually connected to a Mac's SCSI port or serial port and use the Mac's QuickDraw screen-drawing routines for printing.

Laser Printer

1 Printing commands that come in through an AppleTalk port are described by PostScript. In nonnetwork printers, the commands are from QuickDraw (see Chapter 24). The signals describing the document to be printed are sent to RAM on the printer's logic board for processing by the printer's CPU.

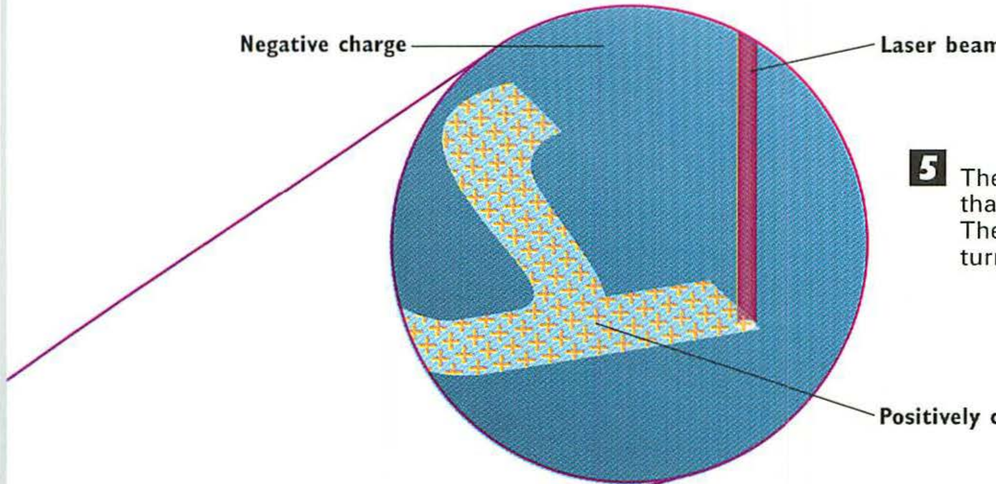
4 A laser beam is aimed at a rotating drum using a rotating polygonal mirror. The beam hits the drum one dot at a time. The laser is turned on where black dots will occur and is turned off where the page will remain white. Some printers use an array of light-emitting diodes instead of a laser. In standard printers, a black area will have 300 dots in every inch.



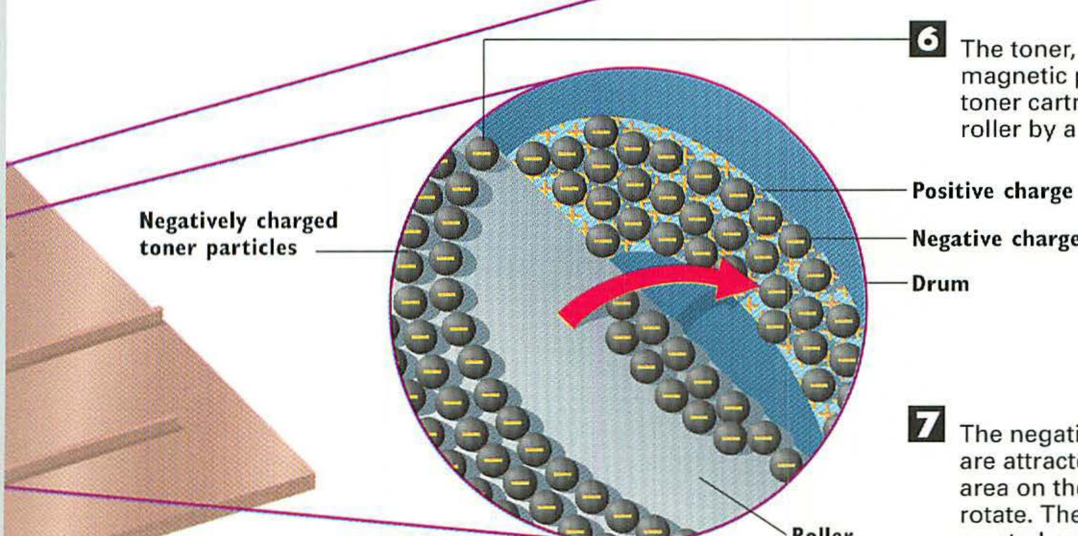
3 The printer's CPU converts the commands to light signals and motion control signals for the aiming of the light beam and the paper.

2 Some of the printing commands describe the fonts to be used in the document. Fonts are stored in the printer's ROM, in an external hard disk, or occasionally on a user's Mac. The requested fonts are loaded into RAM.

9 The toner is bonded to the paper by passing between two rollers heated to about 160 degrees centigrade.

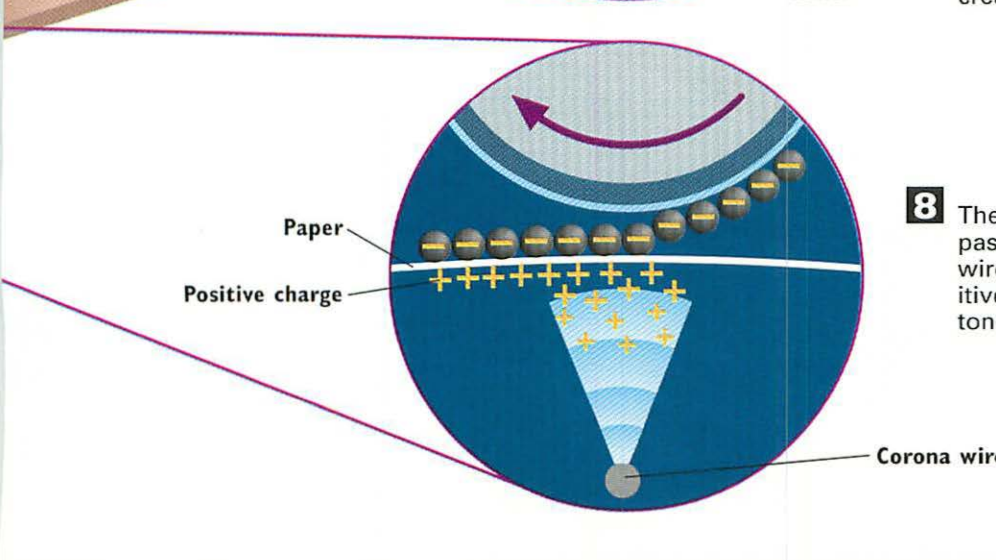


5 The rotating drum is coated with a material that has a negative electrostatic charge. The coating is light sensitive, and it will turn positive where the laser beam hits it.



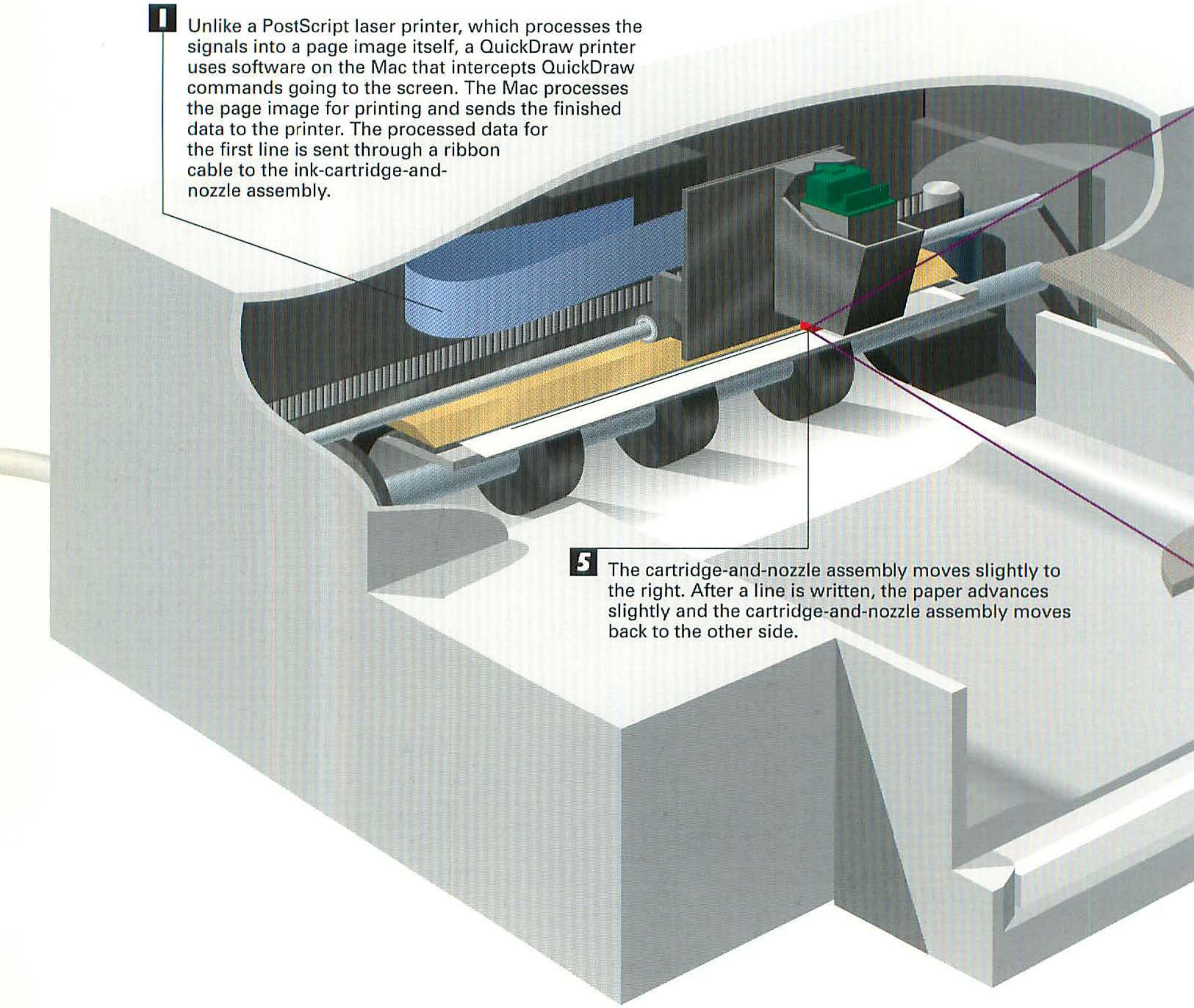
6 The toner, a fine powder containing magnetic particles, is drawn from the toner cartridge onto the developing roller by a magnet inside the roller.

7 The negatively charged toner particles are attracted to the positively charged area on the drum as the drum and roller rotate. The image to be printed is now created on the drum.



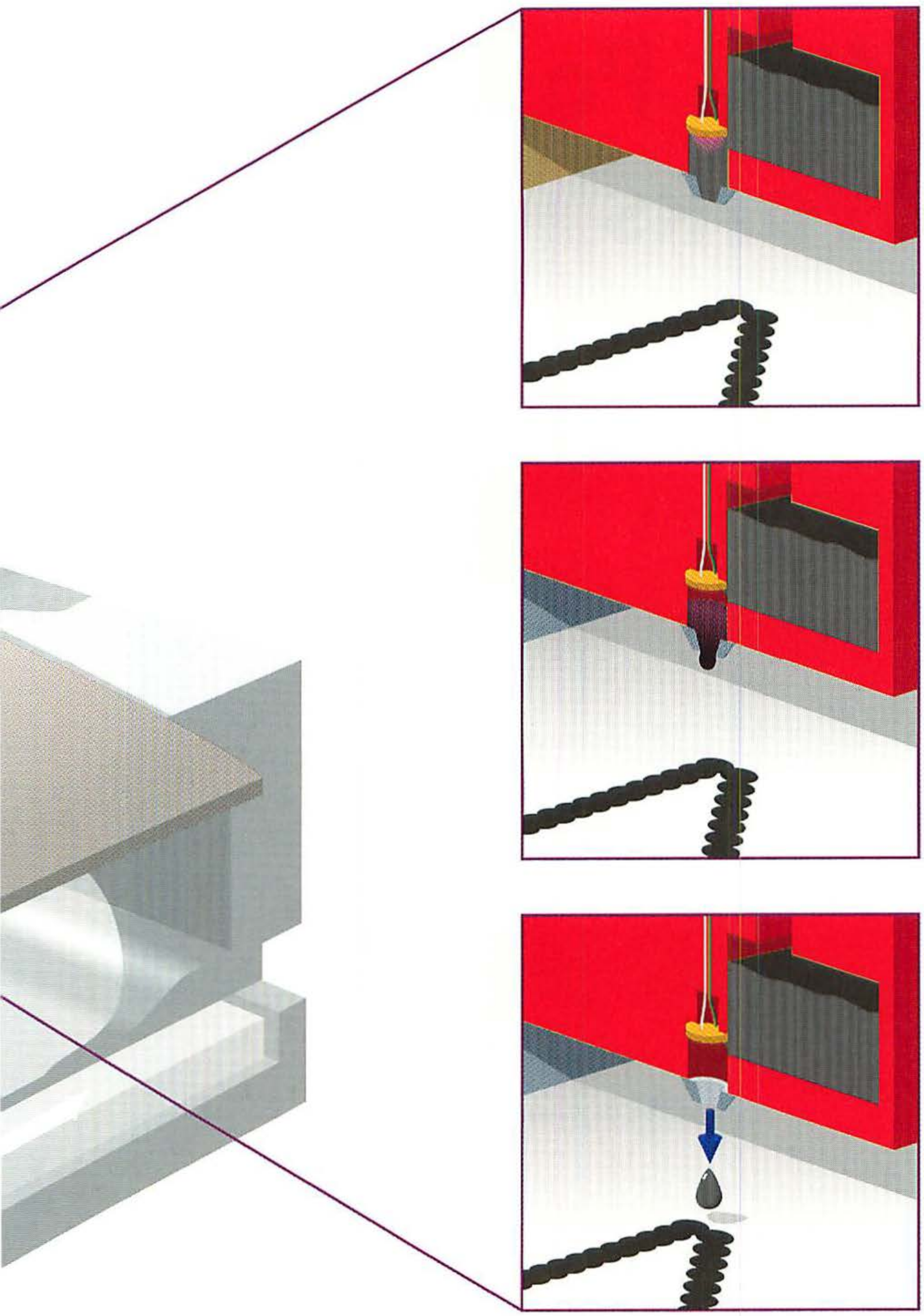
8 The paper pulled in from a paper tray passes between the drum and a corona wire, which is positively charged. The positive charge draws the negatively charged toner particles from the drum to the paper.

Color Ink-Jet QuickDraw Printer



1 Unlike a PostScript laser printer, which processes the signals into a page image itself, a QuickDraw printer uses software on the Mac that intercepts QuickDraw commands going to the screen. The Mac processes the page image for printing and sends the finished data to the printer. The processed data for the first line is sent through a ribbon cable to the ink-cartridge-and-nozzle assembly.

5 The cartridge-and-nozzle assembly moves slightly to the right. After a line is written, the paper advances slightly and the cartridge-and-nozzle assembly moves back to the other side.



2 The print head contains four cartridges, one each for cyan, yellow, magenta, and black ink. Liquid ink is pumped into 50 chambers, each containing a heating element.

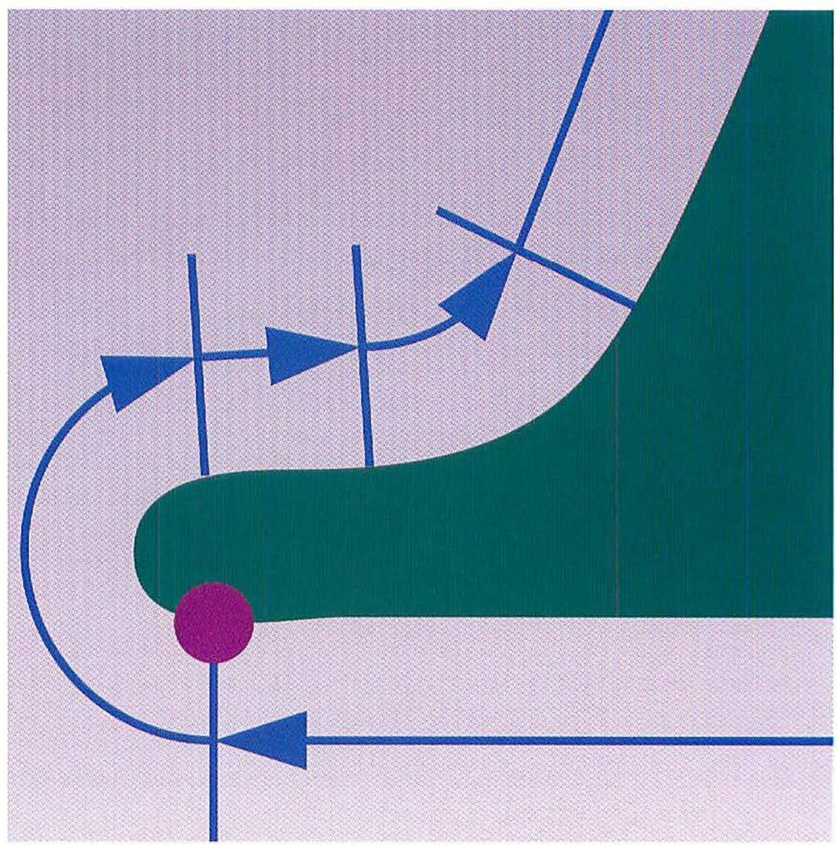
3 The heating element is switched on, and it heats the ink to its boiling point and vaporizes it.

4 The increased pressure of the gaseous ink forces the ink through the tiny nozzle, squirting a dot of ink on the paper.

MAC FACT Nonnetwork ink-jet or laser printers connected to a Mac's serial port can be shared with other users on a network through printer-sharing software running on the Mac the printer is connected to. With some software, such as Apple's GrayShare for the StyleWriter ink-jet printer, the Mac connected to the printer does the processing and will slow down when other users are accessing the printer. With Hewlett-Packard's software for the DeskWriter ink-jet printer, the processing is done by the user's Mac, not by the Mac the printer is connected to.

CHAPTER 31

How PostScript Works



PRINTERS and Macs speak different languages. When you print a file, you're not actually sending the file itself to the printer, you're sending a description of what's on the pages in a language the printer can understand.

You could send printers a bitmapped representation, which is a dot-by-dot account of what's on the page. This is how the Mac prints to QuickDraw printers, such as 1984's dot-matrix ImageWriter and today's ink-jet printers. However, bitmapped printing moves a large amount of data over cables, which presents a problem on networks. You also have to teach the Mac how to talk to each different printer you use.

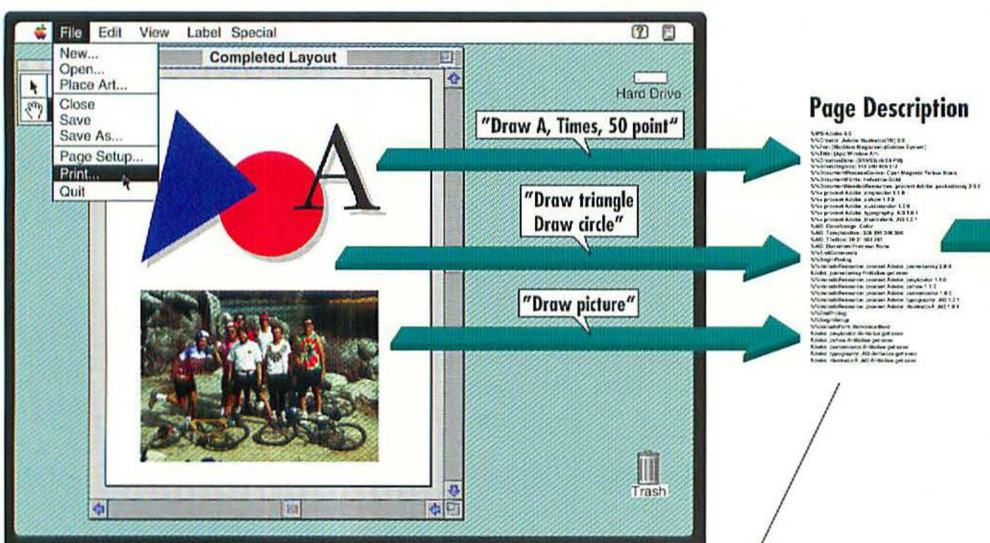
In 1985, Adobe Systems Incorporated introduced the PostScript page description language on the Apple LaserWriter. Since then it has been a standard method of telling printers what to print. Instead of describing each dot, PostScript describes everything in a document—including text, fonts, style, shapes, fills, and colors—mathematically. It draws the dots on the paper after the description of the file has been transmitted, not before. This is a much more efficient way of describing a document, requiring the transmission of much less data over the cables and producing more consistent results. PostScript is also printer independent because it describes text and graphics in a mathematical manner. PostScript sends its commands in a format that all PostScript devices can understand—ASCII (American Standard Code for Information Interchange) text.

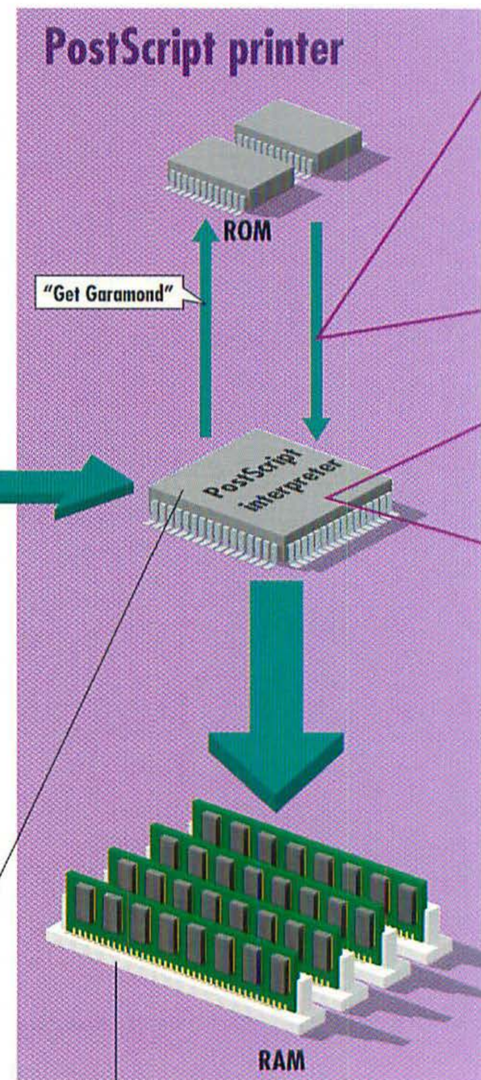
The PostScript page description language uses a type of font called *outline* because the outline of each character is used to draw each type of character. Outline fonts are usually stored in the printer. Most professional desktop publishers prefer to use PostScript Type 1 fonts or Adobe's Multiple Master fonts, which lets you create your own font styles. Mac OS comes with a type of outline font called *TrueType*, which can also be printed on non-PostScript printers, but is somewhat slower to print on PostScript printers than Type 1 fonts. Apple has another type of outline font, called QuickDraw GX, that offers advanced features such as automatic character positioning based on where a character is in a paragraph or sentence. However, few applications support QuickDraw GX fonts.

In the early 1990s Adobe released an extension to its page description language called *PostScript Level 2*, which adds new features, such as the ability to select paper trays from the printer dialog box on your Mac. PostScript Level 2—implemented in most newer printers—is a superset of Level 1, and can also do everything Level 1 can.

PostScript Printing and Outline Fonts

1 When you print a document to a PostScript printer, the application creates a small program called a *page description* (also called a print job), which will be executed by the printer. The page description contains a set of print commands written in the PostScript page description language; these commands treat the entire document as a single graphic. The commands describe graphic objects, such as lines, curves, circles, and squares, and more complex objects made up of these elements. They also describe scanned graphics, such as photographs. The page description contains all the characters in the document and the names of the fonts and styles used. The commands describe where on the page to draw characters and graphics, as well as their sizes, colors, and other attributes.

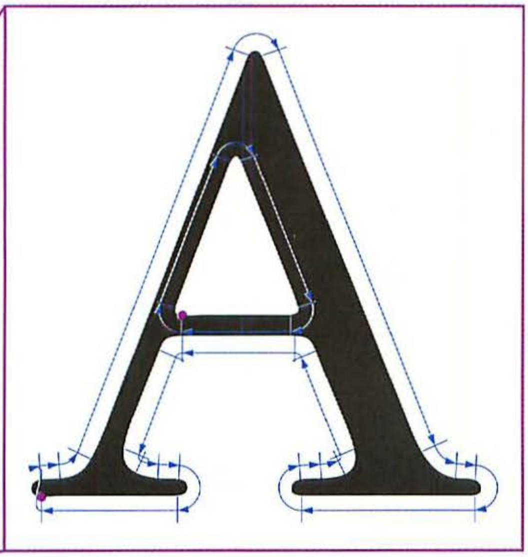




2 The page description is sent over the network to the printer as PostScript code written in ASCII text. Sending a list of commands is much more efficient and requires less code than describing the location and color of each pixel on the page, the way that monitors and QuickDraw printers do. Using ASCII allows any PostScript printer or device to understand the characters contained in the code.

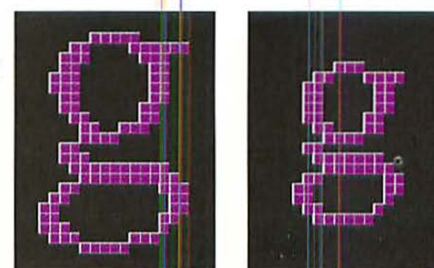
3 The page description is processed by the *PostScript interpreter*, which is hardware and software in the printer that executes each command and draws the picture. The PostScript interpreter recreates the document by performing *raster image processing*, turning mathematical descriptions of shapes into dots placed on the paper.

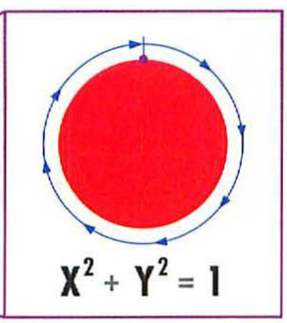
4 The interpreter first creates an ideal image of the page in the printer's memory.



5 When the page description commands specify the use of a particular font, the PostScript interpreter fetches the font stored in the printer's ROM, hard disk, or RAM if the font is downloaded from the Mac. PostScript can use several types of outline fonts. Each font is a set of mathematical descriptions of characters. Outline fonts are shapes, which are treated as graphical objects.

6 Because outline fonts are smooth outlines described mathematically, a character is shaped the same in any font size. *Bitmapped* fonts, on the other hand, are not scalable, and the character looks different at each size.

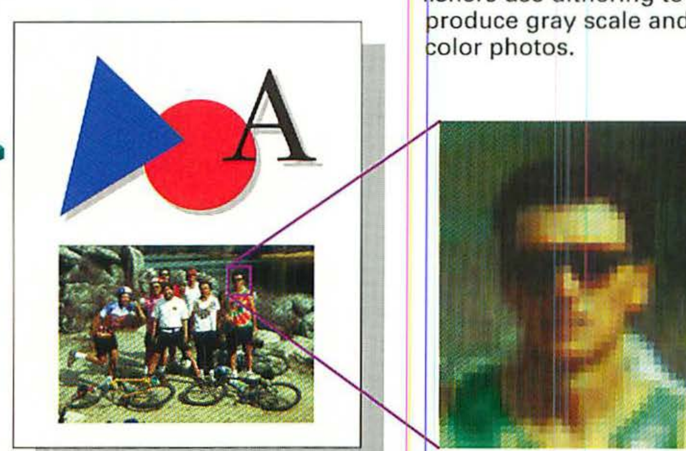




7 When the interpreter receives a command such as "draw a circle," it creates the path of the shape's outline using a simple mathematical equation. The outlines of more complex shapes are described by many mathematical equations and geometric relationships. Page description commands tell the interpreter the thickness of the outline and the colors (for a color printer) of the fill pattern.

8 With the document image recreated in printer RAM, the interpreter tells the printing mechanism to print dots of toner on the paper. This is called raster image processing. The number of dots per inch depends on the resolution and settings of the printer; the same page description code could produce 300 dots per inch on one printer and 1,000 on another. The resolution of the printout is completely independent of the monitor resolution.

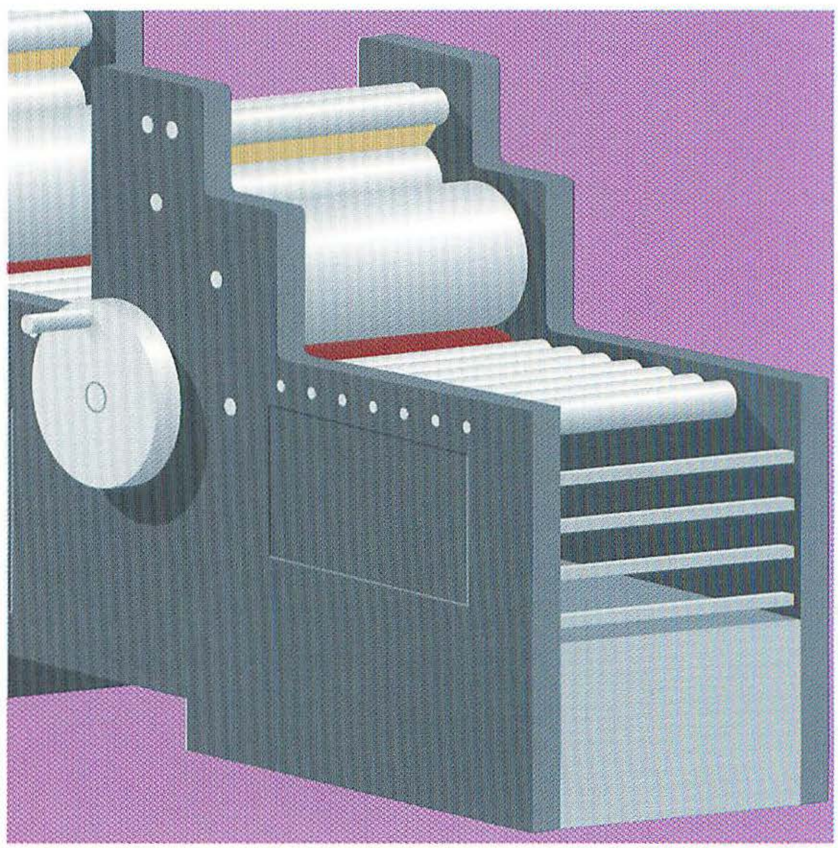
9 Most color printers mix dots of cyan, magenta, yellow, and black in different proportions to approximate colors; this is called *dithering*. For creating gray scale images, such as photographs, on printers that can only print black and white, the interpreter also uses dithering to produce approximations of grays called halftones. Dithering mixes dots of black and white in different amounts to appear as shades of gray. Newspaper publishers use dithering to produce gray scale and color photos.



MAC FACT Although applications create PostScript page descriptions automatically, PostScript is also a programming language that can be used to create special print effects not included in your applications—presuming you know the language. To get a look at PostScript code, many applications let you create a file containing all the PostScript code for a particular document. Usually, you create a file of PostScript code by clicking a button in the Print dialog box: Instead of sending a page description to a printer, the application saves it to your hard disk.

CHAPTER 32

How Desktop Publishing Works



DESKTOP publishing was born on the Macintosh. It was not something Apple invented, but grew out of a need to make publishing easier. The Mac supplied the first tools to put out a newsletter or brochure from your desk, tools that replaced the scissors and glue of page layout—formerly known as paste up—as well as the hundred-thousand-dollar machinery required for graphics. By reducing costs, making revisions easier, and making publications look better, the Mac put publishing in the hands of the individual and streamlined the empires of the publishing industry.

In the professional publishing industry, the Mac is the favorite desktop publishing platform, mostly because its integrated design makes it more versatile and easier to set up and learn than other platforms. Macs are used to put out newspapers and magazines all over the country. Even magazines covering IBM PCs and Microsoft Windows often use Macs in their production departments. This book was produced entirely on Macs. Of course, desktop publishing isn't limited to book and magazine publishers. The result of your work can be a color brochure, an eight-page newsletter, or an 8½-by-11-inch sheet of paper.

At the heart of desktop publishing is page layout software, which allows you to manipulate text and illustrations on a page. Page layout software has grown increasingly powerful, allowing detailed adjustments in the spacing of individual characters, as well as graphics editing. Today, you have a wide selection of page layout programs to choose from, and you can even do page layout in some powerful word processors.

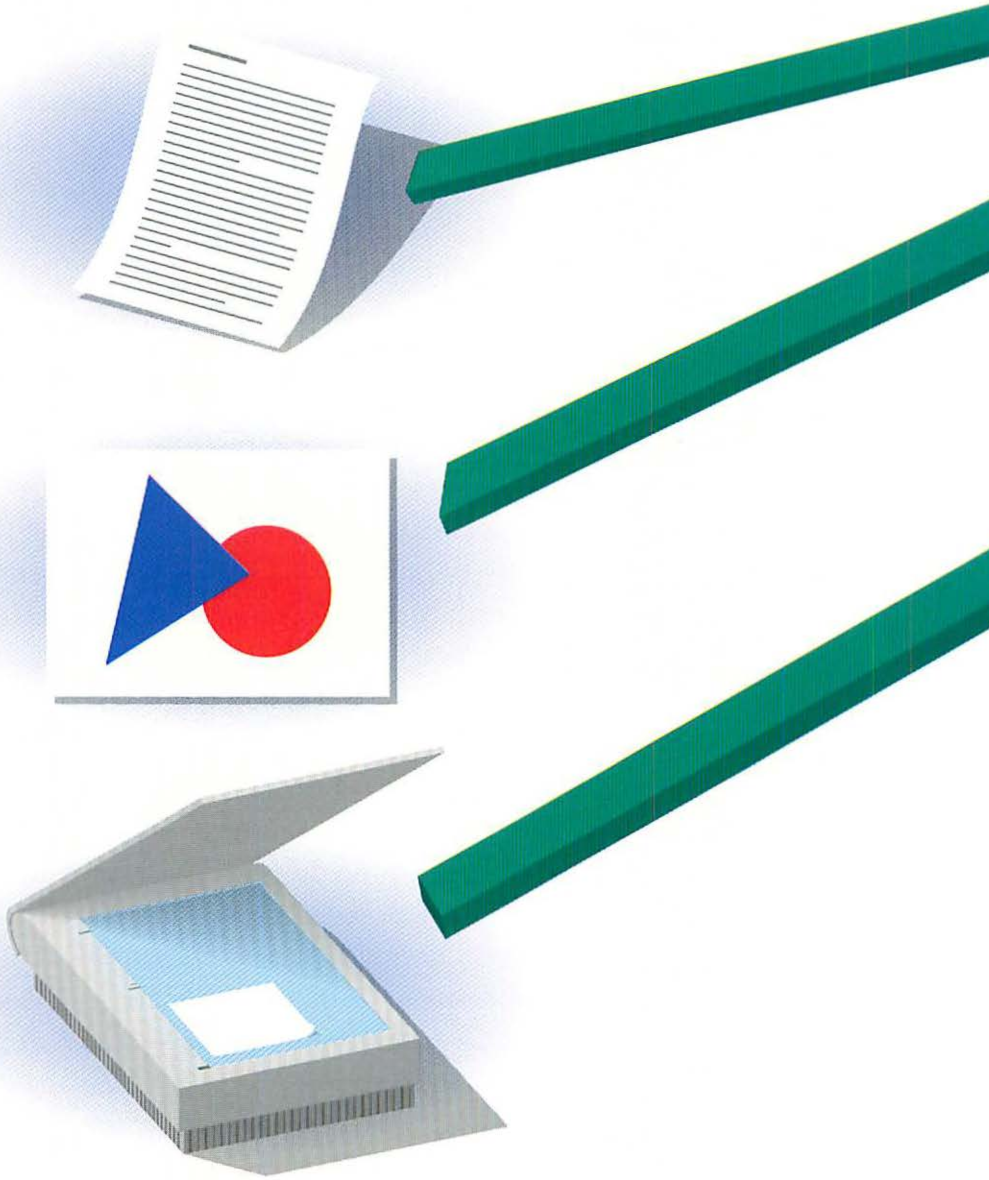
Placing a color illustration in a document was once costly and time-consuming. Not only is this an easy task with desktop publishing, but now a skilled artist can create the illustrations on the Mac. The possible styles range from simple informational graphics used in newspapers, to drawings that look like they were created with chalk and ink, to realistic, three-dimensional renderings.

Desktop publishing has been fully integrated in the production process all the way through professional printing. In most service bureaus, Macs control the process of color separation, a production step before color printing.

However, because the color one sees on a monitor is not usually the same as what prints, part of the color desktop publishing procedure involves color matching systems. These systems calibrate computer devices to a device-independent color model, and use this standard to match the colors produced by different devices. The most common device-independent color model is the XYZ model created by the Commission Internationale de l'Éclairage (CIE) in 1931. The *CIE XYZ* color model is based purely on experimental data gathered from standard light sources rather than mathematical color models such as RGB or CMYK.

Mac OS's ColorSync uses CIE XYZ to do color matching itself, and can also enable an add-in matching system from a third party. ColorSync is therefore flexible and extensible, and open to innovations in color matching that may come along in the future.

Desktop Publishing

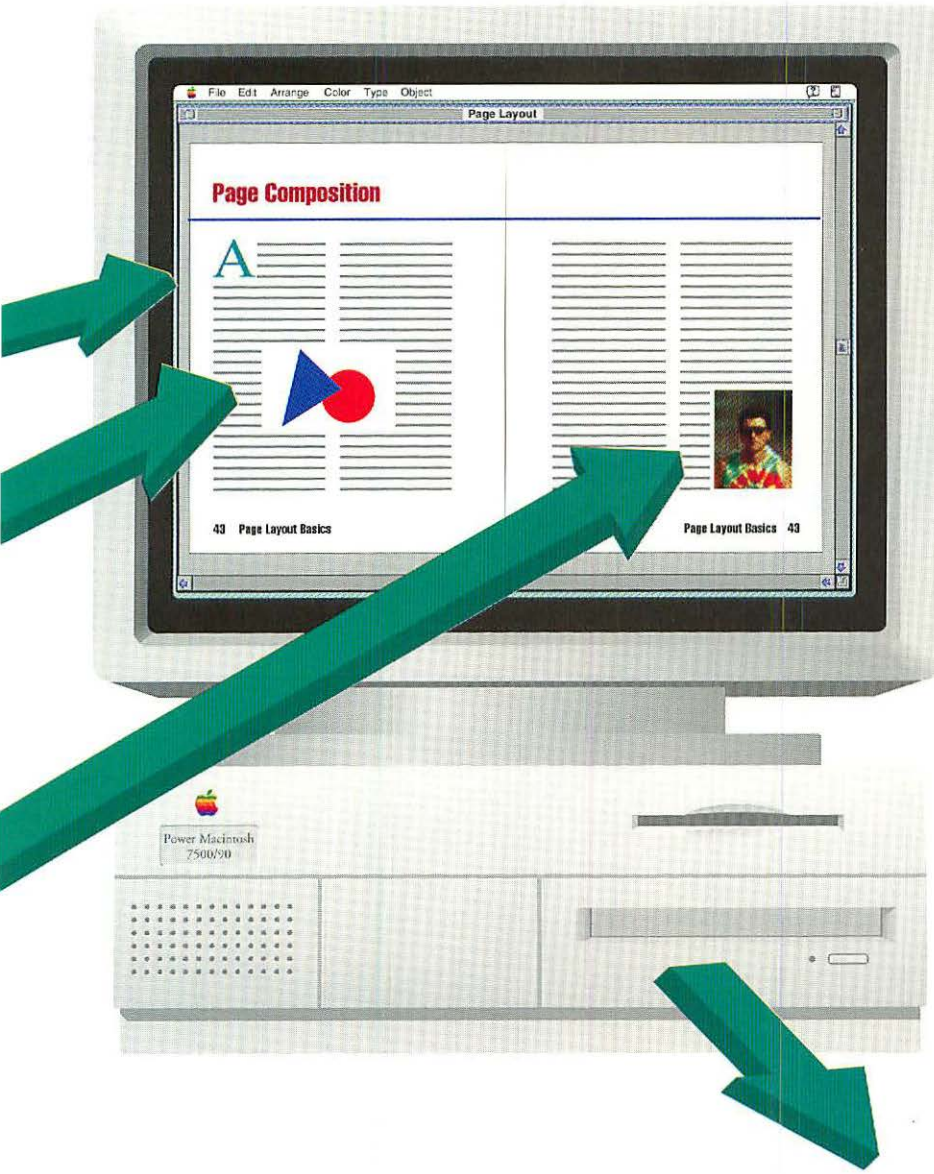


1 A writer creates the text of the document in a word processor. A full-featured word processor can be used as a page layout program for less complex documents.

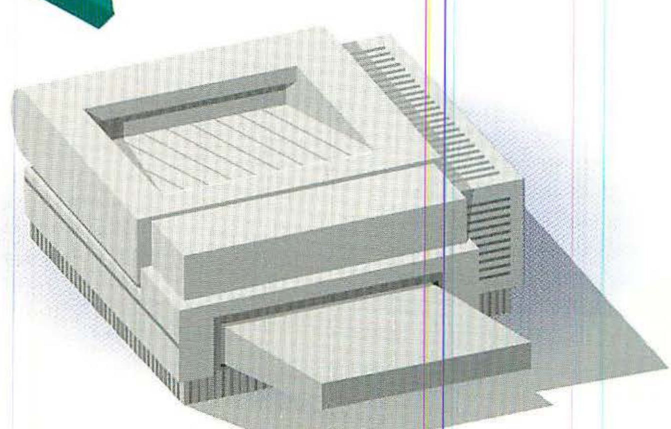
2 Artists use drawing or painting software to create illustrations (such as the ones in this book), as well as decorative elements on a page.

3 A scanner electronically captures images of photographs or other art. A flatbed scanner is typically used to capture photographs or images from books, although a slide scanner can be used for capturing 35-millimeter slides.

4 The electronic files are sent to a Mac running a page layout program. The files can be sent over a network or a telephone link, or they can be transported on floppy disks or other removable media.



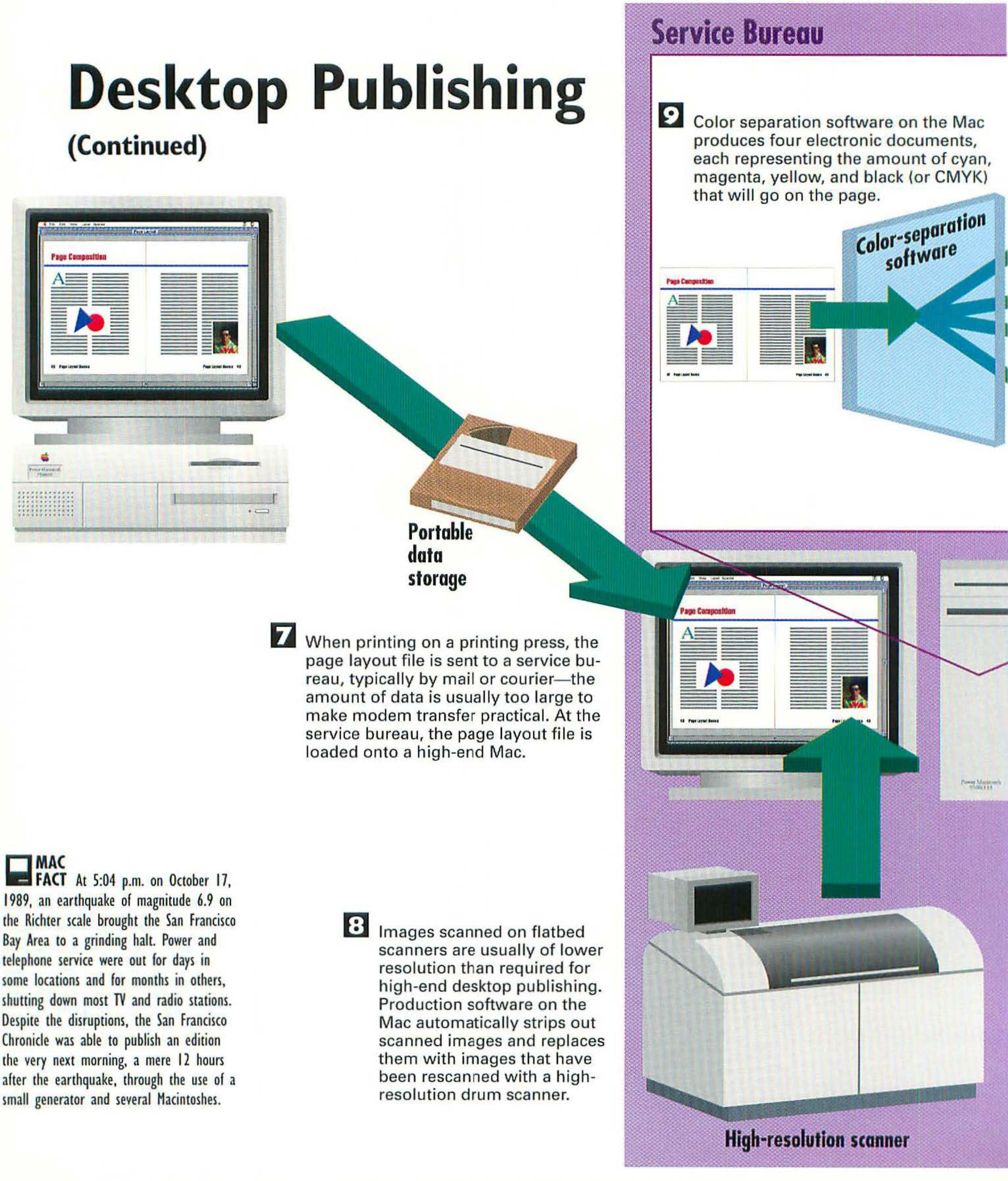
5 A page designer incorporates the text and graphics using a page layout program, which can import these elements directly from within the page layout file.



6 The pages are printed on a color or black-and-white printer. This could be a laser printer or a high-end dye-sublimation color printer. In either case, this printout can be used either as the final output or as a page proof to check for errors before continuing on with the process.
[*Continued on next page.*]

Desktop Publishing

(Continued)

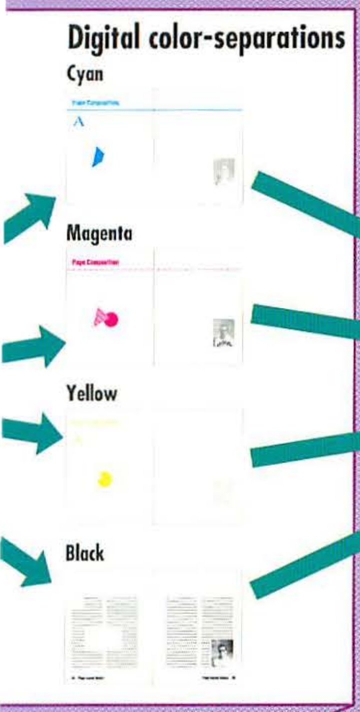


9 Color separation software on the Mac produces four electronic documents, each representing the amount of cyan, magenta, yellow, and black (or CMYK) that will go on the page.

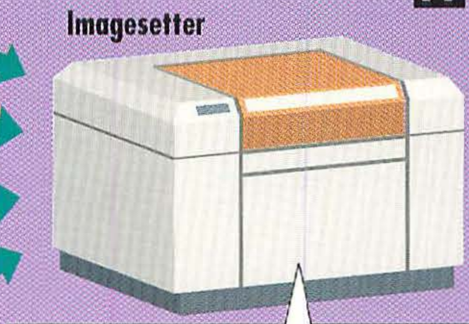
7 When printing on a printing press, the page layout file is sent to a service bureau, typically by mail or courier—the amount of data is usually too large to make modem transfer practical. At the service bureau, the page layout file is loaded onto a high-end Mac.

8 Images scanned on flatbed scanners are usually of lower resolution than required for high-end desktop publishing. Production software on the Mac automatically strips out scanned images and replaces them with images that have been rescanned with a high-resolution drum scanner.

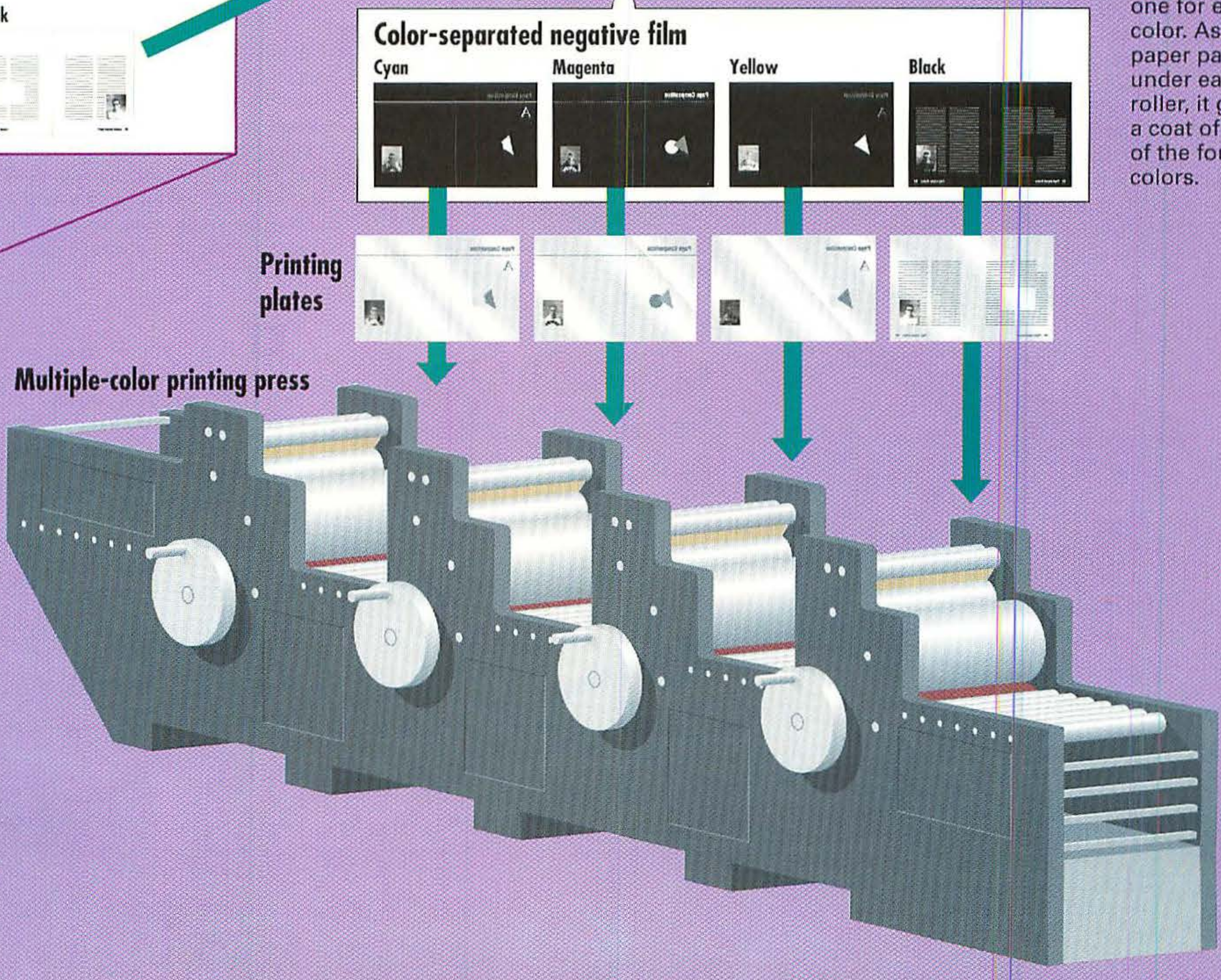
MAC FACT At 5:04 p.m. on October 17, 1989, an earthquake of magnitude 6.9 on the Richter scale brought the San Francisco Bay Area to a grinding halt. Power and telephone service were out for days in some locations and for months in others, shutting down most TV and radio stations. Despite the disruptions, the San Francisco Chronicle was able to publish an edition the very next morning, a mere 12 hours after the earthquake, through the use of a small generator and several Macintoshes.



10 The four electronic documents are fed into an imagesetter, which produces four full-sized transparent negatives (the white area of a page is black, and black text is white). Although there is one negative each for cyan, magenta, yellow, and black, the negatives themselves appear as black, white, and gray, not color.



11 The color separation negatives are used to make flexible plates for the printing press, one ink color per plate. A clear area on the film creates an area on the plate that the ink can stick to, and will allow all the color to print on the paper. A gray area on the film creates an area on the plate that accepts and prints a limited amount of ink. A black area on the film creates an area on the plate that ink will not stick to, leaving the paper white. The plates are attached to four rollers on the printing press, one for each color. As the paper passes under each roller, it gets a coat of one of the four colors.



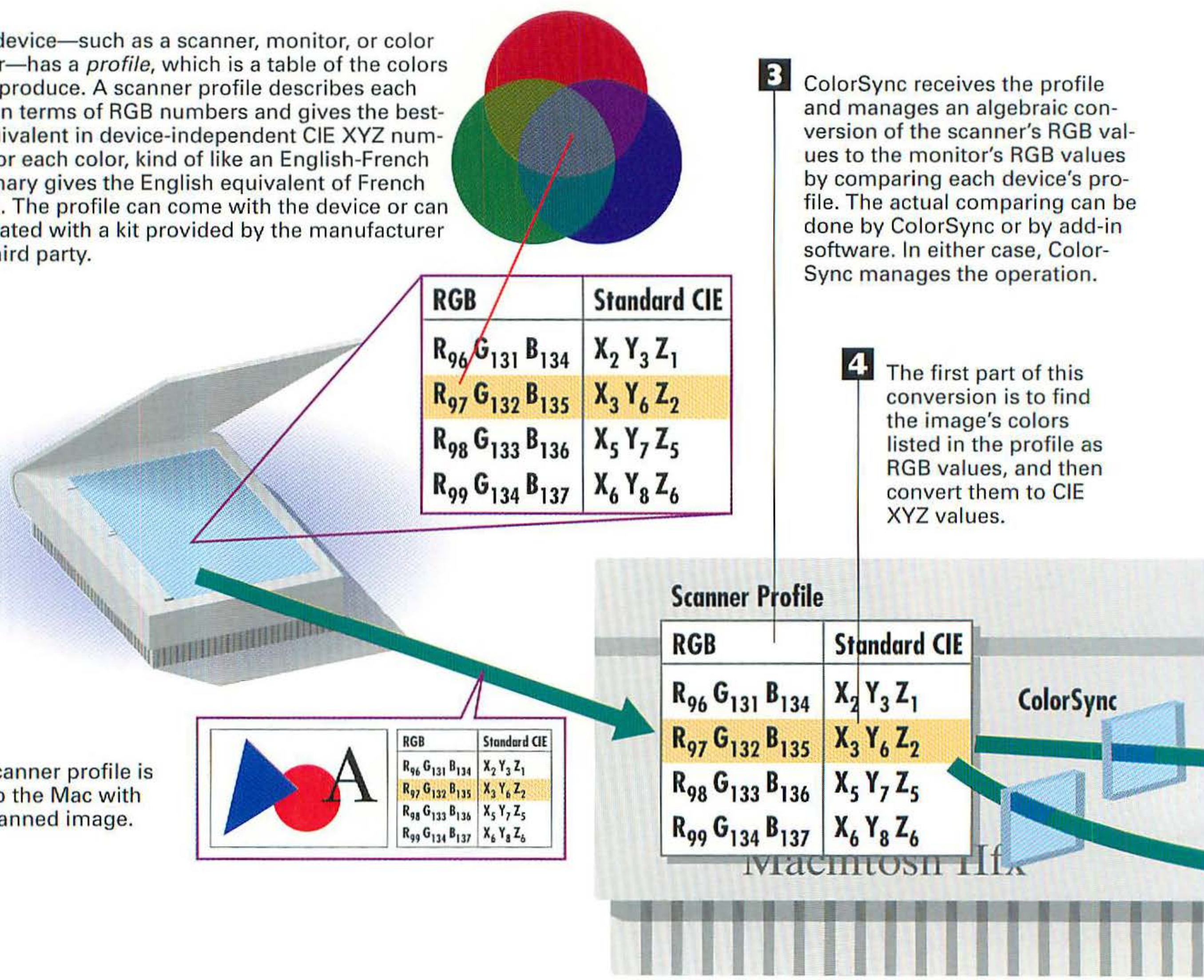
ColorSync Color Matching

ColorSync manages the matching of colors between scanners, monitors, and printers by comparing the description of a color by each system to the standard CIE XYZ color model. It doesn't actually alter the electronic file being passed from scanner to printer. Scanners and monitors describe a color using three numbers that represent the amounts of red, green, and blue (or hue, saturation, and brightness) in the resulting color. Printers use four numbers that represent the amounts of cyan, magenta, yellow, and black ink used to create a color. However, each device can only create a subset of the colors described in the standard CIE XYZ system.

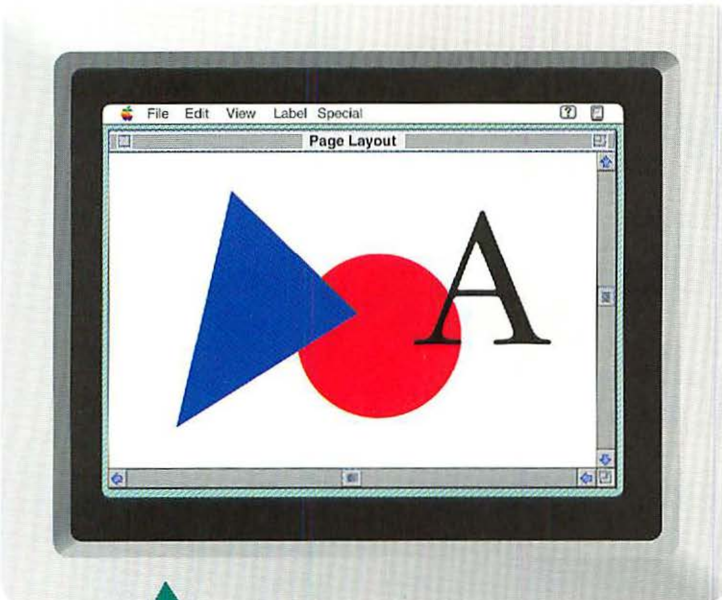
1 Each device—such as a scanner, monitor, or color printer—has a *profile*, which is a table of the colors it can produce. A scanner profile describes each color in terms of RGB numbers and gives the best-fit equivalent in device-independent CIE XYZ numbers for each color, kind of like an English-French dictionary gives the English equivalent of French words. The profile can come with the device or can be created with a kit provided by the manufacturer or a third party.

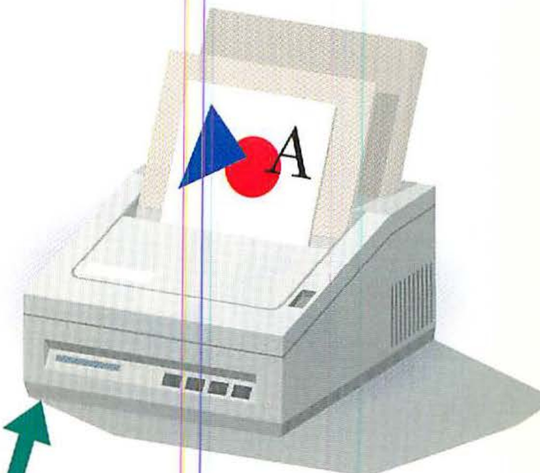
3 ColorSync receives the profile and manages an algebraic conversion of the scanner's RGB values to the monitor's RGB values by comparing each device's profile. The actual comparing can be done by ColorSync or by add-in software. In either case, ColorSync manages the operation.

4 The first part of this conversion is to find the image's colors listed in the profile as RGB values, and then convert them to CIE XYZ values.



2 The scanner profile is sent to the Mac with the scanned image.





5 ColorSync then uses the monitor profile to find the monitor's RGB values that best fit the CIE values. Corrections can be made at this time to compensate for any nonuniform display characteristics of the particular monitor.

6 When printing the original image, ColorSync compares the scanner profile with the printer profile, converting the scanner's RGB description of colors to the printer's CMYK (cyan, magenta, yellow, and black) numbers.

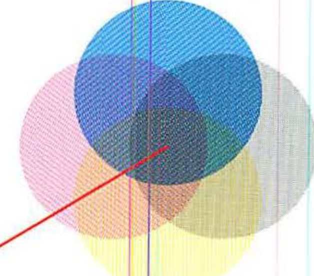
8 Printers use ink instead of light to create colors (scanners and monitors create colors using light). Unlike light, inks are *subtractive,* so that a superposition of all colors makes black. However, inks are not ideally subtractive in that a solid black is not produced, so a fourth ink, black, is added to get better quality black and dark colors. However, only three colors are required to create any color. The fourth color, the black ink, adds another variable, so there are multiple combinations of CMYK that can result in the *same* color. In addition, the type of paper used can alter color. For these and other reasons, printers are the most difficult devices to calibrate, that is, to create an accurate profile for.

Monitor Profile

Standard CIE	RGB
$X_3 Y_6 Z_0$	$R_{95} G_{123} B_{135}$
$X_3 Y_6 Z_1$	$R_{96} G_{123} B_{135}$
$X_3 Y_6 Z_2$	$R_{97} G_{123} B_{135}$
$X_3 Y_6 Z_3$	$R_{98} G_{123} B_{135}$

Printer Profile

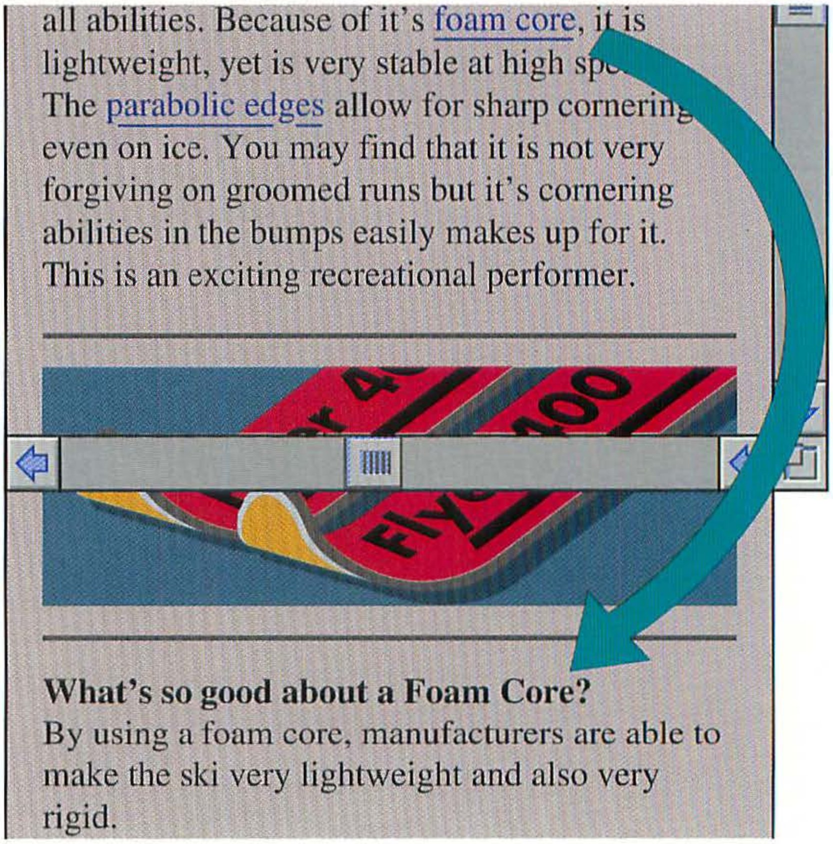
Standard CIE	CMYK
$X_3 Y_6 Z_1$	$C_{60} M_{22} Y_{25} K_{25}$
$X_3 Y_6 Z_2$	
$X_3 Y_6 Z_3$	$C_{61} M_{22} Y_{25} K_{25}$
$X_3 Y_6 Z_4$	$C_{62} M_{22} Y_{25} K_{25}$



7 If ColorSync finds a color the printer cannot reproduce (a color that is not in the printer's profile), it picks another that is close.

CHAPTER 33

How Electronic Publishing Works



DESKTOP publishing in the eighties changed the publishing world forever, creating new industries along with new opportunities for disseminating information. Electronic publishing is doing the same in the nineties. Not only are publications originating in digital form, but they're being delivered in digital form to computer users over networks. The user gains the ability to search for information by typing in keywords, and publishers get an efficient, low-cost method of distribution.

There are several avenues of electronic publishing. Local area network software programs such as AppleSearch and Lotus Notes have provided the means for the distribution of electronic information within an organization. For publishing by the general public, the Internet offers several venues for electronic publishing, including electronic mailing lists, FTP (file transfer protocol) servers, and gopher servers. However, the most popular method of Internet publishing is also the newest—the World Wide Web.

Unlike the older FTP and gopher services, which are merely depositories for electronic files, the Web is a sophisticated publishing medium of its own. It is also easier for publishers to set up a Web site than to set up an FTP site. You can fill your Web pages with graphics as well as text, and the Web is now incorporating sound, video, and animation.

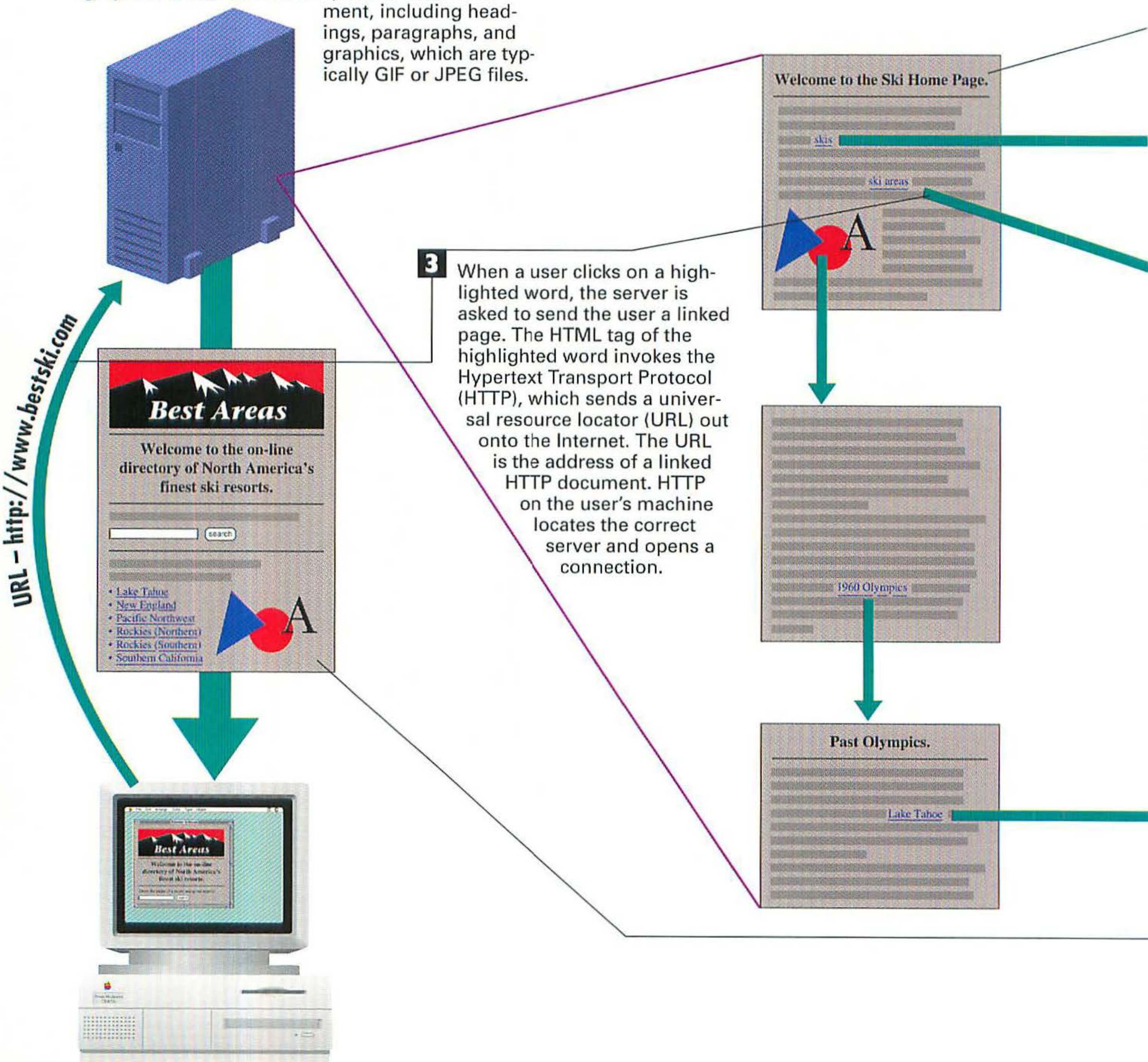
Like a paper publication, a Web document must be created with the graphic design concepts of space, color, and proportion to make it attractive yet easy to read. But the Web as a publication medium is fundamentally different from paper. The biggest difference is the added dimension of user interactivity. The Web's central interface concept is called *hypertext*, an idea conceived in 1965 by computer futurist Ted Nelson. (The concept is called hypermedia when graphics are involved.) Publishers post Web pages in blocks based on content with highlighted links. The user clicks on a highlighted word or picture, and the server sends the user to another portion of the document or to a related document. The link can send the user to material residing on other Web servers on the Internet, and the user doesn't even have to be aware of the other server's name.

Behind the Web's hypermedia interface is a programming language called the Hypertext Markup Language, or *HTML*. However, with the right electronic publishing tools, you don't even have to learn HTML programming to create an effective Web page.

With all the talk of hypertext and hypermedia, the Web is often hyper-hyped. For the user, searching for information can be difficult. The user interface of the Web is often not as effective as that of online services, and is usually quite a bit slower. The Web doesn't have the resolution of print media, yet Web documents filled with complex graphics take several minutes to access for the majority of users, who access the Internet with modems. But for electronic publishers, the cost-effectiveness and ease of publishing on the Web are unmatched at this time. Electronic publishing is not going to replace print media, but rather has joined it as another mode of distributing information.

A Web Site

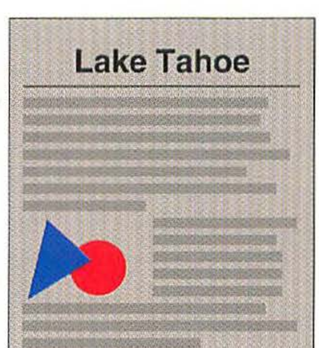
1 A Web site resides on a Web server, which can be a Mac, UNIX workstation, or a PC. It consists of a collection of HTML documents containing text and graphics. HTML defines the parts of each document, including headings, paragraphs, and graphics, which are typically GIF or JPEG files.



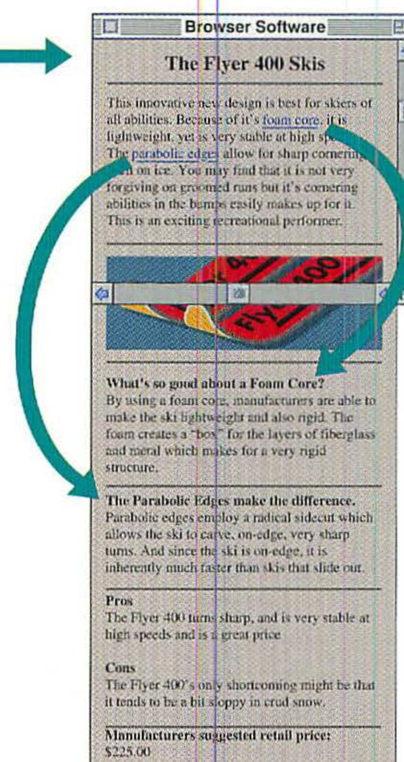
3 When a user clicks on a highlighted word, the server is asked to send the user a linked page. The HTML tag of the highlighted word invokes the Hypertext Transport Protocol (HTTP), which sends a universal resource locator (URL) out onto the Internet. The URL is the address of a linked HTTP document. HTTP on the user's machine locates the correct server and opens a connection.

2 A home page is the first Web document you want users to access, a jumping-off place for the rest of your site. Like a newspaper story, information is posted in order of decreasing importance. HTML links called tags link specific words and graphics to other documents.

5 HTML documents can be reached by links from several different pages. Links can also look up pages at Web sites residing on other servers anywhere on the Internet.



6 HTML documents don't have to fit in a single computer screen, but can be very long, like word processing documents. On these pages, it is useful to use HTML tags that jump to other places in the same document.



4 The server downloads the page to the user and disconnects the link. The user receives the document as an HTML description of what's on the page. The user's browser software then recreates the page based on the HTML description. (This is similar to the way a printer receives a PostScript description of a page.) This is much faster than it would be to send a bitmap image.

INDEX

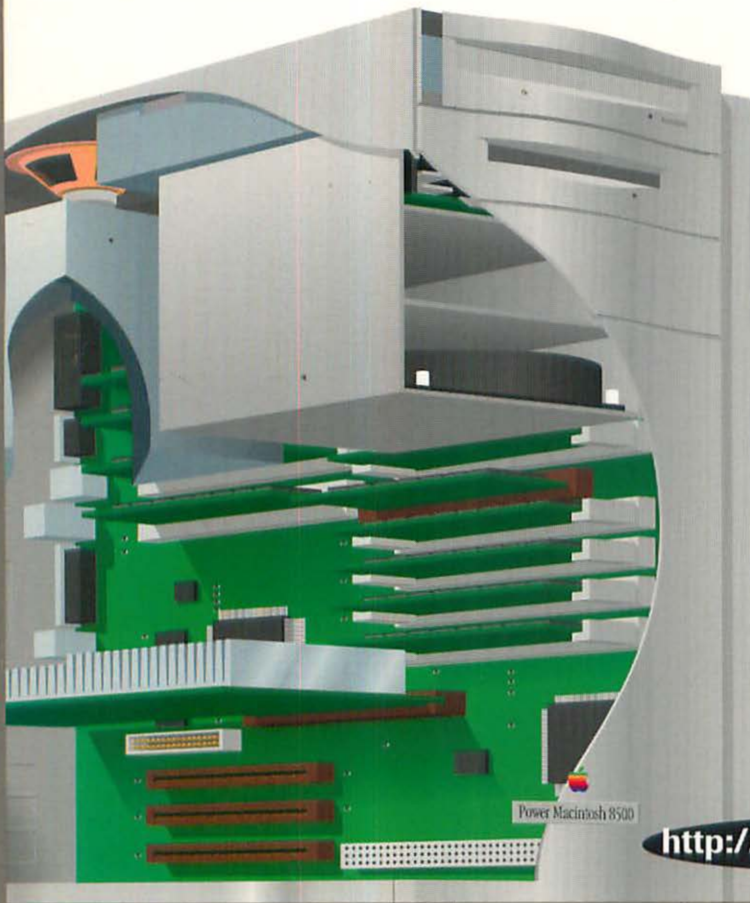
A

B

C

Get Inside Your Mac!

In *How Macs Work*, you will witness the technological beauty of the Mac at a level of detail and clarity not found anywhere else. This book answers such questions as . . .



- How does the Mac kick itself into action when you start it up?
- How does computer memory work in the Mac and how does a RAM cache make it work better?
- How do floppy disks and hard disks store information and send it back to the computer?
- How does the highly secret Mac ROM chip give the Mac its unique look and feel?
- How can a personal Mac become part of a finely tuned office network?

QuickDraw . . . PCI . . . NuBus . . . SCSI–Fast and Wide . . . Ethernet . . .

You'll not only understand the jargon but also marvel at the way technologies mesh to form the unified wonder known as the Macintosh.

This Bestseller Edition has been updated to cover new models, the latest operating systems, and new technologies such as QuickTime and Powerbook Wireless Networki
and Open

http://www.mcp.

John Rizzo is a contributing editor of *MacUser*, a frequent contributor to *Computer Currents*, and the author of *MacUser Guide to Connectivity*.

K. Daniel Clark is a freelance informational-graphics designer for international print, CD-ROM, and online publishers.

Both Rizzo and Clark live in San Francisco, California.

Ziff-Davis Press is an imprint of Que New Technologies, a division of Macmillan Publishing USA. Macmillan also includes Sams Publishing, New Riders, Hayden Books, BradyGAMES, Sams.net, QUE Education and Training, and a software division, QUE Software. Macmillan Publishing USA is a division of Simon & Schuster, the publishing operation of Viacom.

HOW MACS WORK

BESTSELLER EDITION

JOHN RIZZO AND K. DANIEL CLARK

Part of the **BEST-SELLING** HOW IT WORKS series

ISBN 1-56276-401-2

.S. $24.99
ANADA $33.99
.K. £22.99 Net
SBN 1-56276-401-2
OOKSHELF CATEGORY
pple/General Computing