Jargon

an informal dictionary of computer terms

Other books by Robin Williams

The Little Mac Book

The Mac is not a typewriter

The PC is not a typewriter

PageMaker: An Easy Desk Reference (Mac and Windows versions)

Tabs and Indents on the Macintosh

How to Boss Your Fonts Around

The Non-Designer's Design Book

Other books by Steve Cummings

DeskJet Unlimited

Understanding Quicken (Windows and DOS versions)

ABCs of Ventura
with Bob Cowart

The Little Laptop Book

LaserJet IIP Essentials

Everybody's Guide to Homeopathic Medicines
with Dana Ullman

Jargon

an informal dictionary of computer terms

Robin Williams

with Steve Cummings

Peachpit Press ▾ Berkeley ▾ California

Jargon, an informal dictionary of computer terms

Peachpit Press, Inc.
2414 Sixth Street
Berkeley, California 94710
510.548.4393
510.548.5991 fax

Several of the wonderful cartoons created by John Grimes
are excerpted from an anthology of his work, *Reality Check.*
Copyright ©1991–1993, by John Grimes.
Reprinted courtesy of Ten Speed Press
P.O. Box 7123
Berkeley, California 94707

Notice of Liability

The information in this book is distributed on an "As is" basis, without warranty. While every precaution has been taken in the preparation of this book, neither the authors nor Peachpit Press, Inc. shall have any liability to any person or entity with respect to any liability, loss, or damage caused or alleged to be caused directly or indirectly by the instructions contained in this book or by the computer software and hardware products described herein.

Trademarks

Throughout this book trademarked names are used. Rather than put a trademark symbol in every occurrence of a trademarked name, we state we are using the names only in an editorial fashion and to the benefit of the trademark owner with no intention of infringement of the trademark.

ISBN 0-938151-8-43

0 9 8 7 6 5 4 3 2
Printed and bound in the United States of America.

This book is dedicated,
with great love,
respect, and
admiration,
to BMUG,
for what they gave to me,
and for what they give to the worl d.

Acknowledgments

Shannon Williams, my little sister, for everything imaginable (if you like my books, you should thank her for keeping me together enough to write them)

Steve Cummings, for adding so much value to this book

Carole Quandt, for proofreading and editing and suggestions (but I added and edited so much after she proofread that all typos and grammatical faux pas are entirely my fault)

John Grimes, for the wonderful cartoons

Guy Kawasaki, definitions and moral support

Clay Gordon, tech edit

Bob Weibel, tech edit

Doug McCasland, editing and tech edit

Pam Mason, for her great appendix

Harrah Argentine, illustrations under font

Olav Martin Kvern, bailing me out of corrupted file trouble

Barbara Sikora, Drew Cronk,
Janet Butcher, and **Pam Mason**
for writing some of the definitions

Sam Hunting, Dave deBronkart, Tom Hagan, and **Jim Alley** for contributing to several of the definitions

Terry O'Donnell from Adobe Systems, for clarifying *font* vs. *typeface*

Andy Baird, moral support

Phillip Russell, whom I have never met, but who gives me tremendous moral support (Phillip, I do hope this book meets your expectations!)

Jon Winokur, for the perfect quote

Hassan Herz of Typecast, Inc., for imaging this book so efficiently and kindly

Cary Norsworthy at Peachpit, for pulling this all together and being so patient with me

Ted Nace, publisher of Peachpit Press (Ted, you're too good to me)

Scarlett Williams, for the font design

My kids, for being so patient with me for the past 18 months (did they have a choice?): **Ryan** (now 15 years old), **Jimmy** (11), **Scarlett** (7), and **Brian** (18, our foreign exchange student for the year). Do you think they will ever thank me for all the things they learned this year, like how to do their own laundry, cook their own meals, mend their clothes, grocery shop, clean the house, etc. etc. etc.?

The following people, none of whom I have met (isn't it an interesting phenomenon that we can make friends electronically), added and added and added words they felt needed to be defined, which is what made this book 688 pages instead of the original 120, which also made the book a year late in the production schedule, which also made the book much more useful. Thank you!

John Merideth
Herb Schulsinger
Jim DeWitt
Jehuda Saar
John Holland
Alleta Baltes
Brad Mohr
Erik C. Thauvin
Richard M. Parres
Glenn Brown
Pam Michaelson
Brian Carter
Etienne Grosjean
Ken Schneider
Lee Hinde
Jeffrey Kane
Alan Touchberry
Dan Druliner
Michael Lilly
John Thatcher
John Raymonds
Forrest F. Carhart, III
David Bass
Robert L. Hoover

Contents

JUST BEFORE WORK, ROBIN WOULD SLIP INTO THE LADIES ROOM AND ADJUST HER PERSONALITY.

Personal note

This book was not easy to write. I'm not a very technical person, and I had to struggle with many of these esoteric concepts. As I was writing this (and constantly complaining) many people would tell me (especially male technonerds) that no one needs to know this stuff anyway. But computers are becoming part of everyone's lives and we are all bumping into technical terms in magazines, catalogs, promotional literature, manuals, and so on. I have read letters from readers to magazines complaining that the articles don't explain the acronyms, but y'know what? Even if they told you what the acronym stood for, it wouldn't make any more sense—big deal that SCSI stands for Small Computers Systems Interface.

I wrote this book because I was frustrated with not being able to understand most computer dictionaries. The explanations were usually so brief, they assumed I knew a host of other things, and they never told me how this information was connected to anything else in my life or why it might be important for me to know this. After reading the following definition for SIMM (which was not alphabetized under "SIMM" but was under "Single In-line Memory Module" as if I would know what SIMM stood for), I realized I had a personal obligation to write definitions that average users like me could understand. This was the definition for SIMM:

> *A memory module that contains the chips needed to add 256K or 1M of random-access memory to your computer. A SIMM plugs into a motherboard or a logic board.*

This definition assumes I know what *memory* means, what a *memory module* is, what a *chip* is, what *256K* could be, what *1M* stands for, what *random-access* means, what a *motherboard* is, and what a *logic board* is. If I knew what all those things were, I wouldn't have to look up Stupid In-line Memory Module in the first place. (And I don't understand why computer dictionaries can't start a paragraph with a complete sentence.)

Oh, I shouldn't be so crabby—a few of the existing dictionaries are excellent, and I can recommend the ones that helped me tremendously and that you should use if you need more than what these definitions offer. And anyway, this book has a different purpose than most dictionaries. This book is not meant to be the ultimate reference—its purpose in life is to provide you, assuming you are a beginning to average computer user, with the information necessary to just get your everyday work accomplished. And (I do hope) you may even understand *why* you are doing what you are doing.

Yes, it's true that I am a Mac addict, and if I was less opinionated I would apologize for it. Steve Cummings (whom I have never met!) added so much value to this book by rounding it out with his PC-specific definitions, as well as adding to many of the generic defs. He was truly wonderful to work with, and we were able to hack away at each other's writing without remorse. I would be happy to work with him again.

Note from Steve: All I want to say is that I had a blast working on this book. Robin is as funny, gracious, and generous a collaborator as anyone could ask for, and I'm lucky I got to help out. Anytime, Robin, anytime.

Technical note

This book covers terminology for all computers. But if you are strictly a PC user, you probably don't give a hoot about the Mac references, and I know (cuz I'm one of 'em) Mac users don't like to slog through PC information. So every definition has been carefully edited so you won't accidentally read something distasteful or useless.

This icon means this paragraph and all the paragraphs that follow contain information that is specific to the Macintosh.

This icon means this paragraph and all the paragraphs that follow contain information that is specific to the PC ("PC" as defined below).

If there is no icon, the paragraph applies to all computers.

An *italic* word means I realize you might not know that term and it is defined as a main entry (but some words I used so often that I thought it would be irritating if I italicized it every time, words like "hard disk" or "Windows"). A word in "quotation marks" means it might be a new term to you, but it is not defined elsewhere—its definition is probably right there in that sentence.

Throughout this book, we use "PC" as the generic term for all computers patterned after the original IBM PC, meaning that they can use the same software. Since 1981 when the original IBM PC came out, hundreds of companies have made countless computer models that work basically like the IBM PC. No matter what the official model name, all of these machines are PCs. IBM itself stopped using the name PC in 1987, but IBM's newer models still qualify as PCs because they run PC software.

Likewise, when we use the term "DOS," we mean any brand and any version of DOS for PCs. DOS, the Disk Operating System, is by far the most popular kind of master control software, or operating system, used with PCs. You might have MS-DOS—the most common brand name—or PC-DOS, IBM DOS, or DR DOS, but fundamentally, they all work the same way. And even though there are some important differences between specific versions of DOS—between, say, MS-DOS 2.11 and MS-DOS 6—the basic commands are always the same.

I indexed this dictionary. I know, you might think it is a dumb idea to index a dictionary, but when I was trying to figure out what 32-bit addressing was, I found it helped me tremendously to read every definition in every book that referred to that topic. Without an index, how would you know all the possible definitions that contain information about graphic file formats or various features of hard disks or the different kinds of buses and cartridges and fonts? And how would you find all the words that don't have a separate entry but are defined as part of another definition? If you don't find what you need in the main entries, definitely try the index. Personally, I would start with the index first.

Anyway, I do hope you find this book useful. Let me know what you like or dislike and we'll make it better next time!

Have you ever wondered...

Why the ruler on your screen is not exactly one inch? See *resolution.*

Why you keep getting the message that you are "out of memory" when you know your hard disk is not full? See *RAM.*

How to install a card yourself? See *add-in board.*

If you should install Windows on your PC? See *Windows.*

When mixing red and green makes yellow? See *additive color.*

Why some fonts look smooth on your screen and some don't? See *font.*

Why you don't have an option in your Memory Control Panel to use virtual memory? See *32-bit clean.*

Where those alternate sorts of characters are that you see in other people's documents? See *Key Caps (Mac)* or *Character Map (PC).*

Why some fonts have city-names and some don't? See *fonts (city-named).*

What the difference is between **lines per inch** and **dots per inch**? See *lines per inch.*

Why small letters are called "lowercase"? See *lowercase.*

How to giggle online? See *baudy language.*

Why you should never move your hard disk while your computer is on? See *park.*

What this means when you see it in a menu: … ? See *ellipsis.*

Who is the best publishing house in the world? See *Peachpit Press.*

What is the important and significant difference between round radio button and square checkbox buttons? See *checkbox.*

What my older brother looks like? See *propeller head.*

Who really owns your software (it's not *you!*)? See *license agreement.*

Why you can't type a colon in a file name? See *illegal character.*

What that little piece of plastic is that came in the box with your Macintosh (don't throw it away!)? See *programmer's switch.*

If a monitor has only three colors in it (red, green, and blue) how does it show black or white? See *RGB.*

How to take a picture of the screen? See *screen capture.*

Why photographs are made of dots when they come out of your printer? See *halftone.*

What those flashing lights on your modem box mean? See *modem.*

How to recover as much as possible when you crash? See *Ctrl-Alt-Del (PC)* or *force quit (Mac).*

How a computer gets PMS? See *Pantone Matching System.*

When a nice arpeggioed chord means Big Trouble? See *Chimes of Doom.*

How to make a movie play when you turn on your Mac? See *Startup Movie.*

Why ATM makes type so smooth on your screen and even on your non-PostScript printer? See *fonts (ATM)*

Who lives in Silicon Forest? See *Silicon Forest.*

How to use the built-in screen saver in Microsoft Word? See *screen saver.*

What to look for when shopping for a VGA monitor? See *VGA.*

If you're using System 6 or 7? See *System 7* for the instant clue.

What to do if things on your computer start acting a little weird? See *virus.*

How to use the keyboard to select and activate buttons? See *Tab key.*

Where to find men's toggle switch? See page 595.

What kind of software can create side-by-side paragraphs easily? See *table*.

What are the really important things to know? See *Three Rules of Life*.

Where all those acronyms come from? See *TLA*.

What is the difference between **margin** and **indent**? See *indent*.

What to do if your Mac is sad? See *Sad Mac*.

Why you can't connect two computers to the same external hard disk? See *SCSI address*.

If you are guilty of using the single most visible sign of unprofessional type? See *smart quotes*.

Why you shouldn't rub balloons on your head while sitting at your computer? See *antistatic device*.

How to make any Mac document a stationery pad? See *template*.

How to prevent your application from crashing so much due to lack of memory, even if you have lots of memory available? See *application heap*.

How to communicate the subtleties of your online message (wink, wink) since the other person can't read your body language? See *baudy language*.

How to get the squiggle over the n in piñata? See *tilde*.

Why we call starting the computer, "booting"? See *boot*.

How to send a file to someone else in a format that you are sure their program can open, but that will save some of the formatting, like bold and italic and font choices? See *Rich Text Format*.

What is the shortcut to selecting and activating buttons in Windows? See *buttons*.

How to get numbers from your document into the Calculator, and from the Calculator into your document? See *Calculator*.

How to make your Mac tell your lover sweet things? See *Startup Sounds*.

How Caps Lock is different from the Shift Lock on a typewriter? See *Caps Lock*.

What tips experts give to help prevent straining your wrists? See *carpal tunnel syndrome*.

What creature says, "Moof!"? See *Dogcow*.

Why you shouldn't compress your only copy of an original file? See *compress*.

If there really was a Murphy who pinpointed one of the laws of life? See *Murphy's Law*.

If there is proof of who really runs the show? See *motherboard*.

The simple trick to killing the WDEF virus? See *WDEF*.

How to speed up your computer a little bit for absolutely free? See *rebuild the desktop*.

How this book was created? See *desktop publishing* and page 674.

What makes some ROMs dirty and some clean? See *32-bit addressing*.

How a laser printer actually works? See *laser printer*.

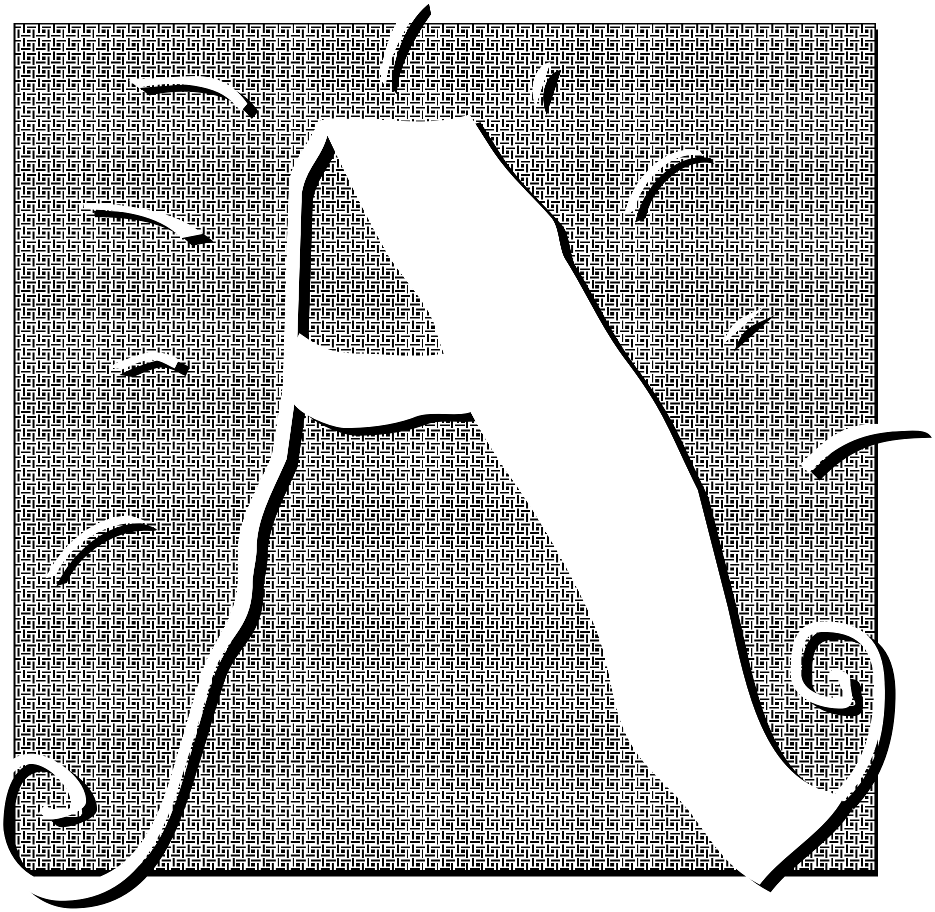
What is the instant clue to tell whether your Macintosh fonts are downloadable or not? See *downloadable font*.

What is the problem with being too close to the front, sides, or back of a monitor? See *ELF*.

What happens late at night on computers across the country? See *electronic sex*.

In this work,
when it shall be found
that much is omitted,
let it not be forgotten
that much likewise
is performed.

Dr. Samuel Johnson
upon completion of his dictionary, 1755



Abort, Retry, Ignore, Fail?

DOS displays an ominous *error message* like **Abort, Retry, Ignore, Fail?** whenever it has trouble using one of your disks or files. Why it has to be so brusque and frightening, I have no idea, but a message like this usually doesn't mean the end of the world. Here's a quick guide on what to do:

1. If you get the error message when you're trying to use a floppy disk, first be sure that the disk is all the way in the disk drive, and that the drive door (if there is one), is closed. If not, fix things, and then press **R** (for retry). If that doesn't solve the problem, go to Step 3.

2. If you tell DOS to switch to a floppy disk (by typing, say, **B:** followed by the Enter key), but there's no disk at all in the floppy drive, you'll get an error message. If the message includes the Fail option (older versions of DOS don't have the Fail choice), press **F**. That should give you a prompt something like:

```
CURRENT DRIVE IS NO LONGER VALID>
```

Just type the name of a valid drive, like **C:** followed by the Enter key. Unless you put a disk in the drive, don't press **A** (for Abort)—if you do, DOS will just keep trying to get the floppy drive in gear, which it can't do without a disk.

3. If the error message appears when you're trying to copy or open a file, it probably means that DOS is having trouble, big trouble, reading the information on the disk. You may have lost some of the file, but maybe not all of it. In this situation, start by pressing **R** for Retry at least a few times. DOS may be able to find the correct information if you keep trying. Eventually, if the error message just keeps reappearing, you may want to press **I** for Ignore, which tells DOS to go ahead with copying the file or whatever despite the problem. The new copy will have gaps in it, but at least some of the information may be intact.

4. What you do after completing Step 3 depends on what kind of disk produced the error message.

   If it's a **floppy disk,** assume the disk is bad. Copy all the files you can to a new floppy. If important files that didn't copy correctly, you might try taking the disk to someone else's computer and copying them there. After you've made your copies, throw the bad disk away.

   If it's a **hard disk,** you're in trouble. Don't use the disk until you can get help from somebody who really knows what they're doing, or use a *utility* like Norton Disk Doctor to diagnose and fix the problem.

About

In most Macintosh and Windows programs, you can get a little background information about the software by choosing the **About** menu item. You'll see a box containing tidbits like the full name and *version number* of the program, who wrote it, and perhaps the name of the person it's *registered* to.

PC In Windows programs, "About . . " is usually the final choice on the *Help menu.*

On the Mac, "About . . ." is the first item under the *Apple menu.*

While the About window can be extremely boring, some programs use it to deliver rather creative commercials, with glitzy graphics and snazzy special effects. Sometimes if you hold down a *modifier key* (such as the Command or Control key or the Option or Alt key*)* as you choose About, you'll get a surprise (called an *Easter egg*).

For instance, at the *Finder* in System 7, hold down the Option key and choose "About The Finder" from under the Apple menu.

absolute cell reference

An **absolute cell reference** is a reference to one specific cell in a spreadsheet or sometimes in a database. It's easiest to explain in relation to a "relative" cell reference. In a spreadsheet, each cell has a name (like A4 or B6, kind of like a Bingo card). In formulas, you need to tell the computer which cell to act on, or which cell to refer to. A **relative** cell reference tells the spreadsheet to use the information in a cell that is in a **relative** position to this one. For instance, you may want the data in cell A5 to be divided by the data in cell A3. **You** may think you are telling A5 to divide by the cell A3, but the **computer** is hearing that you want to divide by the cell that is **two spaces above** A5. If you took the same formula from A5 and copied it to cell B5, the computer would divide by the data in cell B3, because it is still referring to **two spaces above,** relative to wherever the formula is. This is great because then you don't have to retype formulas all the time, you can just copy and paste them.

But sometimes you don't want the cell to be referenced to wherever the formula happens to be. Maybe in cell B5 you really do want the data to be divided by what is in A3, not what is in B3. Then you have to make cell A3 an **absolute** cell reference, so no matter where you copy the formula, the computer will always divide by the data in cell A3, absolutely.

The *default* (or automatic) formatting for a cell reference is usually relative, meaning if you want an absolute reference you have to tell it to be so. Different programs have different ways of making a cell reference absolute. A common method of specifying a cell to be absolute is typing a dollar sign ($) before the cell name: $A3.

accelerator, accelerator board

An **accelerator** is something you can install in your computer that makes the machine work faster. Almost always this is a **board** or *card,* a thin rectangular piece of fiberglass with a bunch of electronic parts and circuits attached to it. You install the board (or have someone else install it) into your machine—it may be designed to plug into one of the computer's *expansion slots,* or it may fit somewhere else. The accelerator board has a *microprocessor (chip)* of its own that takes over some or all of the work from your computer's original *processor.* You may actually have to remove the existing processor when you install the accelerator.

Adding an accelerator board is often an inexpensive and efficient way to upgrade your computer without having to buy a whole new system. But shop carefully—you can spend a few hundred dollars and end up barely noticing any speed boost.

access

To **access** an item (such as the information in a file) just means to use it. But the term implies that you have to do something first in order to be able to use whatever it is. You may need to switch to the folder or directory where the file you want to access is stored. You may need to enter a password before you can access it. If you can't access an item, it means you haven't told the computer where to find it, or you don't know the password, or maybe the item is *locked*.

access code

An **access code** is another term for a *password*. An access code is usually a specific combination of letters and/or numbers. You always need some sort of access code to use an *online service* or to have your computer talk to another computer.

access time

You'll usually see the term "average **access time**" in advertisements or reviews of hard disks and *CD-ROM* players. Access time refers to how fast the disk can locate and begin retrieving (accessing) a specific piece of information. Because the time needed to access any particular piece of data varies depending on where it's located on the disk, vendors use an average access time when rating the speed of their drives. Access time is measured in *milliseconds*, or *ms*, so ads say things like "average access time: 18 ms."

If you're buying a hard disk, you want one with an average access time of less than 20 milliseconds; don't settle for anything slower than 30 milliseconds.

CD-ROM players are much slower; the fastest now available have average access times of around 360 milliseconds (it takes them about a third of a second to find, say, the definition of the word "nugatory" in a CD-ROM dictionary).

Acrobat™

Acrobat is a technology developed by Adobe Systems that allows documents created on one computer system to be read and printed on other systems, with the fonts, formatting, text attributes, and graphic elements intact. It works in tandem with the *Multiple Masters* font technology, in which a font that exists in one computer can emulate the font in the document that was created in another computer system.

This technology will play a key role in the future of computerized communications, allowing us to send compact, well-designed, attractive, and readable documents from one computer to another, and even from one type of computer to another, without using paper. There are three key elements in this technology that make it outstanding: preserving the formatting and spacing of the original font (have you ever opened a document on your computer without the correct fonts? aaack!); the compactness of the file (generally files are quite large that can hold onto graphics and formatting); and the fact that it will work on virtually any computer system.

active

The term **active** refers to the item in use at the moment, whether you are talking about such diverse things as a hard drive, a window, or a cell in a spreadsheet.

For instance, if you have more than one window open on your screen, the window that is in front and that has lines in its *title bar* is the window that is active (see *active window*). In a *spreadsheet* or *database*, the cell or the field you selected is active. In a *dialog box,* the *edit box* that is highlighted or that contains the *insertion point* is active.

When an item is active and you do something like choose a command or type, **it will happen to the active item.** That's the whole point! For instance, if a row is active in your database and you choose the command Cut, that row will disappear. If a window is active and you create a new folder, the new folder will appear in the active window.

You may have several *hard drives* and/or several *floppy drives* (the thin slots where you insert your disk) attached to your computer, with or without disks in them. The computer has to go to one of these drives to get the information you need from the disk. The drive the computer is using at the moment is the **active drive** (compare with *boot disk*).

active matrix display

An **active matrix display** is the kind of screen used on the more expensive laptop and notebook computers, including the Macintosh PowerBook 170 and an increasing number of PC notebooks. Active matrix screens are a type of liquid crystal display, or *LCD*, like the vast majority of screens on laptop and notebook computers. But unlike the garden variety LCD screen, an active matrix screen has a separate, independent transistor for each and every dot *(pixel)* on the display. The result is a sharper image and much brighter, more lifelike color, with less of the *smearing* that makes moving images hard to see on an LCD screen. Conventional, or passive matrix, LCDs still have one big advantage: they're much cheaper than active matrix displays.

active window

Although there can be many *windows* open on the screen, only one window can be *active,* or in use, at a time. It's easy to tell which window is the **active window:** the title bar at the top of an active window looks different from the title bar of an inactive window. When you choose a command, such as "Close," the command will affect the active window. If you choose to make a new folder, the new folder will appear in the active window. To make a window active, simply click on it.

In Windows applications, the active window's title bar has a distinct color, different from the other title bars.

On the Macintosh, the active window shows horizontal lines or perhaps a pattern in its title bar, whereas the title bar of an inactive window is empty (plain white). To move a window without making it active, hold down the Command key while you *drag* its title bar.

This is an active window. *This window is not active.*

A-D conversion, A/D conversion

See *A-to-D conversion.*

adapter

An **adapter** is electronic circuitry that adapts a device, such as a monitor, to your computer so that the computer can control what the device does. For example, a *VGA monitor* requires a VGA *(video graphics array)* adapter in your PC or you won't see any images on the screen.

An adapter can be an *add-in board* (*card)* or it may be built into the circuits on the computer's *motherboard.* A *controller* does exactly the same thing as an adapter, but the two terms tend to be used differently: the term *adapter* is usually used in reference to monitors and printers; the term *controller* is usually used in reference to disk drives and scanners.

ADB

ADB stands for **A**pple **D**esktop **B**us™, a system that uses certain hardware and software for connecting *input devices.* You use input devices (like mice, keyboards, graphic tablets) to *input,* or put in, information to the Macintosh. Older Macs, like the 512 or the Plus, don't use the ADB (they use an older and different system), which is why you can't interchange older keyboards or mice with newer ones. Also see *ADB port* and *bus.*

ADB port

This icon represents an ADB connection.

A *port* is a little socket on the back of a computer or any other *peripheral device,* such as a keyboard or an external hard disk, where you plug in a cable. An **ADB port** is the one with the diagram, or *icon,* that looks like little worms (shown to the left). *ADB* stands for **A**pple **D**esktop **B**us™.

Input devices, such as keyboards and mice, plug into ADB ports. You can plug any cord that has that little icon on its end into any ADB port. For instance, you can plug your mouse into the extra ADB port on either side of your keyboard. In fact, if you prefer a mouse and your roommate prefers a trackball, you can have the mouse connected to the ADB port on your keyboard and the trackball connected to the ADB port on the back of the computer and use them both. Older Macs, like the Plus and older, do not use the Apple Desktop Bus.

add-in board, add-in card

Add-in boards or **add-in cards** are circuit *boards* or *cards* that you insert into slots designed for them inside your computer. An add-in board consists of electronic circuits (*chips,* other parts, and the connections between them) mounted on a rectangular piece of fiberglass. The bottom of an add-in board has a row of metallic contacts on a smaller rectangular projection; you insert this part into the slot (see the illustration in the appendix).

Each board you add imparts some new capability to your computer. You can get an *accelerator board* to make your computer run faster, a "video board" to control your monitor, a "memory board" to add more memory, etc.

Opening up your computer to plug in an add-in board can be a little scary the first time you try it, but it's hard to goof up the procedure. Here are a couple of tips:

1. Before you handle the board or any of the computer's electronic parts, touch a piece of metal to discharge any static electricity that's built up on your body—inside the computer there should be something perfect for discharging the static, like a metal mounting bracket or the *power supply* case.
2. The board can only fit one way, but just to be sure, check that you've oriented it so that the chips face the same direction as the chips on the other boards.
3. Finally, use a gentle rocking motion to seat the board into the slot all the way down—you should feel a satisfying thunk when the board sinks all the way into the slot.

additive color

Additive color is produced by blending different colors of light to make the color you want to see. Color (*RGB*, red/green/blue) computer monitors and television sets use additive color. A monitor isn't like a painting, in which the colors you see are due to light being reflected off the painting to our eyes. Instead, the monitor generates light of its own that shines out from the screen, directly to our eyes.

Remember that white light is made up of all the colors in the spectrum, which you can see separately if you break it down using a prism. Conversely, if you blend equal amounts of red, green, and blue light, the colors combine to create white light. And blending red and green light together make yellow. Yes, yellow. Adding colors of light together always makes a lighter, brighter color—pure blue plus pure green make sky blue, for instance.

But we're not used to working with color this way. Most of the color we see in nature comes from reflected light as *subtractive color*. When we look at a red apple, we are seeing light that has hit the apple and bounced back towards us. The yellow and blue parts of the spectrum colors of light have been absorbed by the apple—they've been "subtracted" from the original white light. Combining subtractive colors—colored paints, for instance—always produces a darker shade, because each color subtracts more of the spectrum. If you mix red and green paints, you get a brownish or olive green, not yellow. The red paint subtracts some of the green light, and the green paint subtracts some of the red light. The only reason you don't get black when you mix red, green, and blue paint is that the paints are not 100 percent pure in color—they all reflect a little light in other parts of the spectrum.

add-on, add-in

An **add-on** is a piece of *hardware* that you can add to your computer system, such as an extra *hard disk*, a *modem*, or a *scanner*. The term add-on usually refers to hardware (the equipment you can ~~kick~~ touch). Additional *software* that enhances your existing software, such as report generation programs for enhancing your database or tiny *chips* that you add to your computer, are sometimes called **add-ins.**

address

An **address** on your computer is similar to the address on your house—it's a way for the computer to know where to send its messages, and a way for the information or the *device* to know it is being called upon.

SCSI address

A SCSI (pronounced "scuzzy") address is a distinct address with a number from 0 to 7 used for SCSI devices, such as hard disks and printers. The Macintosh itself always has a SCSI address of 7; the internal hard disk always has a SCSI address of 0 (zero). See *SCSI.*

memory address

The computer actually works only in *memory.* It takes information out of the hard disk and puts it into memory so it can work. Once the data is in memory, the computer needs to know where to get it and how to send messages to it. The process of finding the information that is tucked away is called **addressing.**

Let's say you have 4 *megabytes* of *RAM (random access memory)* installed in your computer. These four megabytes are divided up into little cubicles, and each of these cubbies can hold one *byte* (one little piece of information, or data). Four megabytes is 4,194,304 *bytes*, which means 4,194,304 little cubbies. Each of these many cubbies has an **address** in this **memory** identifying it so the computer knows where to find or send the information. See *32-bit addressing.*

network address

Within one network, each *node* (each device, such as a printer, computer, or *server* is a node) has its own address so the information buzzing through the lines knows where to go. If several *networks* are connected together, each network has its own address as well.

online address

When you send mail through your computer, you need to know the other person's *online* or "screen" name, which is considered their "address." On some *online services* the names are easy to remember, like Toad Hall or

Moody River. But on other services, the addresses are gawdawful combinations of numbers and letters, like D03H1,45337. Most services offer a way to create a little electronic address book of your favorite correspondents. You can usually apply the address to your mail by clicking a button so you don't even have to type it.

Adobe Font Metrics

See *font metrics.*

Adobe Type Manager®

Adobe Type Manager (affectionately known as ATM) is a *utility* (a little software program) that uses *Type 1 PostScript fonts* (a certain variety of typefaces) to display type on the screen. No matter what type size or style you select, ATM will create the smoothest on-screen rendition that the *resolution* of your display screen allows. This makes it much easier to work with type on the screen, especially if you want to do things like *kern* your type or design pages. Also see *fonts.*

This is what type looks like on the screen without ATM.

This is what type looks like on the screen with ATM.

Like other *scalable fonts,* Type 1 fonts consist of mathematical formulae that describe the outline of each character. ATM takes this outline information and *scales* and *rasterizes* it, figuring out which dots your screen should display to create the smoothest and sharpest rendition of the character, at whatever size you requested (*scaling* refers to the process of creating the final character at a particular size; turning an image into dots is called *rasterizing*).

ATM can also rasterize a Type 1 font to your **printer** so that the text prints smooth and clean on a *non-PostScript printer* (such as an Apple StyleWriter, an HP LaserJet, DeskWriter, or DeskJet, or an Epson dot matrix printer). *PostScript printers* can rasterize Type 1 fonts themselves, so they don't need ATM. But no matter what type of printer you have, ATM always displays Type 1 fonts beautifully on the screen, no matter what size you've chosen.

ATM is available for both the Mac and Windows, and even for some DOS (non-Windows) programs. Now that System 7 and Windows 3.1 come with *TrueType* built in, you can get the benefits of scalable fonts on any screen or any printer without ATM. But TrueType only works with TrueType fonts—if you are using Type 1 fonts, you'll still need ATM.

AFM, AFM files

Those files on your font disks called **AFM** files hold the *Adobe Font Metrics* (on PC font disks they have the file extension .AFM). See *font metrics.*

After Dark®

After Dark is a *utility* from Berkeley Systems, called a *screen saver,* but After Dark goes far beyond most other screen savers. The idea behind a screen saver is to keep your screen dark, or keep a pattern moving, so an image doesn't get burned into the screen by staying in one place too long. Now, you **could** just dim your screen, but that would take a manual effort. You could be content with little stars moving around space, so at least you knew your computer was still on and functioning. But no—you want a little man mowing the lawn on your screen. Or fish and seahorses floating in full-color with an occasional scuba diver and flying toaster drifting through. In fact, you want an entire fleet of flying toasters because you like the gentle rhythmic sound of their little wings beating.

After Dark has an endless collection of screen-saving modules whose only purpose in life, besides protecting our screens, is to delight us. What a noble concept. Actually, once you buy the After Dark utility, you can get hundreds of modules free from any *bulletin board* or *online service,* created by other nice people who want to share the delight.

AI

See *artificial intelligence.*

alert box

An **alert box** is a message that appears on the screen to warn you of some imminent disaster, or to inform you of a situation in which your actions may have irreversible consequences. For example, if you've just chosen the command to delete a file, an alert box might warn you that the file will be gone forever, and ask you if you *really* want to delete it. You have to deal with the alert box before you can move on with your life, meaning your only option when you see something like this is to click one of the buttons it offers you. Typically the dark-bordered button (which is the *default* button) is the safest option available—and because it's the default, you can hit the Return or Enter key instead of clicking the button. If the alert is telling you the computer *crashed,* then the button may not even work, in which case you just have to turn off your machine by its switch or, if you have one, press your *reset switch* (see *programmer's switch*).

This is an example of an alert box.

algorithm

An **algorithm** is a set of rules that is supposed to give the correct answer for solving a particular problem. The rules in an algorithm must have a beginning and an end and they must be well-defined with no chance of ambiguity. A computer can't solve a problem until someone programs it with the algorithm for solving that problem.

We use algorithms in our lives all day long to get things done, from running the dishwasher to balancing a checkbook. Aldus PageMaker uses algorithms similar to those we were taught in school for hyphenating unknown words.

Unfortunately, there is no algorithm for making human relationships work; no, we tend to use the *heuristic* method, which is simply trial-and-error (supposedly **intelligent** trial-and-error).

alias

This is an example of an alias.

An **alias** is a representation of the real thing. System 7 on the Mac has a wonderful feature enabling you to create aliases of any file to aid with organization and convenience. When you create an alias of a file (any file), you are essentially creating a "gofer" (you know—go fer this, and go fer that). The alias is just an icon (using only 1 or 2 K of space) that knows where the real thing is. When you double-click or choose an alias, the alias goes and gets the **original** file and opens it. For instance, you can create an alias of MacWrite, and put the *alias* in your *Apple menu.* Then you can open MacWrite straight from the Apple menu rather than dig through folders to get to it, because when you choose the alias, that alias goes and finds the real MacWrite and opens it.

Why would you want to do this? One reason is that many applications have to be in the same folder as their accessory files, like their dictionary or preferences file, or they won't open. An alias frees you from having to open folder after folder to get to your application. Also, you can have as many aliases as you like of each file, so you can store aliases all over the place, wherever you would find it convenient. You can leave aliases of your favorite applications right out on the Desktop and use the *drag-and-drop* technique for opening documents.

To make an alias in System 7, select a file and choose "Make Alias" from the File menu. The name of an alias is always in italics (see the example, previous page), which gives you an instant visual clue that the file is an alias.

On a PC, Microsoft Windows' Program Manager lets you create icons that function similar to the aliases described above. If you drag a document icon from File Manager into a Program Manager group window, it will create a new icon in Program Manager. Double-click to launch the document's application and the document itself. You can freely rename the icon, and move or copy it to another group, but Program Manager keeps track of where the actual files are located.

Under MS DOS you might have to type some long, cryptic command at the *DOS prompt* just to copy a file from one disk to another. You can, though, create a DOS *batch file* that lets you lump a lengthy command or group of commands under a single, simpler batch command. A batch file can essentially function as an alias. For instance:

instead of typing:	COPY C:\WP51\DOCS\THISWEEK.DOC A:\BACKUP\
you could type:	THISWEEK

The MS DOS User Guide and Reference covers the use of MS DOS batch files. Certain utility programs such as *4DOS* or NDOS (which comes with Norton Utilities) or CED also help you create aliases.

aliasing

Aliasing has two definitions, depending on whether you're talking about pictures or sounds.

When a diagonal line or a curved arc drawn on the screen looks as if it was made out of bricks, when it looks like stair steps instead of a slide, the effect is technically called **aliasing.** Most of us would say it had the *jaggies.* Also see *anti-aliasing.*

Aliasing is also a type of sound distortion you can hear. It affects *digitally* reproduced sound, which is the kind of sound your computer probably makes. (The beeps you hear from a standard PC speaker aren't digital, but a sound board you plug into your PC creates sound digitally; the Mac has built-in digital sound.)

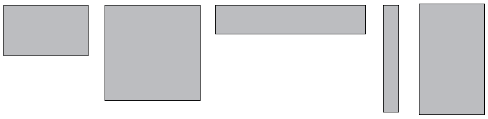
Digital sound is based on a sequence of numbers (digits) that are converted into sound waves by electronic circuits. The computer has to guess at what sound to make between each number in the sequence. If the time between each value is too long (if the "sampling rate" is low), you hear the mistaken guesses as a metallic, static distortion called aliasing. To squelch aliasing, you need a sound *card* with a sampling rate of around 40 *kilohertz* (40,000 times a second) or higher.

alignment

If you're talking about text, **alignment** refers to whether the text in a paragraph is lined up on the left or the right or centered or on both sides. For instance, in a flush left alignment the text is aligned on the left edge and is *ragged,* or unaligned, on the right. The paragraph you are reading right now has a *justified* alignment—the text is aligned on both sides.

This paragraph, however,
is centered.

And this paragraph is fiush left. If you're talking about graphics, alignment means just what you'd think: the relative position of two or more graphic items, or *objects,* on the screen. Many *drawing programs* have an alignment command that automatically aligns objects so, for instance, their left sides all line up along the same imaginary vertical line, or their centers all fall on the same imaginary horizontal line.



These objects are aligned across their top edges.

alphanumeric characters

The term **alphanumeric** refers to the characters you can type and print, such as the entire **alphabet** and all the **numbers**, as well as punctuation. You have keys for *Return, Tab, Alt, Option*, etc., but these are *non-printing characters* so they're not considered part of the alphanumeric group.

alpha channels

If you stumble upon a group of power users who are heavy into graphics, chances are you're gonna hear a lot about **alpha channels.** Alpha channels are pretty incredible, but it can take a long time to get the hang of them and put them to practical use.

First, consider that the Macintosh has the capability to create *24-bit* images, but does so with a technology called *32-bit QuickDraw.* So what happened to the leftover 8 bits? Those are left open for the use of alpha channels. In a 24-bit *RGB* image, the red, green, and blue channels each take up 8 bits, leaving another 8 bits open for software developers to use for special effects, masks, compositions, etc.

The most well-known application that makes use of alpha channels is Adobe Photoshop, which works even more magic by allowing you to have up to thirteen separate alpha channels (of 8 bits each!) in addition to the 24 bits used by the red, green, and blue channels. Each of these channels is a separate entity which can be modified in numerous ways without affecting the entire image. Much like we've always had in draw programs, Photoshop now gives us the ability to accurately alter many separate parts or elements of a *bitmapped* image without disturbing the nearby elements.

Think of the concept as a layering system where you can have, say, a sailboat in one layer, a lake in another layer, and a mountain range in another layer. Each of these three items can be placed in alpha channels and manipulated (colored, scaled, rotated, whatever) independently of all the others. And, if you want to manipulate the whole image, you can do so by making a composition of all three with a host of options as to how they relate to each other.

Alpha channels are a very, very cool technology, but plan on spending some time learning how to use them. If graphics are your thing, it's worth the time and effort.

alpha testing

Before being released to the public, a product goes through extensive in-house testing, often by the developers themselves. This is considered **alpha testing.** After alpha testing the product is sent to selected outside *beta testers* who find all those *bugs* and problems that the alpha testers missed.

It is sad but true that testing only uncovers the **presence** of bugs, not the absence.

alpha version

An **alpha version** of software is the first working draft. In an alpha version, all the features that are going to be in the finished product are supposedly in place and functional, but it is known that there are also *bugs.* The alpha version is tested in-house by the developers and also sent out to "alpha testers"—people who have been selected or who have shown an interest in playing with software that is almost guaranteed to *crash* their computer. After the alpha version comes one or more *beta* *versions*, and when these have been tested and had their known bugs fixed, the software is *mastered* and comes out in the "release," also called the shipping version.

Alt key

The **Alt key** on a PC is most often used in conjunction with other keys to activate commands. In programs that have menus, pressing Alt with another key is a *keyboard shortcut,* an alternative, quicker way to carry out one of the menu commands.

In some programs, including Microsoft Windows and DESQview, pressing Alt by itself does something too. In Windows, this activates the menu bar; in DESQview, it pops up the window you use to control DESQview. The Alt key's equivalent on a Macintosh is the *Option key.* Please see *modifier keys.*

America Online

America Online is an *online service.* This means it's a service business where you can do research, get the latest news, join clubs, make hotel and plane reservations, shop, meet and chat with people, attend conferences, send and receive mail, and thousands of other things, but it is all done *online* while you sit at your computer.

You need to have a *modem* and you need to have the America Online software, which is free. You access **AOL** (as it is affectionately known) by having your computer dial a local number (which to you is just a click on a button). AOL charges you $10 an hour between 8 A.M. and 5 P.M., and $5 an hour after 5. They take it right out of your checking account or add it to your credit card. It is too simple. And addictive. Everyone ends up spending hours of their life online when they first connect with AOL.

Amiga

Commodore Business Machines makes a series of personal computers called **Amiga,** ranging in use from home applications to business use and desktop publishing. Their higher-end Amiga computers are very popular for video work and animation. Standard Amigas aren't compatible with either the IBM PC or the Mac (they won't run PC or Mac software), but Commodore sells an *add-on* that makes the Amiga IBM-*compatible.*

ampersand, &

This symbol, **&**, is called the **ampersand**. The symbol stands for the word "and," and the word is a contraction of "and per se and." The ampersand has slightly different shapes in different typefaces (some of which show clearly the Latin "et" for "and"): &, *&*, or **&**, among others. Press Shift 7 to type the ampersand symbol.

In some programming languages, the ampersand is used to *concatenate,* or join, items such as words and phrases.

analog

Analog is the opposite of *digital,* and I can only explain analog in relation to digital. Analog refers to things that are in a continuous flow or that have an infinite number of values—things that are "analogous" to real life.

For instance, a watch with hands is an analog device, because as the hands sweep over the watch face, time can be shown in a continuous flow, with an infinite number of possible increments. A digital watch, on the other hand, displays time in finite chunks—seconds or tenths of seconds or perhaps even smaller chunks, but at any one moment there's one specific number showing, always countable and limited.

A photograph represents a scene or a face in analog form, in continuously varying tones of grays or colors. When you *scan* a photograph to get it onto your computer screen, the scanning process breaks up those continous tones into *digitized* (digital) *bits* of information, because your computer can only understand digital things.

Guy Kawasaki defines analog as "anything you can read in direct sunlight."

It is possible to make an analog computer, but digital computers are much easier to build, faster, and more reliable. Also see *digital* and *modem.*

ANSI

ANSI is the acronym for the **A**merican **N**ational **S**tandards **I**nstitute. This institute creates standards for a wide variety of industries, including computer programming languages.

As an *end user* (someone who uses the computer to get their work done, rather than programming it), you may still hear of ANSI when someone refers to the *ANSI character set.* See the character set on the following page.

ANSI character set

The **ANSI character set** is the *character set* (collection of the characters you can type in a chosen typeface) adopted by the *ANSI* organization as the standard for computers. (In real life, it isn't a standard, but it is the character set used by Microsoft Windows.) Like any character set, ANSI comprises a particular collection of letters, numerals, punctuation marks, and other symbols, each of which is assigned to its own code number. Included in the ANSI set are typographic quotation marks and dashes, business symbols like copyright marks, and lots of foreign language characters.

Character	Name	ANSI code	Macintosh keystrokes
…	ellipsis	0133	Option ;
‘	opening single quote	0145	Option]
’	closing single quote (apostrophe)	0146	Option Shift]
“	opening double quote	0147	Option [
”	closing double quote	0148	Option Shift [
•	bullet	0149	Option 8
–	en dash	0150	Option Hyphen
—	em dash	0151	Option Shift Hyphen
™	trademark symbol	0153	Option 2
©	copyright symbol	0169	Option g
®	registered trademark	0174	Option r
¼	one-quarter	0188	depends on font
½	one-half	0189	depends on font
¾	three-quarters	0190	depends on font

In Windows, to use characters in the ANSI set that aren't on your keyboard *(extended characters),* hold down the Alt key while you type the four digit code number (using the *numeric keypad*) for the character you want, as shown in the table above. This number always begins with 0 (zero). For instance, pressing Alt and typing 0233 would give you **é**. Another way to use the extended characters is to run the Windows utility *Character Map.* In the chart it displays, click on the character you want, then choose Select, followed by Copy to copy it to the Windows *Clipboard.* Then switch back to your program and choose Paste.

ANSI.SYS

ANSI.SYS (pronounced "ansee dot sis") is a *driver,* a little software module, that you can use with a PC running *MS-DOS.* You "install" ANSI.SYS by running it from your *CONFIG.SYS* file. With this driver installed, you can use many new commands, giving you more control over your screen and keyboard than you have using standard MS-DOS commands.

anti-aliasing

When text or a graphic image is displayed on a *monitor,* or screen, the smoothness of the edges is limited by the resolution of the screen, which means the edges tend to be a little jagged. This jaggedness is also called *aliasing.* There are a variety of techniques used to reduce the jaggies, or the aliasing of text and graphic images, to fool our eyes into thinking the edge is smoother. For instance, in an *image editing* program you can blur the edges, or shade along the lines to make the dark-to-light transition less distinct. **Anti-aliasing,** then, means to use one of these techniques to smooth out the rough edges (the *aliasing*).

antistatic device

If there is too much static electricity hanging around, it can actually disrupt your computer, causing the screen to freeze or creating various other unpleasant disturbances that can destroy data. Static can even destroy the circuits inside your computer. Extra static can develop from the weather or from certain kinds of clothing or from activities like petting your cat while working on your computer, shuffling around on the carpet, or rubbing a balloon on your head. So a variety of **antistatic devices** have been developed to help prevent this, devices such as wrist bands, floor mats, sprays, and little metal pads that say "Touch Me."

It's particularly important to discharge any static on your own body when you open your computer to install another *card* or perhaps a *SIMM* or two. You don't need a special antistatic device to do this—just touch something metallic before you handle the circuits. Most computers have a metal plate inside the box that's perfect for this purpose.

Any key

There really isn't a key called the **Any key.** When a direction tells you to "Press Any Key," it simply means to press whichever key you like on the keyboard.

AOL

See *America Online.*

APDA

APDA stands for **A**pple **P**rogrammers and **D**evelopers **A**ssociation. It's a support service from Apple Computer that provides documentation and technical products for people who program or develop Apple equipment.

Apollo Computer

Apollo Computer is the name of a company (now owned by Hewlett-Packard) that creates high-powered *workstations.* (A workstation is a bigger computer than we use at home and is usually specialized for scientific, engineering, or graphics applications; see *workstation.*) Apollo developed the Network Computing System which allows people to *network,* or connect, computers from different manufacturers.

append

When you **append** something, you add it on to the end of something else, such as to the end of a document or to the end of a list. This is different from "inserting" an item, which is when you add an item into a specific position in the list, not necessarily at the end.

An example of using a button or command called "append" is when you have a list of *fields* in a *database* that you want to export. When you append another field to the existing list, the one you append will be at the bottom of the list. Since the order of the exported fields is critical, be conscious of whether you need to insert or append.

Apple computer

An **Apple computer** is one of the computers made by Apple Computer, Inc. that is called "Apple," such as the Apple II, Apple IIGS, or Apple IIE. We do not call a *Macintosh* computer an Apple. An Apple and a Mac are very different machines with different *operating systems.* In fact, the Apple operating system has more in common with the operating system used by *IBM PCs* than the one used by the Mac, and Apples and Macs cannot use each other's software without special adaptations. Being able to use an Apple does not mean you automatically know how to use a Mac, and vice versa.

These days, Apples are found mostly in elementary schools and in the homes of educators since these folks were able to buy them cheap. Apple (the company) no longer sells them.

Apple Computer, Inc.

Apple Computer was founded by Steve Wozniak and Steve Jobs in a garage on April Fool's Day in 1976 and was guided by Mike Markkula. Later that year Apple introduced the Apple I at the local *user group,* the Palo Alto Homebrew Computer Club. In 1977 they introduced the Apple II, which became the most widely used computer in the classroom and home. *VisiCalc,* the first spreadsheet on earth, was launched on the Apple II, which took the computer into the business market. The forerunner of the Macintosh computer, the Lisa, was introduced in 1983 but didn't last very long—it was slow and expensive (for a while they tried to turn it into a hybrid Mac). And of course, in 1984, the Macintosh was introduced.

Apple popularized the user *interface* invented at Xerox *PARC* and this interface has since influenced every other personal computer development. By the fourth quarter of 1992, Apple was selling more Macintoshes than IBM was selling PCs.

Apple Desktop Bus™

See *ADB.*

Apple Events

Apple Events is the name of a System 7 feature that allows two-way communications between separate applications, either on a single Mac or through a *network.* For instance, while you are working in one program you can tell another program to do something. Apple Events is a component of the *Inter-Application Communication* protocol that lets applications share features with each other. Many applications are being upgraded to take advantage of this capability.

Apple File Exchange

Apple File Exchange is a great little *utility,* or small program, on the Macintosh that allows you to convert a *PC* file into a Mac file. It's so cool—you just double-click on the Apple File Exchange *icon*, insert a PC disk into your Macintosh *SuperDrive* (which is only on newer Macs), select the file, and click the Translate button. Et voilà, you get a copy of the PC file that's readable by your Mac. The software comes free with your System disks. (Note that you must open the File Exchange utility *first,* then insert the PC disk.)

Apple key

The **Apple key** is the key near the Spacebar that has the Apple symbol on it ( ). This key doesn't do anything all by itself, but it's used in conjunction with other keys for *keyboard shortcuts* (also see *modifier keys*).

Apple II users call this the Apple key or Open Apple. On an older Macintosh keyboard, you might not see an Apple symbol on the key; the same key is called the *Command key* on the Mac, and has the cloverleaf symbol on it. Newer keyboards have both the apple and the cloverleaf symbols on the same key.

AppleLink

AppleLink is an *electronic mail* and information service that is not open to the general public, but is reserved for Apple employees, developers, universities, user groups, dealers, and a few other privileged individuals (not including me). Like any information or *bulletin board service,* you use AppleLink through your *modem.* AppleLink provides product announcements and updates (for *third-party* products as well as for Apple products) and various technical information, and, of course, *e-mail* to other privileged individuals.

Apple menu

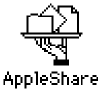
On the Macintosh, if you position the mouse pointer over the little apple you see in the far left of your menu bar, and then press the mouse button, you get a menu which is called (guess) the **Apple menu.** This menu holds your *desk accessories,* such as the Calculator, the Scrapbook, the Chooser, and the Alarm Clock.

In System 7 you can add any file you want into the Apple menu simply by putting its icon or, preferably, an *alias* of its icon into the Apple Menu Items folder, located in the System Folder.

The Apple menu you see to the left is my menu that I have customized. See how it's organized into groups—applications first (aliases of applications, that is), templates for my fax sheet and letters, control panels, and then desk accessories. I was able to organize them this way by taking advantage of the fact that blank spaces get alphabetized before characters, so I added one, two, or three blank spaces in front of the names. The dividers are aliases of blank documents that have been named "--------." In fact, everything in this entire menu is an alias except the desk accessories.

AppleShare

AppleShare is the name of *file server* software from Apple. If several Macintoshes are *networked,* or connected together so they can share data and programs, there is usually one or more computers, or at least hard disks, that are dedicated to the job of just serving files, or information, to you. Working in conjunction with the server software, the AppleShare software lets you choose the server from which you want to get data. AppleShare also lets you create private servers that need a password to *access,* or to get into.



This icon represents the Appleshare software.

AppleTalk

The **AppleTalk** network is how your Mac talks to your laser printer, to other Macs, or to other sorts of machines, provided they are hooked together, or *networked,* with *cables.* (You can use other types of networks with a Macintosh, but they require special *cards,* while AppleTalk is built in.) If several computers need to share one laser printer, you can get little adapters that let more than one computer plug into the printer through AppleTalk.

Apple II, Apple IIE, Apple IIGS, etc.

These **Apple** computers belong to a family of personal computers from (take a wild guess) Apple Computer, Inc. They run under Apple's *DOS* or ProDOS operating systems, which means they function more like IBM-compatible computers than like Macintoshes. If you own a Macintosh, don't call it an Apple or people will think you have one of these Apple machines. Also see *Apple Computer.*

applets

Applets are miniature programs. The name comes from the term "applications" which is one variety of a software program (see *application*). Applets is just a cute name for what is generally called a *utility* program, but it's used almost exclusively in the Windows environment. A utility program is designed to do one little function, generally in the service of housekeeping for your computer system, rather than actually producing something. The term "applet," though, implies that these little programs are actually doing something practical, beyond what a utility would do.

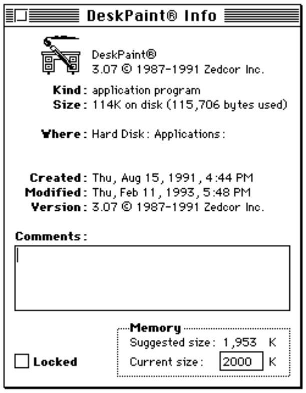
application

There are several varieties of computer *programs,* but the ones most of us are familiar with are the **applications.** An application is a program you use to get some practical work done, such as word processing or accounting or illustrating.

Programs such as WordPerfect, PageMaker, Excel, and Illustrator are all examples of applications. Programs such as *Adobe Type Manager* or *Suitcase* are *utilities* because we don't create work with them—they just help us manage our computer system.

application heap

The term **application heap** refers to the amount of *memory* set aside for an *application* or software program. On a Macintosh you can choose to increase the application heap, which you may want to do if you find the program running very slow, or perhaps telling you there is not enough memory to complete an operation or even to open the application.



DeskPaint® Info

DeskPaint®
3.07 © 1987-1991 Zedcor Inc.

Kind: application program
Size: 114K on disk (115,706 bytes used)

Where: Hard Disk: Applications:

Created: Thu, Aug 15, 1991, 4:44 PM
Modified: Thu, Feb 11, 1993, 5:48 PM
Version: 3.07 © 1987-1991 Zedcor Inc.

Comments:

Memory
Suggested size: 1,953 K
Current size: 2000 K

□ **Locked**

To increase the application heap, the application must be closed. Click once on the application icon, then from the File menu choose "Get Info." In the "Get Info" dialog box there are little boxes at the bottom, under "Memory." The box "Suggested size" is the minimum amount of memory needed to run the program efficiently. The box "Current size" is the amount of memory currently allocated to the application. You can change the amount shown in "Current size" by typing in it.

Notice in the example above that I increased the application heap in DeskPaint. It had previously been set to 1400 and I had trouble opening a graphic file. After giving DeskPaint a little extra memory, I had no problems.

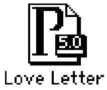
Now, you don't want to go arbitrarily jacking up the size of this application heap. Keep in mind that any extra memory you add to *this* heap takes it away from the System heap (see *heap*), and if you don't have a lot of memory you can get into trouble. Just increase the application heap to the minimum value you need to get your work done efficiently.

application icon

An **application icon** is the icon (picture) that represents a particular *application,* or software program. The term "application icon" is used even when the program is considered a *utility* rather than an *application.* Application icons are usually fancier than the corresponding *document icons* that represent documents created with the application. (Document icons often look like a piece of paper with the corner folded down.)

PageMaker 5.0

This is an application icon. It's fancy looking.



This is a document icon; it's obvious which application it was created with.

In the Microsoft Windows manual, application icons are called "program item icons," but real people call them program icons, application icons, or just icons.

.ARC

When you see these characters, **.ARC** (pronounced "ark" or "dot ark"), at the end of a PC file name, it means that the file has been compressed to save space, or *archived,* using a program called ARC (or a compatible archiving *utility*). Archived files are usually *compressed ,* so you will need to *unarchive* or *uncompress* the file before you can use it. For that, you will also need the ARC program. ARC files used to be the most popular type of archived file, but now ZIP and LZH files predominate.

architecture

Similar to the architecture of a building, the **architecture** of a computer refers to the design structure of the computer system and all its details: the *system,* the circuits, the *chips,* the *busses,* the *expansion slots,* the system *firmware, BIOS,* etc. The architecture largely determines how fast the computer is and what it can do. It also decides whether one computer is compatible with another. Can the same *boards* be used? Yes, if the architecture is compatible. Different models will have basically the same uses, but with varying degrees of performance. The architecture is what ensures *backward compatibility,* which means that your old software can run on a new computer. Also see *platform.*

If a computer has an **open architecture,** it means the company that builds the computer has published the specifications of the computer's design. Open architecture lets *third-party* developers create *add-in boards* (for customizing your computer, assuming the computer has *expansion* *slots*) and other *add-on* hardware that will work in the machine. So, open architecture means that any other company can create *memory* boards, video *adapters,* hard disks, and so on that function properly in the computer. An open architecture even allows other manufacturers to build computers that work identically. The Apple II and the IBM PC have open architectures, which is why so many *clones* (copies that work the same but usually cost less) of these machines exist.

If a computer has a **closed architecture,** the specifications for creating it are not public. That makes it difficult, if not impossible, for anyone else besides the machine's original manufacturer to create peripheral hardware to add-on to the machine. It also means no one else can recreate the same sort of computer. The Macintosh has a partially closed architecture—Apple released enough information for other manufacturers to build add-in boards and peripherals, but not enough for them to build Mac clones.

The terms "open" and "closed" architecture can also simply refer to whether or not a computer has *expansion slots* inside of it that allow you to plug in other add-in boards to expand or customize your machine. The original Macintosh had, in this sense, a closed architecture, whereas the later Macs have an open architecture.

archive

The term **archive** has the same general meanings in computer jargon as it does in English. As a verb, to archive means to store information, to back it up, generally with the idea that you are going to preserve it for a long time. As a noun, an archive is the information you store.

You archive a file to store it for safe keeping, or perhaps because you need it infrequently and so you want to keep it on a floppy disk in a filing cabinet instead of on your hard disk. Some people make a distinction between archiving files and *backing* them *up*. In the strict sense, an archive is intended as a permanent record that you may want to refer to later. For instance, architects and engineers who use computers keep archived copies of their plans in case a roof should cave in or some other catastrophe should befall them and an inspector needs to look at the plans.

A backup file, by contrast, is intended as a safeguard in case the original file gets lost or damaged. A given backup file may serve its purpose only temporarily. If you're working on a long report, you'll create a new backup file every day or so, containing the most recent revisions. Yesterday's backup file may become more or less worthless.

In the process of archiving files, most people also *compress* them so all the backup copies don't take up so much disk space. So, you'll often hear the term *archive* used to mean the entire process of compressing and storing. (To *compress* your files, you need special software; see *compress.*) Also in the process of archiving, many people pack a set of files into a single file (with or without compressing them) to store or send them as a unit.

But be aware: *compressing an archived file may not be such a good idea. Even though you are archiving files, you never want to compress your original and only file. All it takes is to lose one bit, one electronic signal, from a compressed file and that file is destroyed. Bits get lost all the time. In an uncompressed file, it's usually not a problem, but in a compressed file, it's a catastrophe of considerable dimension.*

To **unarchive** means, logically, to go get those files that you stored away (and to uncompress them if necessary). Or perhaps you received an archive through your modem. Before you can use it you must unarchive the file.

arrow keys

Most computer keyboards have a set of four **arrow keys** that can be used to move the *cursor* or the *insertion point* around the text on your screen. Each time you press an arrow key the cursor moves one character left or right, or one line up or down. To move the cursor in larger increments, or to select text as it moves, you can often use other keys in conjunction with the arrow keys, such as the Shift, Command, or Control keys.

The *numeric keypads* on many keyboards often work as arrow keys also. You may need to press the *Num Lock key* to switch from typing numbers to moving the cursor.

artificial intelligence

Artificial intelligence is a very general term applied to computer applications that attempt to imitate human reasoning. The term is usually used to describe the capabilities of *expert systems.* Expert system programs can be taught a set of rules for handling a well-defined yet complex problem, such as choosing the right antibiotic for meningitis. Artificially intelligent software can "learn" in a rudimentary fashion, adjusting the solutions it comes up with on the basis of past experience. But no expert system can begin to imitate the flexible intelligence of human beings.

Alan Turing proposed a strict test for artificial intelligence years ago. He said, "A machine has artificial intelligence when there is no discernible difference between the conversation generated by the machine and that of an intelligent person." His test goes like this: In Room A you sit a person at a computer. In Room B you sit another person at a computer, plus you have a separate computer that claims to be intelligent, controlling itself. Using her computer, the person in Room A converses with the person and with the "intelligent" computer in Room B, one at a time, without knowing which one she is "talking" to. The person in Room A could type "How's the weather" or "What is the speed of light in a vacuum?" or "What are you going to do after breakfast?" and the person or computer in Room B would send back the answer. If the person in Room A can't tell whether she's talking to a person using a computer or to the self-controlled computer, then the computer that is controlling itself is artificially intelligent.

As Arthur Naiman of the Macintosh Bible says of this test, "No computer in existence today can even dream of passing the Turing test—but then again, neither can most newscasters, corporate executives, or four-star generals."

ascender

The part of a lowercase letter of the alphabet that is taller than the letter **x** is called an **ascender.** For instance, the letters d, b, l, and h each have an ascender. The parts of the letters that hang below the *baseline* (the invisible line on which the letters sit) are called descenders; the letters p, q, and j have descenders. Also see *x-height* and *baseline.*

ascending order, ascending sort

First of all, to *sort* is computer jargon for alphabetize (or, if you're talking about numbers, to organize into numeric order). So an **ascending sort** means to alphabetize starting at the bottom and going up, like starting at A and going to Z (which is the way you were taught to do it in school). If your data consists of numbers, though, rather than text, an ascending sort would arrange them with the smallest number first, the next largest number second, and so on. Or if your data is a set of calendar dates, an ascending sort would put them in chronological order from earliest to latest. The opposite of ascending sort is *descending sort,* of course. Also see *sort,* which includes the difference between sort and *find.*

ASCII

ASCII (pronounced "askee") stands for **A**merican **s**tandard **c**ode for **i**nformation **i**nterchange. It's a standardized coding system used by almost all computers and printers for letters, numerals, punctuation marks, and invisible characters such as Return, Tab, Control, etc. The fact that almost everyone agrees on ASCII makes it relatively easy to exchange information between different programs, different *operating systems,* and even different computers. It also means you can easily print basic text and numbers on any printer, with the notable exception of *PostScript printers.* If you are working in the MacWrite word processing application on the Mac and you need to send your file to someone who uses WordStar on the PC, you can save the document as an ASCII file (which is the same as *text-only*). After you transfer the file to the PC (on a disk or via a *cable* or *modem*), the other person will be able to open the file in WordStar.

In ASCII, each character has a number which the computer or printer uses to represent that character. For instance, a capital *A* is number 65 in the code. Although there are 256 possible charcters in the code, ASCII standardizes only 128 characters, and the first 32 of these are "control characters," which are supposed to be used to control the computer and don't appear on the screen. That leaves only enough code numbers for all the capital and lowercase letters, the digits, and the most common punctuation marks.

Another ASCII limitation is that the code doesn't include any information about the way the text should look (its *format*). *ASCII* only tells you which characters the text contains. If you save a formatted document as *ASCII,* you will lose all the font formatting, such as the typeface changes, the italics, the bolds, and even the special characters like ©, ™, or ®. Usually carriage returns and tabs are saved. Also see *high ASCII.*

aspect ratio

Aspect ratio is a fancy term for "proportion," or the ratio of width to height. For instance, if a direction in a software manual tells you to "hold down the Shift key while you resize a graphic in order to maintain the aspect ratio," it simply means that if you *don't* hold down the Shift key you will stretch the image out of proportion.

Ooops—I did not maintain the aspect ratio when I resized this graphic.

Some combinations of computers and printers have trouble maintaining the correct **aspect ratio** when the image goes from the screen to the printer, or when the image is transferred from one system to another, so the aspect ratio can be an important specification to consider when choosing hardware.

assembler

An **assembler** is a software program that takes what the programmer wrote in *assembly language* and translates it into a program the computer can run. (Actually, the output of the assembler must be processed by a "linker" to produce the finished program.)

assembly language

An **assembly language** is a *programming language* that allows a programmer (a human) to tell the *microprocessor* (the *chip*) in the computer exactly what to do, in terms of the specific operations the processor knows how to perform. You can think of the difference between assembly language and a *high-level language* such as BASIC, C, or Pascal this way: A program in a high-level language is like saying "point at me," while the assembly language version is like telling a person to contract the muscles that elevate the shoulder, then contract the muscles that extend the elbow, and finally contract the muscles that extend the index finger. This analogy isn't perfect, but it should give you the flavor of the difference. In assembly, each programming line corresponds directly to an instruction in the processor's *machine language.*

Besides being laborious to write, assembly language programs have another drawback: They only run on one *microprocessor family,* sometimes only on a single microprocessor. In other words, an assembly program written for the Mac won't run on a PC, and vice versa. That's because each processor knows a different set of operations.

Of course, there must be a reason people write programs in assembly language. Actually, there are three: speed, program size, and control. Assuming equal skill of the programmers, an assembly language program is almost always faster than the equivalent high-level program, and in its finished, *executable* form, it's usually much smaller (even though the assembly programmer had to write many more lines of *code*). And because you can control the microprocessor on a step-by-step basis, your program gives you exactly the results you want.

asterisk *

The **asterisk** (*) is often called a *star* (I think it's called a star because so many people can't spell it or pronounce it; it is *not* "ateriks" or "asterik," although sometimes it is a "splat.") The asterisk has several functions on a computer. One is as the symbol for multiplication in on-screen calculators, in spreadsheets and databases, in programming, etc. You can use the asterisk on your keyboard's numeric keypad to multiply, and usually the asterisk above the number 8 on the keyboard also.

Another use for this character is as a *wildcard character* in some applications and *operating systems.* For instance, if you *search* for *.* (pronounced "*star dot star"*) on a *DOS* system, you can find a file with any name and extension. Or you can, in some applications, search for something like "rat*" and find rate, rationale, and ratfink, but not brat. Other applications may use a different character for a wildcard.

In *online* communications, an * means a kiss. (And this: -----/---@ is a long-stemmed rose.) See *baudy language* for a list of other important online visual communication shortcuts.

asynchronous

The word **asynchronous** is most often used when referring to transmission of data between two devices, as in "asynchronous communications." It usually applies to *serial* (as opposed to *parallel*) data transfers. Asynchronous means that the *clocks* or timers used by the two devices—such as your *modem* and the modem at the other end of the phone line—aren't running in synch with one another. For this reason, the data must include extra information to allow the two modems to agree on when each chunk (*byte*) of data starts and stops. This arrangement makes asynchronous communications less efficient than synchronous, but the latter requires sophisticated, expensive equipment. Synchronous communications are primarily used by large businesses and government agencies.

Asynchronous can also refer to events that occur at different times rather than concurrently. For instance, an exchange of e-mail involves asynchronous events: When someone sends me *e-mail,* I don't have to be *online* at that moment to receive it—I can pick up the message any time I like, which makes it an asynchronous event.

AT

Originally, **AT** was the name IBM gave to its second generation of PCs, introduced in 1984 (the full name was IBM PC-AT). The acronym stands for Advanced Technology, though a lot of people didn't think the machine lived up to its name, even when it first hit the market. As its main *microprocessor*, the AT used the *80286,* a faster, more capable chip than the one in the original IBM PC, but much less powerful than today's *80386* and *80486.* Nowadays, when someone says they have an AT, they usually just mean their PC has an 80286, whether or not the computer was made by IBM.

Despite its limitations, the AT did set standards that PCs have been following ever since. Almost all PCs sold today, even the ones with 80386 and 80486 processors, are "AT-*compatible*." This is important to you because it means these computers can run the same software and accept the same *expansion cards* (see *bus* and *ISA* for more on this last point).

AT command, AT command set

When you are using a *Hayes-compatible modem,* the collection of commands you can use to control your *modem* is called the **AT command set** (pronounced as the two letters, "a tee"). An **AT command** is one of the individual commands in the AT command set. They are called AT commands because each one must start with the letters AT, for **at**tention code.

You don't need to learn the AT command set. When you use *telecommunications software,* the program lets you control your modem easily—usually by picking from options listed in English on a menu. The software then translates your choices into the corresponding AT commands, sending them on to your modem.

A-to-D conversion

A-to-D conversion (**a**nalog-to-**d**igital conversion) is the process of converting *analog* information from life into *digital* information that a computer can understand. Most information in nature is in analog form, covering a continuous range of values (such as the passing of time, as exemplified by the hands of a regular clock). Digital information is in discrete, or separate, values, such as the countable chunks of seconds and minutes in a digital clock. You might want to use a computer to record analog information from nature by hooking it up to some kind of sensor like a light or pressure meter. But to store this information in the computer, the analog readings from these meters must be passed through an A-to-D converter to turn them into a digital form. Also see *analog* and *digital.*

attributes

Attributes means characteristics. For instance, in a database or a spreadsheet you can apply attributes to each *field* or *cell* to customize your document. As a general attribute, you can choose whether it is to be a text field or a numeric field or perhaps a computed field, whose value the application calculates for you. Then you can apply more specific attributes to the field or cell, such as making the text bold and right-aligned and perhaps in a particular typeface. If a field is numeric, you will have other attribute options available, such as how many decimal places to display, whether to use a dollar sign or a percent symbol, or whether to start a formula in the cell.

ATM

See *Adobe Type Manager.*

authorized Apple dealer

An **authorized Apple dealer** is, logically, a dealer who is authorized to sell Apple computer equipment. Apple only authorizes certain dealers, and they give these dealers authority to sell only specified machines. That's why you don't see Macintoshes in Home Club. You can sometimes buy Macs from unauthorized dealers, which is considered the "gray market."

Apple's *Performa* line of Macintoshes were developed to be sold in mass market stores such as Sears, Office Depot, and Circuit City, without upsetting the authorized Apple dealers.

AUTOEXEC.BAT, AUTOEXEC file

The name of this *batch file* comes from a combination of the words **auto**matically **exe**cuted **bat**ch file. Like other batch files, **AUTOEXEC.BAT** contains a series of *DOS* commands that your (*IBM-compatible) PC* runs for you, one after the other, so that you don't have to type the commands individually. What's special about AUTOEXEC.BAT is just that DOS automatically runs this particular batch file each time you turn on or restart the computer. Most people wind up calling AUTOEXEC.BAT the "AUTOEXEC" file for short.

You can customize the AUTOEXEC file yourself, filling it with exactly the commands you want to get your system up and running to your specifications and to suit the needs of your software and *peripherals*. AUTOEXEC.BAT is typically used to set the look of the DOS prompt, tell DOS which *directories* it should search when looking for programs to run, configure the *serial ports,* load the mouse *driver*, and start *memory resident* programs and utilities. If you like, you can use the AUTOEXEC file to start a particular application program (such as your word processor) or to start Windows.

Here's an excerpt from a typical AUTOEXEC.BAT file:

```
PROMPT $p$g

PATH=C:\WINDOWS;C:\DOS;C:\UTILS

SET TEMP=D:\TEMP
```

These lines set the DOS prompt to show the current directory; set the DOS *path;* and tell the system to look for temporary files in the D:\TEMP directory.

auto-repeat key

An **auto-repeat key** is a key that will automatically repeat, after a brief pause, while you keep it pressed.

Every key on a Macintosh keyboard auto-repeats unless you choose to turn it off through the Keyboard control panel. You can also choose how fast you want the keys to repeat, and how heavy or light your touch should be on the keyboard to start the repetition.

On a PC keyboard, all the keys auto-repeat, and you can control the rate they repeat—see *typematic rate.*

autosync

Autosync refers to a type of computer monitor and is synonymous with *multiscan.* Please see *multiscan* for details.

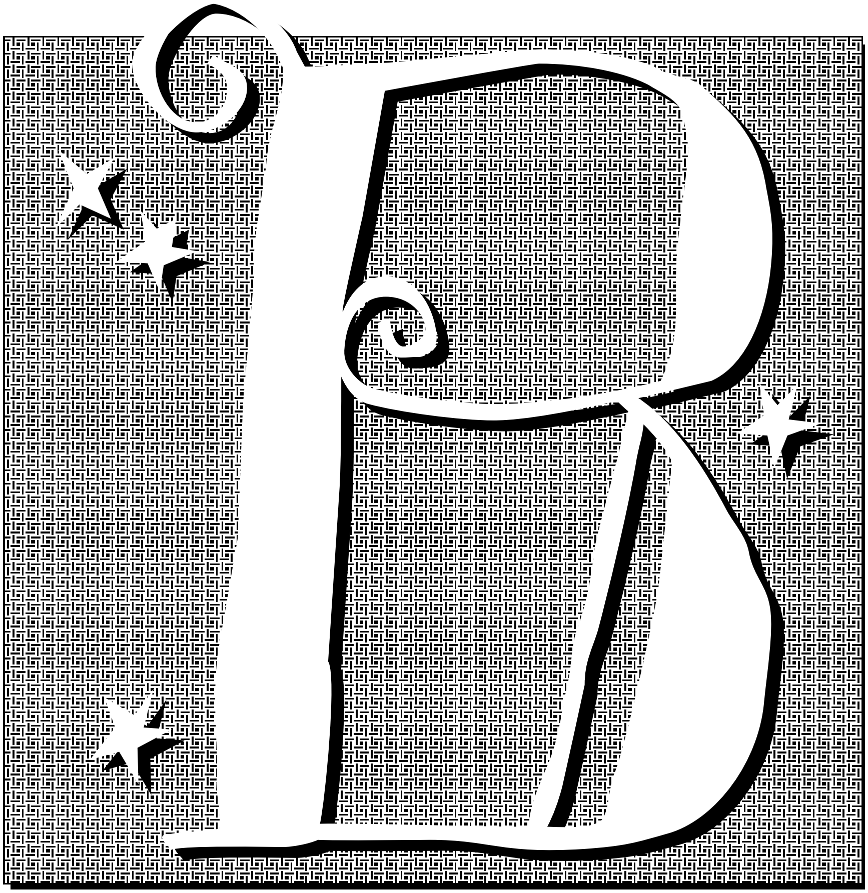
auto-save

Some programs have an **auto-save** feature that will automatically *save* your document to disk every so many minutes, just in case you forget. You can control how often you want the auto-save to interrupt your work. This is nice in theory, and it is nice for people who forget to save. Personally, I hate to be interrupted in the middle of my work as the auto-save feature stops what I am doing and does its own thing—I prefer to have control over it myself.

A/UX

A/UX (pronounced “ox”) is a version of the *UNIX operating system* that can run on any of the computers in the Mac II series. If you need to run a UNIX application, as many people in government and industry do, A/UX allows you to run an A/UX-*compatible* version of the application on the Mac. Regular Macintosh applications run on A/UX as well. A/UX has enabled many government departments to requisition Macs, since the availability of UNIX for a given system is a requirement for many government branches.

The A/UX system does require generous amounts of *memory* and disk space, as well as plenty of processing power, and is not as easy to use as the standard Macintosh operating system.



background, background process

The plain ol' everday word **background** takes on a multitude of meanings in the jargon jungle. If something happens "in the background," that means it happens while you are doing something else. When you print, for instance, you might be accustomed to having to wait until the printer is finished before you can have your screen back to work on. But some systems allow you to print in the background, meaning while the printer is processing and printing your document, you can continue working on the screen (also see *spool* and *spooler.*) Some systems, including Microsoft Windows and the Macintosh, let you copy files or format disks in the background, and you can buy software that lets you *telecommunicate* or receive faxes in the background too. A **background process** is anything that's happening in the background.

In *HyperCard,* a document is called a *stack* because it is made up of a set, or stack, of "cards" on the screen. Each card has two layers—the **background** layer and the "card" layer (they should have named the "card" layer something different from the card itself). The background layer is comparable to the blue lines on a 3 x 5 index card in that every card shares that same background. The card layer is comparable to the information you write on a 3 x 5 card in that every card can have something different on it.

background color

On a PC running in "text mode," each character has jurisdiction over a small rectangular block on the screen. The background is any part of this block that isn't used by the character itself (the foreground). There are eight **background colors** available, and each character's background color can be controlled independently. Many programs let you pick the foreground and background colors you want for different kinds of information on the screen, such as menus, help boxes, and regular text. When you're using *DOS* itself, you can set text colors only if you have *ANSI.SYS* loaded.

Backgrounder

The **Backgrounder** icon you may see in your Macintosh's System Folder in *System 6* is a system file that tells the Mac what to do when you want to use background printing. With background printing, you can send the document off to your personal printer and the Mac will let you have your screen back so you can continue to work while the printer prints. Taking advantage of background printing requires many things: you must have the Backgrounder icon in your System Folder, must have MultiFinder turned on, you must have the "Background printing" button checked in the Chooser, and you must have enough RAM available to run the printing and your software program at the same time.

backslash

The **backslash** is the slash character with the top angling toward the left, like this \. On PC keyboards, the backslash is on the same key as the split vertical bar or "pipe" character. You might find this key near either Shift key or above the Enter key. The only common use for the backslash is to separate subdirectories in DOS commands. For example, typing the command CD\UTILITY\PRINT switches you to the PRINT subdirectory of the UTILITY directory. A backslash all by itself refers to the *root directory* of the current disk drive. If you type **CD ** you move to the root directory.

Backspace key

When you're typing, pressing the **Backspace key** will move the typing cursor to the left one character at a time, erasing the character as it goes. The Backspace key is labeled "Delete" on the keyboards of some computers, because it does delete text as it backs over it.

On PCs, there's a separate key labeled *Del* or Delete that deletes the character that the cursor is resting on (in text mode), or the one to the **right** (yes, the right) of the cursor (in graphics-based programs, including Microsoft Windows).

back up, backup

The term backup can be a verb ("It's important to **back up** your files") or an adjective ("Did you create a **backup** file?") or a noun ("I lost my **backup**!"). Any which way you use it, the idea is that you have a copy of every important file, just in case something happens to the original. It's frighteningly easy to lose data when a disk goes bad because of heat, magnets, coffee, dogs, kids, or unknown forces. You can lose a disk, and your computer might even get stolen. So make the backups, and put them somewhere out of harm's way.

For very important data you really should have two or three backups and keep them in separate locations. This is not a rare scenario: Something happens to your original; it gets corrupted or eaten or who knows what. You're not worried because you have a backup of that original. So you stick the disk with the backup in the computer and before you have figured out what went wrong with the first file, this backup file also gets eaten ("eaten" as in "destroyed"). By the time you've figured out there is really a problem here, all traces of your vital information have vanished—permanently.

BMUG once lost the original and both backups of their 400-page newsletter just days before it was supposed to go to press. If it can happen to BMUG, it can happen to anybody.

Many people *compress* files before backing them up because a compressed file takes up less space on a disk, so you can fit more backups on one disk. But you should always keep at least one uncompressed copy of your truly irreplaceable files. If something goes wrong with a single bit of data in a compressed file, the whole file may be unable to be uncompressed and therefore worthless. Use compressed files *only* as second-line backups, or for information that you can readily replace.

backward compatible

Computer products—from software applications to peripherals to the computers themselves—are constantly being updated. Programs start out at *version* 1.0, then as they progress the version numbers increase. Typically once you have updated your application, any document you create in the updated version cannot be opened by an earlier version of the same application. For instance, the application PageMaker version 4.2 cannot open any publication created in PageMaker version 5.0. Some products, though, are **backward compatible** (also known as "downward compatible"), which means that a document created in an updated version of a software application can still be opened by an earlier version of the same application. For instance, documents created in Microsoft Word 5.1 (which is backward compatible) can be opened by Word 4 (which is now considered upward compatible). Of course, as you go backwards your document will lose the newer features that are part of the updated version.

Backward compatibility also applies to computers: if a newer model can run the same software as an older model, the newer model is considered to be backward compatible. Almost all Macintoshes are backward compatible with all older versions of the Mac. Machines like the IBM PC/AT are backward compatible with the IBM PC/XT.

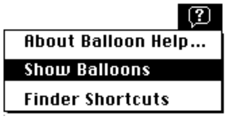
bad sector

A disk has two sides (a top and a bottom). Each side of the disk has tracks, or concentric rings on the surface. Each ring is divided like a pie into equal wedges, or sectors, which are the smallest units of storage space on the disk. If one of these units is damaged or flawed, it is considered a **bad sector** and cannot be used. If there was already data in that sector when it got damaged, chances are slim that you can recover that data unless you have the specialized hardware and software necessary for that sort of operation. Almost all hard disks are born with bad sectors, so don't freak out if your software utility reports them. Other bad sectors should not start appearing, though, after you start using the disk.

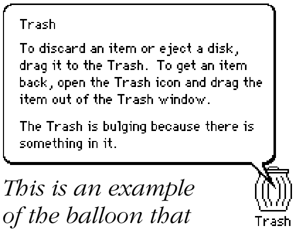
Balloon Help

Balloon Help is a feature of System 7 on the Macintosh where little "balloons" (like cartoon-character talk) automatically pop up when you point to items on the screen. The balloons contain helpful information about the item you are pointing at. You can view the available balloons at the *Finder* (also known as the *Desktop*) and in some applications by choosing "Show Balloons" from the Help menu (the one with the question mark). Many software packages are including balloon help right in the application so you can just point (you don't have to click) to a menu item or a dialog box to see a balloon that explains how that item works.

The idea sounds good, but in reality the balloons tend to have pretty wimpy information in them. Plus they can drive you nuts the way they pop up in front of your face constantly. Some software companies have elected to ignore them altogether in their new software, while other companies say they'll be putting information that's actually useful in their balloons.



Choose "Show Balloons" from the Help menu, which is the one with the question mark in the balloon. If you don't have the question mark, then you are not on System 7 and won't have balloons.


Trash

To discard an item or eject a disk, drag it to the Trash. To get an item back, open the Trash icon and drag the item out of the Trash window.

The Trash is bulging because there is something in it.

This is an example of the balloon that appears when you point to a full trash can.

banding

Some *draw* and illustration programs let you create a fountain fill or "split fountain" effect in which the color or shade of a part of your picture changes gradually from light to dark or from one color to another. But due to the relatively low resolution of most lower-end printers, a fountain often shows the gradation from dark to light in distinct, contrasting bands rather than in one continuous flow. This effect is called **banding.** The higher the resolution of the printer, the less banding will be visible.

Depending on the number of colors or grays you can see on your screen (see resolution, monitor*), you may see banding on the screen. That does not mean the banding will print. The rectangle on top is how it looked on a screen with only 16 colors; the rectangle on the bottom is how it printed on an* imagesetter.

bandwidth

Bandwidth measures the amount of information that can flow between two points in a certain period of time. The “broader” the bandwidth, the faster the information flow. You can use the term to describe how quickly information moves from the hard disk into *memory,* or from the computer to an *add-in board* on the expansion *bus*, or from one *modem* to another across a telephone line. Depending on whether the transmission is *digital* or *analog,* the rate is measured in *bits per second (bps)* or in *hertz (cycles per second).*

The bandwidth of a particular part of your system can have a big impact on the speed of the system as a whole. Consider the situation when the computer needs to display a graphical image on the screen (*all* Macintosh and Windows screens are graphical). The computer’s main processor is responsible for deciding which dots on the screen should be white, which red, and so on. But it must transfer that information through a *bus* to the video circuits that actually control your screen. No matter how fast the processor “composes” the screen, your monitor will seem sluggish if the bandwidth of the bus is narrow.

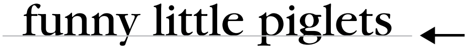
The newer expansion buses for the IBM and compatible PCs, the *Micro Channel* and the *EISA*, were developed partly to improve the bus bandwidth and so take better advantage of increased processor speed. And some new PCs have video circuits built right into the main circuit board of the computer, linked directly to the main processor via a high-speed *local* or direct *bus.*

bank

A **bank** is a collection of identical electronic devices that are connected so they can be used as a unit. For instance, a bank of memory consists of a number of separate memory chips that can be accessed as if they were one continuous section of memory.

baseline

Referring to type, the **baseline** is the invisible line that the letters sit on. Some letters, such as p, y, or g, have *descenders* that drop below the baseline. Also see *descender, ascender,* and *x-height.*



This is the baseline, upon which all the letters sit. Notice the descenders of y, p, and g.

BASIC, BASICA

BASIC stands for **b**eginner's **a**ll-purpose **s**ymbolic **i**nstruction **c**ode, and is a computer *programming language* developed in the mid-sixties. BASIC has the advantage of English-like commands that are easier to understand and remember than those of most other languages. Even so, the latest versions of BASIC can do just about anything programming languages like *C* or *Pascal* can do.

BASIC programs have a reputation for being very slow, which they certainly were in the early days of personal computing. This sluggishness was mostly due to the fact that in those days, BASIC was an "interpreted" language; that is, every time you ran a BASIC program, you were really running an "interpreter" which executed your program code line by line, converting it *on the fly* into a form your computer could understand. That conversion process takes time. Now many good BASIC *compilers* are available. A compiler does the conversion ahead of time and only once, turning the program code into an executable program your system can run directly, at top speed. So modern, compiled BASIC is easier to use and just about as fast as C.

Interpreters do have some advantages, though. The process of writing and testing an interpreted program is actually quicker and more convenient than with a compiled program (for the explanation, see the entry for *interpreter*). A BASIC interpreter makes especially good sense for creating short, simple programs, which is all that most personal computer users would be willing to tackle. If you're interested, you can find lots of old computer books full of BASIC programs at public libraries. At any rate, some computers come with a BASIC interpreter. The best example is Microsoft's BASIC, or GW BASIC, the interpreter that comes with MS-DOS. True IBM-brand PCs had slightly different versions of the interpreter called BASIC and BASICA (Advanced BASIC) that only worked on those computers.

BAT or .BAT file

PC

When you see the file extension **.BAT** (pronounced "dot bat") at the end of an MS-DOS file name it means the file is a batch file. Please see *batch file*.

batch file

PC

A **batch file** is a file containing a series of commands that the operating system will carry out for you, one at a time. All you have to do is start the batch file—from that point on, the process is automatic. Batch files are great when you use a given set of commands repeatedly—instead of activating each command separately every time you want to carry out that set of commands, you can accomplish the same thing in one step.

Many computer operating systems have the built-in ability to execute batch files (or their equivalent). In *DOS*, the operating system for IBM-PCs and compatibles, you can recognize a batch file by its name, which always has the .BAT extension (as in GO.BAT, SEND.BAT, or AUTOEXEC.BAT). *DOS* includes a group of special commands that are only used in batch files.

Here's an example of a simple DOS batch file used by a writer who often sends one of his poems to someone else on floppy disk. This batch file *formats* the disk, then goes on to automatically copy the poem file there:

```
FORMAT A:
COPY C:\NEWPOEMS\%1.POM A:
```

If this batch file was named SEND.BAT, our writer would just type

```
SEND DAISY
```

to do the formatting and copy the file *DAISY.POM* to the disk. (DOS knows to substitute "whatever he types after SEND" for the "%1" in the batch file.)

Computers and operating systems with graphical user interfaces, such as Windows and the Macintosh, usually don't have batch file capability because things are so much easier to do anyway. You can buy software products that let you create batch files for these systems (Frontier for the Mac), or you can use *macros.*

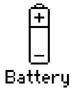
batch processing

Batch processing refers to an operation in which the computer automatically performs a given command on a number of files, or performs a series of predetermined commands without further intervention on your part.

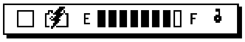
Running a *batch file* is one example of batch processing, but there are plenty of others. When you select several documents from the same application and print them all in one step (if the application allows you to do that), you are "batch printing," which is a form of batch processing. Or let's say that you want to send a whole group of files to someone else via your modem—if your *communications software* permits batch processing, you can choose all the files you want to send, and have the software send them off in a batch while you go to the kitchen for a snack. Batch processing is a good feature to have in most applications.

Battery

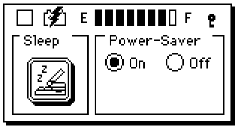
The **Battery** icon in your Mac's System Folder (in System 7 it is inside the Apple Menu Items folder) and the Battery item on your Apple menu are only for use with the battery-powered Macintoshes such as the PowerBooks. The Battery is a little *utility* that lets you know how much charge is left in the computer's battery. If you're using any type of Mac other than a Portable or Powerbook, you can throw away the utility (move the icon from your System Folder into the trash).



This is the icon for the Battery.



When you choose Battery from the Apple menu, you can see how much power you have left. Option-click on the picture of the battery to put the computer to sleep.



Power-Saver ● On ○ Off

When you click the flag in the upper right corner, you'll get this tiny control panel.

(The lightening bolt through the battery icon indicates the battery is being charged because the PowerBook is plugged in.)

baud, baud rate

When you use a modem to send information from one computer to another over the phone lines, the information moves at a certain speed. This speed is measured in *bits per second,* a *bit* being one electronic unit of information. This bits-per-second rate is also commonly called the **baud rate** (pronounced "bod"). It's a measure of how fast your modem can send and receive information. Modems most typically send at 1200, 2400, or 9600 baud.

Now, I do have to warn you that even though the masses (that's us) often use the term "baud" to mean the same thing as "bits per second," this is technically incorrect, and someday someone will surely scold you. One baud is actually one "modulation change" or one "signal event" per second on a communications channel. It's only at 300 baud that one signal event equals one bit per second. A modem operating at 1200 bits per second needs only one signal event to send 2 bits, so if you want to be persnickety you should call it a 600 baud modem. See *bps* for a discussion of common communications speeds and how they translate into speed ratings you can relate to.

The term "baud" is named after the inventor of the Baudot telegraph code, J.M.E. Baudot.

baudy language

The collection of abbreviations and symbols used in electronic mail messages is sometimes called **baudy language** (see *baud*). It is an integral part of online communication to use baudy language, since many messages can be misinterpreted unless accompanied by a smile, a frown, a wink, a giggle.

The following list of symbols are to be read with your left ear on your left shoulder. As you become accustomed to using and reading them, you'll find that you will be able to understand them without bending your head. All of the symbols can have a nose added to them with a hyphen, as in :-), which is how you will commonly see them on CompuServe. People on America Online typically leave out the nose.

:)	smile
;)	wink
: D	big smile, laughing
: >	sarcastic or devilish smile
: (	frown
:' (	crying
: l	neutral; bored
}:)	horny smile
:-{)	smile with a mustache, usually indicating a male
: {>	devilish smile with a mustache
: O	O Dear, O Woe is Me
: P	tongue sticking out; raspberry; bronx cheer (really rather rude and not to be used lightly)
: X	my lips are sealed
0:)	angel
}: >	devil
::: :::	colons surround an action that the writer is pretending to do, such as **:::going to bar and getting InfraDig a daiquiri:::** *or* **:::whacking FooFoo upside the head:::** *or* **:::getting massage oil now:::**
{{ }}	indicates the person whose name is between the curly brackets is being hugged
<---	indicates the person writing this message is the subject of the sentence, as in **ToadHall: <—- female, in answer to your question**
*	kiss
P*	french kiss
^ ^ ^	giggles (this solves the problem of how to spell "tee hee"); type the caret (^) with Shift 6.
-------\----@	long-stemmed rose

The following list of shortcuts are abbreviations for commonly used phrases that give the other readers important information.

!	I have a comment
?	I have a question
afk	away from keyboard
ALink	AppleLink
AOL	America Online
asap	as soon as possible
awc	after while, crocodile
bak	back at keyboard
brb	be right back
btw	by the way
byob	bring your own bottle
byom	bring your own Mac
CIS	CompuServe
CI$	CompuSpend
cul	see you later
cula	see you later, alligator
esad	eat sh*t and die
fubar	f**ked up beyond all repair
fyi	for your information
ga	go ahead
gmta	great minds think alike
ima	I might add
imho	in my humble opinion; in my honest opinion
imnsho	in my not-so-humble opinion
imo	in my opinion
lol	laughing out loud
obtw	oh, by the way
oic	oh, I see
otoh	on the other hand
rotflol	rolling on the floor, laughing out loud
rsn	real soon now
snafu	situation normal: all f**ked up
sol	sh*t out of luck
sos	help!
tanj	there ain't no justice
tanstaafl	there ain't no such thing as a free lunch
tptb	the powers that be
ttfn	ta-ta for now
ttyl	talk to you later
wb	welcome back
wtg!	way to go!
xoxoxo	kisses and hugs

BBS

BBS stands for **b**ulletin **b**oard **s**ervice, which is a service usually set up by an organization or a club to provide or exchange information. You *access* the BBS through your *modem*. After your modem has dialed the phone number for the BBS and established a connection to the BBS's modem, you'll see on your screen the computer equivalent of a bulletin board. You can read messages, post messages of your own, ask questions, answer questions, make new friends. Also see *online service.*

BCS

The **BCS** is the **Boston Computer Sociey.** This information came directly from Pam in Boston:

The BCS is made up of over 28,000 members of *user groups* and special interest groups (*SIG*, which is a specialty subset of a user group). The user groups include: Amiga, Apple II, Atari, Commodore, ZITEL (CP/M and DOS), DEC, Hewlett-Packard, IBM PC & Compatibles, Laptops, NeXT, Macintosh, Tandy, and Texas Instruments.

The *SIG*s include: Amateur Radio, Artificial Intelligence, Beginners, Business & Management, Church & Synagogue, CASE, Construction, Consultants & Entrepreneurs, Computer-Aided Publishing, Databases, dBase, Education/Logo, E-Mail, Graphics, Hypermedia/Optical Disk, International, Investments, Legal, Lotus, Medical/Dental, Music, Networking, Programming, Real Estate, Robotics, Science & Engineering, Social Impact, Special Needs, Telecommunications, Training & Documentation, and User Interface Design."

As you can see, the BCS is hard to define because each of these user groups and SIGs is independent, having their own Directors and Activists who run each group, meetings, workshops/seminars, etc. It's pulled together by the BCS Office and a staff which handles the overall management of the groups with budgeting and finance, newsletter printing and distribution, promotion, membership, a central telephone contact, and the Resource Center, a room with about 30 different computers where members can test or try out hardware and/or hundreds of software packages.

There are 25 *bulletin board services* (BBS) set up and maintained by these user groups and SIGs, with another 12 BBSs participating in a message exchange through BCS Echomail Conference.

The BCS offers the world's largest network of information and support for personal computer users. At the BCS, there's no such thing as a too-simple or too-complex question. They offer support services for people at all levels of experience. You learn at your own pace, according to your own needs.

BCS-Mac has its own separate office and telephone number: 617.625.7080.

The **IBM-PC Group** also has its own office and telephone number, which is 617.964.2547.

The Boston Computer Society
One Kendall Square
Cambridge, MA 02139
Tel: 617.252.0600
Member relations and assistance: ext. 3314
Join or renew by phone: ext. 3316
Regular 1-year membership: $39/year
(other rates available, such as Senior Citizen, Student, etc.)

beep

Beep is the generic term for whatever sound a computer makes when it's trying to tell you something. If you press the wrong key or click on something you shouldn't, the computer will beep at you. Sometimes it beeps just to let you know that it has started or finished doing something (like copying a file). You can customize the beep sound in some computers. In particular, newer Macs and PCs with Windows 3.1 come with several sounds you can assign to various types of "events" in the system. Hundreds can be added, and you can even make your own sounds.

bells and whistles

When a product is said to have **bells and whistles,** it means the product has some fancy features that go beyond the basics. These features may allow you to work faster or more conveniently, they may entertain you, or they may be there just to look impressive.

benchmark

A **benchmark** is a test used to measure the relative performance of hardware or software products. If you want to know which of two computers is faster, you can't take the manufacturers' word. Instead, you run the same benchmark software on both computers and compare the results. Magazines generally report benchmark results in comparative reviews.

Bernoulli box

A **Bernoulli box** is a removable cartridge disk system from Iomega Corporation. Technically, it is distinctly different from other cartridge disk systems in that Bernoullis use flexible disks (sort of like floppy disks, but these can hold something like 90 megabytes of data apiece) rather than hard disks. Something about the type of disk and the fluid dynamics in the disk mechanism make Bernoullis much more stable and better able to withstand bumps and jolts, so they are considered to be the most reliable of all removable cartridge, magnetic disks. They are also the most expensive, but you have to balance the price against the cost of lost data.

beta, beta tester, beta version

Computer products go through extensive testing before they can be released to the unsuspecting and trusting public. When a product has passed the in-house testing stage (see *alpha testing*), it goes into **beta testing,** often just called **beta. Beta versions** of the product are sent out for beta testing to “normal” people who don’t work for the company. (These people are then, logically, called **beta testers** or **beta sites**.) The beta testers work with the software or hardware in real-life situations and report back the things that go wrong or that need improvement.

Usually, the software company sends out several beta versions during the development process. The first one often has lots of *bugs,* and some of the features may not work at all. But at least it gets the ball rolling, so the developers can get feedback on the product design from people in the outside world. Based on this feedback, the developers may decide to revise the way the program looks and works. The next beta version should have fewer bugs, and look more like the final version. The beta testers are supposed to try every feature to find anything that doesn’t work right. Based on their reports, the programmers find and fix the problems.

Ideally, this process repeats itself a time or two more, until someone decides all the bugs have been found and everyone’s pretty happy about the way the program works. The sad truth: A few bugs always remain, even after beta testing. Sadder yet: Products with known bugs are sometimes released commercially, just to get the product out on the market on schedule. See *bug*.

betaware

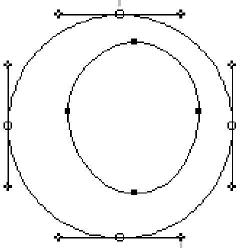
Betaware describes a product that is in the process of going through *beta testing* before being released to the public. Also see *vaporware.*

Bézier curves

When you draw a curved line in a sophisticated *draw* or illustration application (such as Adobe Illustrator, Aldus Freehand, or Corel Draw), the application represents your line in the computer as a particular kind of mathematical formula. Curved lines defined with these formulas are called called (pronounced "bayz yay"). Bézier curves are kind of tricky to work with, but they are an efficient way to create clean, smooth graphic images. Scalable fonts, like Type 1 and TrueType fonts, are also based on Bézier curves.

You don't need to know anything about the formulas involved, but they do affect the way you work with curves on the screen. They define a Bézier curve precisely by specifying a few points: the points where the line starts and stops, the points where the curve changes direction (known variously as the anchor points, handles, or nodes) and the control points for each of the anchor points (sometimes called BCPs, Bézier control points).

These are the anchor points.



These are the control points, or BCPs.

On the screen, the anchor points (where the curve changes direction) appear as big squares or rectangles directly on the path of the curve. At each anchor point you must also know the angle of deflection, and how far the curve extends in that direction. These characteristics are represented on screen by the length and direction of dashed lines which extend out on either side of the anchor points and end in control points.

Once the anchor points and control points are defined, the program just connects the anchors with a smooth curve. Some programs let you draw freehand; after you've drawn a line, the program converts it to a Bézier, calculating where the anchor points and control points should go. In other programs, you place the anchor points by hand, one at a time. This method may sound less intuitive than the freehand approach, but it can actually be quicker and more precise. To modify the shape of the line, you can move the anchor point itself, or manipulate the control points to lengthen or shorten the deflection lines or change their angles.

All this sounds quite complicated, and it does take awhile to get the hang of these bizarre Béziers. But if you keep at it, you'll soon find that working with Bézier curves becomes a fairly intuitive process, and it definitely lets you place and structure curves exactly where you want them.

Big Blue

Big Blue is the nickname for IBM, the company. How did it get the name? Four different sources gave me four different reasons why it's called Big Blue. One said it's because the logo is blue. One said it's because all the machines were blue. Another said it's because all the typewriter and computer dust covers were blue. Another said it's because everyone was required to wear blue suits. Apparently there is some fixation on blue.

big red switch

On most PCs, the on/off switch (the power switch) is bright red and fairly large. But usually it's called the **big red switch** only in the context of something going really wrong with your system—so wrong that you can only fix it by shutting off your computer altogether, then re*booting* it. For instance, "Just as I pressed the save key, the program crashed again. I found myself reaching for the big red switch for the hundredth time that day."

bilevel bitmap

The image you see on your computer screen is made up of thousands of tiny dots, or pixels. On a monochrome monitor (black-and-white screen) each pixel on the screen can either turn on or off. Each pixel is controlled by a *bit* (a bit is the smallest unit of electronic information) that tells it whether to be black or white. Thus the **bilevel,** or two levels, of **bitmap** are black or white.

A "deep" bitmap, on the other hand, can be many more shades, or levels, of gray because it can deal with more than just one bit of data. See the entries for *bitmap* and *bit* for more information, plus the entry in the Symbols section on *2-bit, 4-bit, 8-bit (etc.) color.*

binary, binary system

The binary system is a method for working with numbers based on only two digits: 1 and 0 (binary is also known as "base two"). It's kind of a fun system to set your brain working on, if you like puzzles. In a binary number such as 10101, each digit represents a power of two; 10101 is 21 in the "base ten" system we're used to (base ten is our culture's counting system, but it isn't universal among humans). (I've always wondered how it works on powers of 2 if the only numbers are 1 and 0. But Steve reminded me that in the decimal system, which is based on powers of ten, there's no digit for "ten" either.)

On the computer, these binary numbers can easily be represented by tiny electronic signals, each of which is either "on" or "off" (1 or 0). Each one of these single signals, 1 or 0, is called a *bit.* The computer uses groups of these signals, or bits, to do all its work and to store all your information. For instance, when you press a key, a sequence of bits (such as 01000001) is sent to the computer, which in turn displays on your screen the character that corresponds to that number—in this case, the letter A. Also see the definitions for *bit, bitmapped, bilevel bitmap,* and *deep bitmap.*

For those who like number puzzles, this is from Steve: To make the conversion between binary and decimal, start at the right and multiply each digit by the next higher power of two, then add up all your answers. So, with 10101, the first digit equals 1 (1 times 1: 2 to the 0th power is 1); the next digit is 0 (2^1 equals 2, but 2 times 0 makes 0); the third digit equals 4 (1×2^2); the fourth digit is 0 again; and the 1 on the left equals 16 (1×2^4). And 16 + 0 + 4 + 0 + 1 = 21. What fun!

BIOS, bios

Every computer is equipped with built-in instructions that tell it how to perform the really basic functions: start itself up, test itself, and communicate in a rudimentary way with the disk drives, keyboard, and screen. On an IBM-type personal computer, these built-in instructions are called the **bios** (pronounced "by ose," that's "ose" as in comatose). BIOS stands for **b**asic **i**nput-**o**utput **s**ystem. The BIOS is *firmware*, meaning it is a program built into the *read-only memory (ROM)* in your computer, rather than stored on a disk (because the BIOS is stored in ROM, it's sometimes called the ROM BIOS).

When you turn on your computer, the BIOS is responsible for checking all the hardware, including memory; it will display an error message if it finds a problem. The BIOS then loads the operating system—whether it's DOS, OS/2, Unix, or what have you—into memory from disk. Even after the operating system is running, the BIOS handles many essential chores, putting characters on the screen, getting characters from the keyboard, reading and writing sectors to the floppy or hard disk. You'll see this as your ROM BIOS chip on your computer that works with your software.

bis

The term **bis** (pronounced "biss") added to the name of a product, file, etc., indicates that the item in question (file or software) is the second version. For example, V.42*bis* is the designation of a particular standard for modem operations; two V.42*bis* modems can automatically compress data transferred between them, while plain V.42 modems can't do the compression but otherwise work the same. "Bis," which comes from Italian through Latin and Old Latin, originally meant "twice."

bisync, bisynchronous

Bisynch (pronounced "by sink") and **bisynchronous** are short for **bi**nary **synchronous,** which is a communications *protocol,* or set of rules for sending information between different machines. Bisynchronous communications are extensively used in *networks* on *mainframes.* Bisynch communications require that the *clocks* on both the computer sending the information and the computer receiving the information are synchronized before the information starts to transmit. Compare with *asynchronous* transmission.

bit

Bit is short for **binary digit**. One bit is the smallest unit of information that the computer can work with, and all the information it works with is made up of bits.

When you hit a key or click a mouse button, you send tiny electronic on/off signals to the computer. Each tiny electronic signal is **one bit.** The computer usually groups these tiny signals, or bits, into bigger chunks to work with: a series of eight bits strung together is a *byte;* a byte typically creates one character (letter, number, etc.) on your screen. A series of 1024 bytes strung together is a *kilobyte.* You've probably noticed that most computer storage devices (such as disks) and software files (your documents, for instance) are measured in kilobytes or *megabytes* (a megabyte is 1024 kilobytes). Well, those measurements are referring to how many electronic on/off signals it took to create and store the information. The more information there is, the greater the number of kilobytes or megabytes.

The fact that the computer has to work with so many bits is more than offset by another fact: it can handle bits far faster and with much cheaper circuits than it would take to work with decimal numbers directly. Also see *bitmap, binary system,* and *byte.*

By the way, you may have noticed that some of the switches that turn on your computer or other components, such as your hard drive or scanner, have a little 1 (one) and a 0 (zero) on them. Yes, they are a 1 and a 0, not an I (letter i) and an O (letter o). The 1 means on and the 0 means off, just like bits. (I/O, the letters i and o, stands for Input/Output, which is something completely different; see *I/O.*)

More fun from Steve: You can represent any number as a series of bits—you just have to use the "base 2" or *binary* counting system. In our accustomed base 10, or decimal, system, 7 equals 111. Reading from the right, each place signifies a power of 2 rather than a power of 10; in this case, the digit on the right means 1, the middle digit means 2, the one on the left, 4, for a total of 7. The expression 101 in binary equals 5 in decimal (1 plus 0 plus 4). In binary, three bits (digits) are enough to make any number between 0 and 7. Four bits cover everything from 0 to 16, eight bits take you up to 256, and 32 bits allow for numbers as large as 4,294,967,296. Whew!

bitmap, bitmapped

A **bitmap** is an image or shape of any kind—a picture, a text character, a photo—that's composed of a collection of tiny individual dots. A wild landscape on your screen is a **bitmapped** graphic, or simply a bitmap. Remember that whatever you see on the screen is composed of tiny dots called *pixels*. When you make a big swipe across the screen in a *paint program* with your computerized "brush," all that really happens is that you turn some of those pixels on and some off. You can then edit that bitmapped swipe dot by dot; that is, you can change any of the pixels in the image. Bitmaps can be created by a *scanner*, which converts drawings and photographs into electronic form, or by a human artist (like you) working with a *paint program*.

This painted image on the left, created with two brushstrokes, is bitmapped. On the right is an enlargement of a piece of the image, where you can see the individual pixels that have been mapped to the screen by the bits of information.

The simplest bitmaps are *monochrome,* which have only one color against a background. For these, the computer needs just a single *bit* of information for each pixel (remember, a bit is the smallest unit of data the computer recognizes). One bit is all it takes to turn the dot off (black) or on (white). To produce the image you see, the bits get "mapped" to the pixels on the screen in a pattern that displays the image.

In images containing more than black and white (such as colors or shades of gray, as you can in the close-up example above), you need more than one bit to specify the colors or shades of gray of each dot in the image. Multicolor images are bitmaps also. An image that can have many different colors or shades of gray is called a "deep bitmap," while a monochrome bitmap is known as a "bilevel bitmap." The "depth" of a bitmap—how many colors or shades it can contain—has a huge impact on how much memory and/or disk space the image consumes. A 256-color bitmap needs 8 times as much information, and thus disk space and memory, as a monochrome bitmap.

The *resolution* of a bitmapped image depends on the application or scanner you use to create the image, and the resolution setting you choose at the time. It's common to find bitmapped images with resolutions of 72 *dots per inch (dpi),* 144 dpi, 300 dpi, or even 600 dpi. A bitmap's resolution is permanently fixed—a bitmapped graphic created at 72 dpi will print at 72 dpi even on a 300 dpi printer such as the LaserWriter. On the other hand, you can never exceed the resolution of your output device (the screen, printer, or what have you); even though you scanned an image at 600 dpi, it still only prints at 300 dpi on a LaserWriter, since that's the LaserWriter's top resolution.

You can contrast bitmapped images with *vector* or *object-oriented* images, in which the image is represented by a mathematical description of the shapes involved. You can edit the **shapes** of an object graphic, but not the individual **dots**. On the other hand, object-oriented graphics are always displayed or printed at the maximum resolution of the output device. But keep in mind that an object-oriented graphic is still displayed as a bitmap on the screen.

bitmapped font

All fonts (typefaces) that you see on the screen are **bitmapped.** That's the only way the computer can display the typeface on the screen, since the screen is composed of dots (pixels). Some fonts have no other information to them than the bitmapped display you see on the screen, while other fonts have additional data that is used by the printer to print the typeface smoothly on a page (*outline,* or *scalable* fonts).

Please see the definition under *font* for a discussion of bitmapped and screen fonts, outline and printer fonts, resident and downloadable fonts, PostScript and TrueType fonts, etc..

bleed

Bleed refers to any element on a page that is printed beyond the edge of the paper. Whenever you see anything (text, graphics, photographs) that is printed right up to the edge of the paper, it was actually printed onto larger paper over the margin guidelines, and the paper was trimmed.

Some press operators might tell you, “No bleeds allowed.” Some might say, “We charge extra for bleeds.” They might ask, “Does it bleed?” You might say, “Can I bleed this line?” An image that bleeds off all four edges of the paper is called a “full bleed.” This page you are reading has no bleeds; for this page size and the press we used, bleeds were not allowed.

The reason the printer might not allow a bleed is because she plans to print the job on the right-size paper, without any trimming. If the image prints right up to the very edge of the paper, then the ink “bleeds” off the paper and gets all over the press and makes a mess.

bleeding edge

People who are on the leading edge in the computer field often feel like they are really on the **bleeding edge,** as they struggle with making incredibly complex new technology function properly, as they push the borders and find limitations, as they are overwhelmed with the acceleration of information, as the pace of their lifestyle increases with the pace of technological advances and what is expected of their machines.

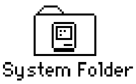
sigh . . .

blend

In many graphics programs you can choose to create a **blend,** where a color starts at one part of an object and blends into another color at another end of the object, like the sky at sunset. A good blend avoids *banding,* where you can visibly see the steps in the range of color.

blessed System Folder

With later versions of the Macintosh System, including System 7, the System Folder has a little icon of a Mac on its cover. This tells you that the folder is blessed (pronounced “bless edd,” with two syllables). What that really means is that this folder contains the System and the Finder that are running the computer.



The blessed System Folder

Some people make the dreadful mistake of putting more than one System Folder on their hard disk. (Don’t do that—the computer can get confused and check out. Like seriously crash.) Even though there may be more than one System Folder on a hard disk, only one of those folders should be blessed. Remove any System Folder you may find that is not blessed. (Look inside first to see if you want to keep any of the *extensions, control panels,* or other *files* that may be in there.)

block

In any kind of text editing, a **block** of text refers to any chunk you select, whether it is two pages, three paragraphs, a line, half a word, or even a single character. You can perform “block operations” on all the text in the block at once. When you take the block and move it somewhere else in the text, that’s a block move. “Block move” accomplishes the same thing as a *cut-and-paste.*

In telecommunications, a **block** is a group of characters that is transmitted as one unit. Blocks can be different lengths, and the longer the block, the faster the transmission goes, generally speaking.

BMUG

BMUG used to stand for the Berkeley Macintosh User Group, but now their official name is just BMUG and it doesn’t really stand for anything. But they are still the largest Macintosh *user group* in the world, with over 10,000 members in more than 50 countries. BMUG is an educational non-profit corporation dedicated to collecting, evaluating, and disseminating information about graphical interface computers (which is what the Mac is). They give users the information they need to use their computers most efficiently and painlessly.

BMUG’s semi-annual newsletter is a valuable tome of 400 pages, offering reviews, tutorials, reference material, and commentary—and no advertising. They run a Helpline with high-quality technical assistance and expert data recovery, available in person and over the phone every business day. The BMUG *BBS* (bulletin board service) is a multi-line information service linking EchoMac, UseNET, the Internet, and local conferences, as well as the latest files from their massive software library.

BMUG holds meetings in Berkeley, San Francisco, and San Jose, California. They also hold a large number of meetings for special interest groups *(SIGs)*, such as graphic design, beginners, networking, programmers, HyperCard, multimedia, etc. etc. etc.

board

A **board** (short for *printed circuit board),* or *card,* is a thin, rectangular sheet of fiberglass with *chips* and other electronic parts attached to it. Shallow metallic “traces” etched into the surface of the board serve as connecting wires between the parts. The main circuits of the computer are mounted on the *motherboard;* smaller boards that attach to other boards are called *daughterboards*; and boards designed to be plugged into slots inside your computer are called *add-in boards.* Please see Appendix A for several illustrations of boards and where they go.

boilerplate

A **boilerplate** is the part of a memo, newsletter, or any other regular communication or design that is repeated in every similar project. For instance, in your newsletter you use the same masthead across the top every month—that is the boilerplate. In an advertisement, the logo and the address are a boilerplate. In a mail *merge,* the entire letter is referred to as the boilerplate, except where the merge data goes.

A boilerplate is different from a template in that a template usually holds information that will change for each publication, while a boilerplate is exactly the same each time. For instance, you might have a template for a newsletter that has all the type set up and all the specifications prepared; each month you replace the existing type with the new text, and you change the volume and date. In this newsletter template, though, you probably have some boilerplate information—like the return address, the masthead, and perhaps a list of the Board of Directors—that stays the same every month.

bomb

Bomb is another term for *crash;* I suggest you read the definition for *crash* for more details on why this may happen to you and some tips on preventing it.

This is the bomb icon you will see.

The Macintosh turns this figure of speech into a visual image, putting a picture of an old-fashioned bomb on the screen when the system crashes. I don't think this is very nice. It lays a guilt trip on us, too: we always think it's our fault. But when you see this little bomb icon on your screen with little fizzies popping off the wick, it does not mean that you need to jump under your desk to avoid the shrapnel as your machine explodes. It just means that the software suffered a malfunction of considerable dimension and decided to check out. Occasionally you may have a clue as to why this happened (such as you ran out of *memory* or you opened very old *software* in a new *System*), but sometimes you just have to accept the fact that you will never know why.

~~If~~ When you get a bomb—the only thing you can do is turn your computer off for about fifteen seconds, and then back on again. If you have a *reset switch* (see *programmer's switch*) and you know where it is, use it. Everything you have not saved onto your disk will be destroyed, which is another good reason to save every few minutes. Really.

Boolean expression, Boolean logic

Boolean logic is a form of algebraic logic where every answer is either true or false. It's used to solve the kind of problems you buy a database program to handle: like, "show me the list of men who dance, *and* who cook *or* like washing dishes, *and* who are *not* married." (Boolean logic is big on **expressions** such as *and*, *or*, and *not.*) You use Boolean logic every time you search your database. Boolean logic comes in very handy on a *digital* computer (which is what you have) because each answer can be a *bit* that is either on or off, true or false. The computer uses Boolean logic in a great deal of its work.

boot, boot up

To **boot** or **boot up** means to start your computer system, usually by turning on the power and/or pushing the "on" button. It's called "booting" because the computer is going inside itself and turning itself on (doing a lot of preliminary checking and adjusting before it's ready to run your programs). Hence the machine is considered to be "pulling itself up by its own bootstraps."

A *cold boot* is when the power to the machine has been completely turned off and you turn it on with the power switch.

A *warm boot* is when the computer is already on but you want to *restart* it without actually turning off the power. You might do this because the system crashed or you want to clear out the *memory.*

On a Mac, you might warm boot to activate a new *extension* or *control panel.* To warm boot a Macintosh, choose Restart from the Special menu while at the Finder.

If you have a PC, maybe you want to activate a new *device driver* or deactivate a *memory resident program* that doesn't have an Unload command. Press *Ctrl Alt Del.*

Reboot simply means to start up the computer again, after it was already on. You can reboot by doing a warm boot, but sometimes you crash so bad that you have to turn off the power and then turn it back on with a cold boot.

bootleg, bootlegger

As an adjective or a verb, **bootleg** refers to illegally copied software. If you make a copy of a *non-shareware* program from a friend, you are **bootlegging** the program; you have a **bootleg** copy; and you're now a **bootlegger.** This term is synonomous with *pirate.*

Boston Computer Society

See *BCS.*

bps

The acronym **bps** stands for **b**its **p**er **s**econd, which is a measurement of how fast information (data) travels between two devices. When two modems converse via the telephone line, the speed of modem communications is measured in bps. (When people talk about modems, they often use the word "baud" to mean the same thing as bps, though technically the two terms are not synonymous. See *baud.*)

The modems most commonly used with personal computers run at 1200 and 2400 bits per second. At 1200 bps a modem can send about 1600 words through the phones lines per minute. At 2400 bps, it can send roughly 3200 words per minute.

Many vendors sell modems that run at 9600 bps, which is close to the maximum that standard metal phone wires can handle reliably. Much faster modems are available, however, and the newer fiber optic phone cables allow transmission rates of more than 500,000 bps. But no matter how good the phone line, be careful about buying a modem any faster than 2400 bps. It can only use the higher speeds if it's communicating with another modem that's equally fast, and if both modems use the same *protocol*, or system for communicating. (Watch out—many faster modems don't adhere to the "standard" protocols for high-speed communications.) Don't even think about installing a 300 bps modem, even if someone gives it to you for free. These things are obsolete now that you can buy 2400 bps models for well under $100.

You may also use a serial connection to link your printer to your computer, especially if you have a Macintosh (most PC printers are connected via the *parallel* port, but speed ratings for parallel connections aren't usually quoted). Typically, a computer communicates with a serial printer at 9600 or 19,200 bps. Under the right conditions, the *serial ports* on your computer can usually handle much higher speeds—57,600 bps for the Mac and 115,200 bps for the PC.

breadboard

A **breadboard** is a thin board, sometimes fiberglass and sometimes plastic, with lots of little holes arranged in a grid. Electronics engineers use breadboards to create prototypes of circuit boards by wiring chips, resistors, and other electronic parts onto the board by hand, with the connecting wires running underneath. The term sometimes refers to the finished prototype itself, but more often it's applied to the hole-filled board to which the electronic components are attached.

b-tree

The term **b-tree** refers to a way of organizing database information so that you can quickly search through it to find exactly what you're looking for. B-tree is a way of organizing database *keys* so you can quickly search them on disk.

A database contains collections of related information, similar to file cards. Each record contains several fields, such as name, address, date of birth, etc. We want to find something based on one of those pieces of information. We could search sequentially through the whole file, but imagine if you had to search through the entire database of California social security numbers one by one. A b-tree allows you to search efficiently for specific names or numbers even as you are adding or deleting from the database. It dynamically grows and shrinks and maintains an up-to-date index stored as a tree, hence the name. (The letter "b" is to differentiate from other types of previously existing trees.)

btw

This cute little acronym, **btw,** stands for "by the way." It's often used in online communications. Please see *baudy language* for more online terms of endearment.

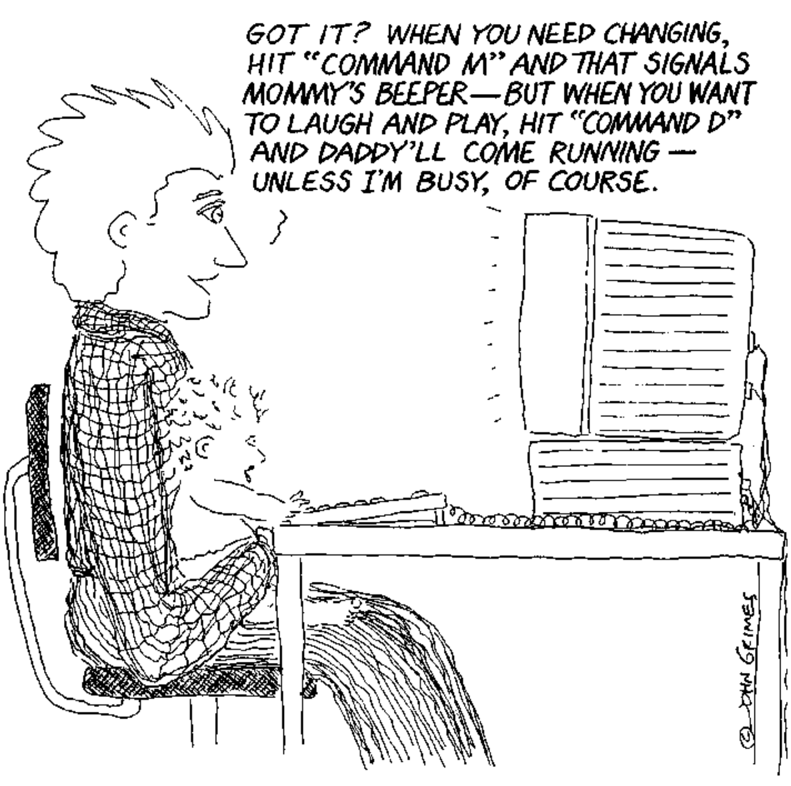
buffer

A **buffer** is a section of *memory* set aside for temporary storage (if you don't understand what memory is, you should probably read that definition first). For instance, you've probably noticed that you can quickly type a bunch of commands on the keyboard, and then just sit back and watch your computer carry out the commands one after another. This is because the computer stores the keys you pressed in a buffer—after doing one command, it goes back to the buffer to get the next one in line.

Some programs use the term **buffer** to refer to the space reserved in memory for the information of a file you're working with. Actually, you can start a new buffer and put information in it before you save it on disk as a file, but the idea is that the buffer represents a collection of information in memory that is handled as a unit.

Printers, modems, and other devices can have **buffers** too. If your printer doesn't have a buffer, your computer can only send it a little information at a time, as much as the printer can output immediately. With a buffer, the computer can send a document at full speed, at least until the buffer fills up. The information waits in the buffer until the printer is ready to print it, but meanwhile you can go back to work with your computer.

In DOS, the word **buffer** refers specifically to the small buffers reserved for DOS's use when transferring information to or from a disk. DOS buffers are a crude kind of *disk cache;* when information is available in a buffer, it doesn't need to be re-read from the disk, and your system operates faster. DOS automatically sets up a standard number of buffers (usually 15), but you can change this by entering a command such as **BUFFERS=20** in your *CONFIG.SYS* file. Why do this? Because, up to a point, the more buffers there are, the faster the system will run (although there's no practical way to figure out the ideal number of buffers). On the other hand, if you are using a true disk cache such as SMARTDrive or Super PC-Kwik, DOS buffers are redundant, wasting memory and actually slowing things down. In this situation you should always set the number of buffers to 3 or 4.


GOT IT? WHEN YOU NEED CHANGING, HIT "COMMAND M" AND THAT SIGNALS MOMMY'S BEEPER—BUT WHEN YOU WANT TO LAUGH AND PLAY, HIT "COMMAND D" AND DADDY'LL COME RUNNING — UNLESS I'M BUSY, OF COURSE.
© JOHN GRIMES

bug

The B word. If something is wrong with a piece of software or hardware so that it stops working or destroys your data or just acts weird, the product is said to have a **bug,** or to be **buggy.** The term actually comes down to us from the real live crawling and flying bugs that used to get into those old giant-sized computers and wreak havoc, particularly a moth found squished in the Mark II machine at Harvard in 1945.

Bugs are not your fault; they are malfunctions accidentally built into the hardware or software. Of course, it can be hard to know whether the problem you're facing is a bug or whether you're just doing something stupid. And occasionally a *virus* can make a program appear buggy.

Bugs can be minor or major. Here's an example of a minor bug: In PageMaker version 4.0 you could search for the characters "f i" and replace them with the *ligature* "fi." But in the dialog box in which you do this, when you typed the key sequence for the ligature "fi," PageMaker displayed the ligature "fl" (although it replaced the characters with the proper ligature). This bug was fixed in PageMaker 4.2.

Bugs give *beta testers* their mission in life, their raison d'être. Beta testers find and report the bugs in products that are under development so the bugs get exterminated (the item gets *debugged*) before we pay lots of money for the products. But almost always, some bugs are missed during the beta testing process, and they wind up in the version of the item that you buy anyway.

bullet •

In printed documents, a **bullet** is often used to indicate an item in a list, like this:

- Here's item one.
- Here's item two.
- Here's item old.
- Here's item new.

The little round mark (•) in front of each line is the bullet—at least the official one. Actually, you can use any character you like as a bullet. The Zapf Dingbats and Wingdings fonts contain lots of interesting symbols that can serve as bullets, and some fonts consist entirely of bullet-type symbols.

Zapf Dingbats ➟ ✶ ❼ ❅ ✆ ◆ ❉ ✾ ✃ ✵ ❒ ❑ ◗ ■ ❄ ❑ ❙ ✬ ✥ ✡ ✰ ✪ ⌘ ▼ ❁ ✎ ●
❦ ◆ ❉ ❤ ☛ ♣ ✗ ❋ ▲

To type the official bullet on the Macintosh, press Option 8 (hold the Option key down while you tap the 8). It's the same principle as typing an asterisk, in which you hold the Shift key down and hit the 8—if you hold the Option key down instead of the Shift key, you get the bullet. Also see *Key Caps*.

On the PC, if you can type a bullet at all, the keys you need depend on which program you're using—consult the manual. In PageMaker, press Ctrl Shift 8. In other Windows programs, hold down the Alt key and type 0149 on the numeric keypad. When you let up the Alt key, the bullet should appear. Also see *character map*.

bulletin board, bulletin board service

See *BBS*.

bundle

When you buy a particular product and it comes packaged with another product for free, often from another vendor, we say that the extra product is **bundled** with the main product. For instance, when I bought my scanner from Microtech, it came bundled with ImageStudio software from Letraset so I could work on the images I scanned.

And similarly, when you buy two or more products for a single price, you're buying a bundle—the products have been bundled together, or one product has been bundled with another. Usually a bundle is a better deal than buying the products separately, but not always—sometimes they're trying to get rid of an older version that you might not really want.

burn-in

A **burn-in test** is often performed at the factory before a product is shipped, or at the store before you buy it. The product or system is kept running continuously so if there are any weak components they will be discovered before the system becomes a vital part of a person's life. The longer the burn-in time, the more effective the test.

Though the phosphors used in today's color monitors are pretty resilient, this burn-in test still isn't a good idea for monitors, because if you leave a bright image on a computer screen for an extended period of time, the image can get burned into the phosphor coating inside the screen, causing **screen burn-in.**

To prevent screen burn-in, you can turn the brightness level down on the monitor, or get a *screen saver*, which keeps a constantly changing image floating or flying across the screen.

Burn-in also refers to a period of time in which a person is so intensely involved in the project on her computer, such as writing a stupid computer dictionary, that she forgets to eat, sleep, or feed her children. They say excessive burn-in can lead to **burn-out,** but personally I think burn-in can lead to burn-out only if the time necessary for the luxury of burn-out is available.

bus

A **bus** is a combination of software, hardware, and electrical wiring that creates a way for all the different parts of your computer to communicate with each other. If there was no bus, you would have an unwieldy number of wires connecting every part to every other part. It would be like having separate wiring for every light bulb and socket in your house. For instance, you see the cables plugged into the back of your computer that connect it with the monitor or the printer or the hard disk or some other device—they all connect into the bus.

There are different kinds of buses. There's a **data bus,** over which data flow (data is plural) back and forth between *memory,* your disk drives, and the *microprocessor* that runs the show inside your system. If your computer has *expansion slots,* there's an **expansion bus**. Messages and information pass between your computer and the *add-in boards* you plug in over the expansion bus. And there are other types of buses as well, such as the "address bus" and the *local bus.*

The design of the bus determines how fast information can flow and how efficiently the various parts of the system can use the system. The details are too complicated for this book, but one important factor is the bus's *bandwidth,* the definition of which you might want to look at.

bus error

A **bus error** is one of the most commonly occurring error codes to appear on your Mac and is therefore respectfully designated as "Error code ID 1" or "Type 1 error." Although most of these error codes can only be made sense of by programmers and technicians, you can often cure a repeating bus error by allocating more *memory* to the application that was active when the error occurred.

In System 7, this type of error can also mean that the application is not *32-bit clean* and you will either need to get an upgrade for the application or simply turn off *32-bit addressing* for the software to run properly. See *32-bit addressing.*

There are, of course, other reasons for a Type 1 error message on your screen, but the above-mentioned suggestions should be the first things you try. See *error codes.*

business graphics

Business graphics are those kinds of images like pie charts, bar charts, and scatter diagrams, etc., that are typically used by business people to show how many eggs they sold in Petaluma or how many days it takes to make a vacuum cleaner. Presentations that use slides during a speech are also lumped in with business graphics, even if they are straight text and not graphic at all.

There are several kinds of software that will draw charts and graphs for you—all you have to do is type in the information. You can then usually edit the graphic, either with the tools provided with the software or by exporting the chart to a graphic program. For instance, in most spreadsheet applications you can just tell the program what information to work with, and then hit a key or click a button. The application takes that information and creates a chart out of it. If you want really fancy graphs, or if you want an easy way to create text charts, you can buy separate business graphics, *charting*, or presentation programs.

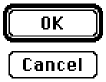
bus width

If you think of the computer's bus as being a freeway that carries data, then it is easy to visualize the **bus width** as the number of lanes on this freeway. Naturally, the more lanes available, the more data that can be carried at any given moment. Bus widths are generally designated in multiples of eight; a *32-bit* bus width has the capability to carry four times the information of an *8-bit* bus in the same amount of time. See *bus* and *8-bit, 24-bit, 32-bit computer.*

button

A **button** on your computer screen is the visual equivalent of a button on a machine: it's a thing that you push and then you expect something to happen. Of course, you can't push a button on the screen with your finger, except on the rare computer that has a *touch screen.* Instead, you click on the button with your mouse pointer.

You can often use the keyboard to press a button, too. If a button has a dark border around it, that means you can hit the Return or the Enter key to make it work. If there's an underlined letter in the name of the button, you can press that letter (or a modifier key plus that letter) to activate the button instead of picking up the mouse and clicking.

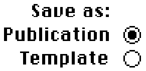


These are action buttons. If you press Return or Enter, the button with the dark border will activate.

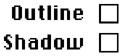
In Windows, the dark border is kind of hard to see. And in Windows, you can activate a button by pressing Tab until the dotted rectangular highlight surrounds the button's label, and then press the Spacebar.

Buttons comes in several varieties. The different kinds of buttons give you a visual clue as to what to expect. Plain ol' buttons (sometimes they're called "action buttons") activate something as soon as you click on them (click once!). You might find an ordinary button in a tool bar (or button bar or palette) on your application's main screen.

Groups of buttons are used in dialog boxes to offer you choices among related items. In a group of *radio buttons*, you can only choose one, just like on your radio. In a group of *checkbox* buttons (or just checkboxes), you can check any number of the boxes from zero to all, just as on a list of checkboxes on a form.


Save as:
Publication ●
Template ○

Radio buttons


Outline □
Shadow □

Checkbox buttons

byte

A **byte** is eight *bits*, eight individual electronic on/off signals, strung together to make a message that the computer can interpret. Although a bit is the smallest unit of information a computer can work with, most of the information it handles is organized into bytes. For example, most computers use one byte to represent each character (letter, number, or other symbol) that you see on your screen. A really small file, like a *batch file* on a PC or a one-page word-processed letter on the Mac, might consist of 20 bytes to 150 bytes.

The information-storing capacity of a computer is usually measured in multiples of bytes. A series of 1024 bytes strung together is a *kilobyte*. Lower-capacity *floppy disks,* for example, hold 360 kilobytes (KB or just K) or 800K of data; old PCs came with just 256K or 640K of *memory (RAM).*

A *megabyte* (MB) is 1024 kilobytes. Your computer probably has 2 to 8 megabytes of RAM and a hard disk that can hold 20 to 260 MB of data.