communicate and "tweet" allow only about 160 characters per message (140 bytes). As today's most pervasive method of human-to-human data communication (more than three billion texters worldwide sending trillions of short text messages from phone to phone each year), users speaking many languages quickly developed word shortcuts to pack the most meaning into the fewest characters. NetLingo (www.netlingo.com) maintains a list of almost two thousand English acronyms and instant messaging jargon words such as XOXO (hugs & kisses), U (you), and NME (enemy). When assembled into a message, you might discover "were I a tear in ur eye i wood roll down onto ur lips. but if u were a tear in my eye i wood never cry as i wood be afraid 2 lose u!" With the arrival of MMS (Multimedia Messaging Service), which allows for 350,000-byte transmissions, perhaps these shortcut spellings will fade away. But perhaps not.

About Fonts and Faces

A **typeface** is a family of graphic characters that usually includes many type sizes and styles. A **font** is a collection of characters of a single size and style belonging to a particular typeface family. Typical font **styles** are boldface and italic. Your computer software may add other style **attributes**, such as underlining and outlining of characters. Type sizes are usually expressed in points; one **point** is 0.0138 inch, or about 1/72 of an inch. The font's size is the distance from the top of the capital letters to the bottom of the descenders in letters such as *g* and *y*. Helvetica, Times, and Courier are typefaces; Times 12-point italic is a font. In the computer world, the term font is commonly used when typeface or face would be more correct.

A font's size does not exactly describe the height or width of its characters. This is because the **x-height** (the height of the lowercase letter *x*) of two fonts may vary, while the height of the capital letters of those fonts may be the same (see Figure 2-1). Computer fonts automatically add space below the descender (and sometimes above) to provide appropriate line spacing, or **leading** (pronounced "ledding," named for the thin strips of lead inserted between the lines by traditional typesetters).

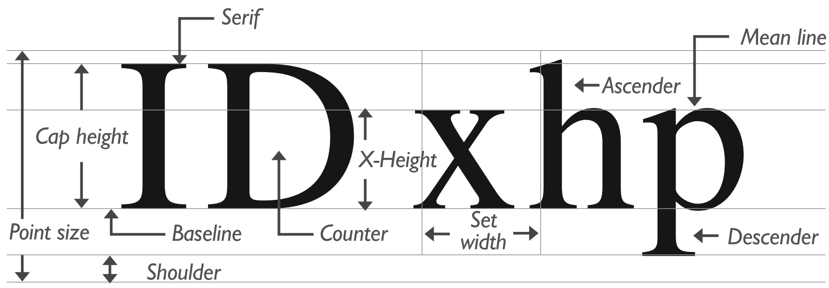


Figure 2-1 The measurement of type

Leading can be adjusted in most programs on both the Macintosh and the PC. Typically you will find this fine-tuning adjustment in the Text menu of image-editing programs or the Paragraph menu of word processing programs, though this is not an official standard. No matter where your application has placed the controls for leading, you will need to experiment with them to achieve the best result for your font. With a font editing program like Fontographer from Fontlab, Ltd. at www.fontlab.com (you'll see an example of it later in the chapter), adjustments can also be made along the horizontal axis of text. In this program the character metrics of each character and the kerning of character pairs can be altered. **Character metrics** are the general measurements applied to individual characters; **kerning** is the spacing between character pairs. When working with PostScript, TrueType, and Master fonts—but not bitmapped fonts—(see "Computers and Text" later in this chapter), the metrics of a font can be altered to create interesting effects. For example, you can adjust the body width of each character from regular to **condensed** to **expanded**, as displayed in this example using the Sabon font:

Regular
Condensed
Expanded

Or you can adjust the spacing between characters (**tracking**) and the kerning between pairs of characters:

Tighter Track	Av	Av
Looser Track	Kerned	Unkerned

When it converts the letter *A* from a mathematical representation to a recognizable symbol displayed on the screen or in printed output (a process called **rasterizing**), the computer must know how to represent the letter using tiny square **pixels** (picture elements), or dots. It does this according to the hardware available and your specification, from a choice of available typefaces and fonts. Search for "free fonts." High-resolution monitors and printers can make more attractive-looking and varied characters because there are more fine little squares or **dots per inch (dpi)**. And today's broad selection of software fonts makes it easier to find the right typeface and font for your needs. The same letter can look very different when you use different fonts and faces:

A A A A A A A A A

Cases

In centuries when type was set by hand, the type for a single font was always stored in two trays, or *cases*; the upper tray held capital letters, and the lower tray held the small letters. Today, a capital letter is called **uppercase**, and a small letter is called **lowercase**.

TIP *Studies have shown that words and sentences with mixed upper- and lowercase letters are easier to read than words or sentences in all caps (uppercase). While uppercase can make your message appear important or urgent, use this sparingly; in online messaging it's known as "SHOUTING" or "YELLING" and can be annoying, if not offensive.*

In some situations, such as for passwords, a computer is **case sensitive**, meaning that the text's upper- and lowercase letters must match exactly to be recognized. But nowadays, in most situations requiring keyboard input, all computers recognize both the upper- and lowercase forms of a character to be the same. In that manner, the computer is said to be **case insensitive**.

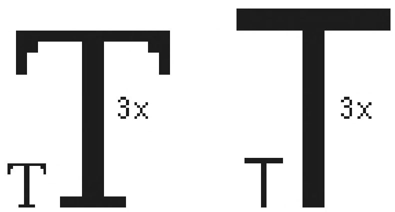
WARNING *The directory names and filenames used in Uniform Resource Locator (URL) addresses on the Internet are case sensitive! Thus, http://www.timestream .com/info/people/biotay/biotay1.html points to a different directory and file than http://www.timestream.com/info/people/bioTay/biotay1.html. On the other hand, the record type (HTTP) and the domain name (www.timestream.com), and e-mail addresses (tay@timestream.com) as well, are usually case insensitive. Read more about addresses on the Internet in Chapter 12.*

Company and product names such as WordPerfect, OmniPage, PhotoDisc, FileMaker, and WebStar have become popular. Placing an uppercase letter in the middle of a word, called an **intercap**, is a trend that emerged from the computer programming community, where coders discovered they could better recognize the words they used for variables and commands when the words were lowercase but intercapped.

Serif vs. Sans Serif

Typefaces can be described in many ways, just as a home advertised by a realtor, a wine described by a food critic, or a political candidate's platform can all be described in many ways. Type has been characterized as feminine, masculine, delicate, formal, capricious, witty, comic, happy, technical, newsy—you name it. But one approach for categorizing typefaces is universally understood, and it has less to do with the reader's response to the type than it does with the type's mechanical and historical properties. This approach uses the terms **serif** and **sans serif**.

Serif versus sans serif is the simplest way to categorize a typeface; the type either has a serif or it doesn't (*sans* is French for "without"). The serif is the little decoration at the end of a letter stroke. Times, New Century Schoolbook, Bookman, and Palatino are examples of serif fonts. Helvetica, Verdana, Arial, Optima, and Avant Garde are sans serif. Notice the difference between serif (on the left) and sans serif:



On the printed page, serif fonts are traditionally used for body text because the serifs are said to help guide the reader's eye along the line of text. Sans serif fonts, on the other hand, are used for headlines and bold statements. But the computer world of standard, 72 dpi monitor resolution is not the same as the print world, and it can be argued that sans serif fonts are far more legible and attractive when used in the small sizes of a text field on a screen. Indeed, careful selection of a sans serif font designed to be legible in the small sizes (such as Tahoma or Verdana) makes more sense when you are presenting a substantial amount of text on the screen. The Times font at 9-point size may look too busy and actually be difficult and tiring to read. And a large, bold serif font for a title or headline can deliver a message of elegance and character in your graphic layout. Use what is right for your delivery system, which may not necessarily be the same as what is right when you're printing the material to paper. This is because when you're printing out what you create on a computer monitor, **WYSIWYG** (What You See Is What You Get) is more of a goal than an absolute fact.

Using Text in Multimedia

Imagine designing a project that used no text at all. Its content could not be at all complex, and you would need to use many pictures and symbols to train your audience how to navigate through the project. Certainly voice and sound could guide the audience, but users would quickly tire of this because greater effort is required to pay attention to spoken words than to browse text with the eye.

A single item of menu text accompanied by a single action (a mouse click, keystroke, or finger pressed to the monitor) requires little training and is clean and immediate. Use text for titles and headlines (what it's all about), for menus (where to go), for navigation (how to get there), and for content (what you see when you get there).

TIP *In designing your navigation system, bring the user to a particular destination with as few actions and as short a wait as possible. If the user never needs the Help button to get there or never has to click the Back button when at a dead end, you're doing everything right!*

Designing with Text

Computer screens provide a very small workspace for developing complex ideas. At some time or another, you will need to deliver high-impact or concise text messages on the computer screen in as condensed a form as possible. From a design perspective, your choice of font size and the number of headlines you place on a particular screen must be related both to the complexity of your message and to its venue.

If your messages are part of an interactive project or web site where you know the user is seeking information, you can pack a great deal of text information onto the screen before it becomes overwhelmingly busy. Seekers want dense material, and while they travel along your navigational pathways, they will scroll through relevant text and study the details. Here is where you must strike a balance, however. Too little text on a screen requires annoying page turns and unnecessary mouse clicks and waits; too much text can make the screen seem overcrowded and unpleasant.

On the other hand, if you are creating presentation slides for public-speaking support, the text will be keyed to a live presentation where the text accents the main message. In this case, use bulleted points in large fonts and few words with lots of white space. Let the audience focus on the speaker at the podium, rather than spend its time reading fine points and subpoints projected on a screen.

TIP *A lengthy text document read by a web browser may scroll for hundreds of lines without annoying the user because it's expected. As a rule of thumb, however, try to make your web pages no longer than one-and-a-half to two screenfuls of text. On a 1024 × 768–pixel monitor, for example, you have about 600 pixels in height to work with before scrolling is necessary while viewing web content in a browser. Limit the width of your lines by using columns—reading a line of text across an entire 21-inch monitor screen is cumbersome, if not uncomfortable, For printing text documents, provide a separate link to a complete document in either* font a reader will use to view your document—the default display font is a preference that can be set in the viewer's browser, which knows it's installed on that viewer's machine. So some viewers may read your words in serif Times Roman, others in sans serif Helvetica or Arial.

Computers and Text

Very early in the development of the Macintosh computer's monitor hardware, Apple chose to use a resolution of 72 pixels per inch. This matches the standard measurement of the printing industry (72 points per inch) and allows desktop publishers and designers to see on the monitor what their printed output will look like (WYSIWYG). In addition, Apple made each pixel square-shaped, providing even measurements in all directions. Until the Macintosh was invented, and the VGA video standard set for the PC (at 96 pixels per inch), pixels were typically taller than they were wide. The aspect ratio for a pixel on older EGA monitors, for example, is 1.33:1, taller than it is wide. VGA and SVGA monitor resolutions for both Macintosh and Windows display pixels at an aspect ratio of 1:1 (square).

The Font Wars Are Over

In 1985, the desktop publishing revolution was spearheaded by Apple and the Macintosh computer, in combination with word processing and page layout software products that enabled a high-resolution 300 dpi laser printer using special software to "draw" the shapes of characters as a cluster of square pixels computed from the geometry of the character. This special software was the **Adobe PostScript** page description and **outline font** language. It was licensed by Apple and included in the firmware of Apple's LaserWriter laser printer.

PostScript is really a method of describing an image in terms of mathematical constructs (Bézier curves), so it is used not only to describe the individual characters of a font but also to describe entire illustrations and whole pages of text. Because each PostScript character is a mathematical formula, it can be easily scaled bigger or smaller so it looks right whether drawn at 24 points or 96 points, whether the printer is a 300 dpi LaserWriter or a high-resolution 1200, 2400, or even 3600 dpi image setter suitable for the finest print jobs. And the PostScript characters can be drawn much faster than in the old-fashioned way. Before PostScript, the printing software looked up the character's shape in a bitmap table containing a representation of the pixels of every character in every size. PostScript quickly became the de facto industry font and printing standard for desktop publishing and played a significant role in the early success of Apple's Macintosh computer.

There are two kinds of PostScript fonts: Type 3 and Type 1. Type 3 font technology is *older* than Type 1 and was developed for output to printers; it is rarely used by multimedia developers. There are currently over 6,000 different Type 1 typefaces available. Type 1 fonts also contain **hints**, which are special instructions for grid-fitting to help improve resolution. Hints can apply to a font in general or to specific characters at a particular resolution.

Other companies followed Adobe into the desktop publishing arena with their own proprietary and competitive systems for scalable outline fonts. In 1989, Apple and Microsoft announced a joint effort to develop a "better and faster" quadratic curves outline font methodology, called **TrueType**. In addition to printing smooth characters on printers, TrueType would draw characters to a low-resolution (72 dpi or 96 dpi) monitor. Furthermore, Apple and Microsoft would no longer need to license the PostScript technology from Adobe for their operating systems. Because TrueType was based on Apple technology, it was licensed to Microsoft. Adobe and Microsoft then developed a new and improved font management system incorporating the best features of both PostScript and TrueType, and by 2007, **OpenType** became a free, publicly available international standard. The font wars were over.

WARNING *TrueType, OpenType, and PostScript fonts do not display (or print) exactly the same, even though they may share the same name and size. The three technologies use different formulas. This means that word-wrapping in a text field may change. So if you build a field or a button that precisely fits text displayed with a PostScript font, be aware that if you then display it with the same font in TrueType or OpenType, the text may be truncated or wrapped, wrecking your layout.*

Font Foundries

Today collections of fonts are available through retail channels or directly from their manufacturers. Typefaces are created in a **foundry**, a term much like case, that has carried over from times when lead was poured into molds to make letter faces. There is also a special interest group (SIG) at America Online (go to Computing:Software Libraries:Desktop & Web Publishing Forum:Fonts) where people who enjoy designing and making interesting fonts post them for others to download—hundreds and hundreds of them with names like Evil of Frankenstein, CocaCola, Kerouac, LED, PonchoVia (sic), Spaghetti, TreeFrog, and Sassy. When you purchase some applications, such as CorelDraw or Adobe Illustrator, many extra fonts are included for free.

www.typequarry.com/
www.oldfonts.com/
www.myfonts.com/
www.bitstream.com/
www.will-harris.com/

Commercial type foundries and font sites. These gateways lead to a discussion of fonts and where to find them. With *Esperfonto,* Will Harris provides an interesting tool for making font decisions: Casual or Formal, Body or Display, Friendly or Serious, Cool or Warm, Modern or Traditional.

WARNING *It is easy to spend hours and hours downloading neat and interesting fonts; they are like the midnight snack table on a Caribbean cruise liner—ice carvings and delectable goodies laid out as far as the eye can see.*

Character Sets and Alphabets

Knowing that there is a wide selection of characters available to you on your computer and understanding how you can create and use special and custom-made characters will broaden your creative range when you design and build multimedia projects.

The ASCII Character Set

The **American Standard Code for Information Interchange (ASCII)** is the 7-bit character coding system most commonly used by computer systems in the United States and abroad. ASCII assigns a number or value to 128 characters, including both lower- and uppercase letters, punctuation marks, Arabic numbers, and math symbols. Also included are 32 control characters used for device control messages, such as carriage return, line feed, tab, and form feed.

ASCII code numbers always represent a letter or symbol of the English alphabet, so that a computer or printer can work with the number that represents the letter, regardless of what the letter might actually look like on the screen or printout. To a computer working with the ASCII character set, the number 65, for example, always represents an uppercase letter *A.* Later, when displayed on a monitor or printed, the number is turned into the letter.

ASCII was invented and standardized for analog teletype communication early in the age of bits and bytes. The capabilities of the technology have now moved far beyond the original intent of the standard, but because millions of installed computers and printers use ASCII, it is difficult to set any new standards for text without the expense and effort of replacing existing hardware. At least, for these 128 characters, most computers and printers share the same values.

The Extended Character Set

A byte, which consists of eight bits, is the most commonly used building block for computer processing. ASCII uses only seven bits to code its 128 characters; the eighth bit of the byte is unused. This extra bit allows another 128 characters to be encoded before the byte is used up, and computer systems today use these extra 128 values for an extended character set. The extended character set is most commonly filled with ANSI (American National Standards Institute) standard characters, including often-used symbols, such as ¢ or ∞, and international diacritics or alphabet characters, such as *ä* or *ñ*. This fuller set of 255 characters is also known as the ISO-Latin-1 character set; it is used when programming the text of HTML web pages.

NOTE *The rules for encoding extended characters are not standardized. Thus ASCII value 165, for example, may be a bullet (•) character on the Macintosh or the character for Japanese yen (¥) in Windows (ANSI).*

Unicode

As the computer market has become more international, one of the resulting problems has been handling the various international language alphabets. It was at best difficult, and at times impossible, to translate the text portions of programs from one script to another. For example, the differences between the Latin script (also known as "Roman") used by western European writers and the kanji script used by Japanese writers made it particularly challenging to transfer innovative programs from one market to another.

Since 1989, a concerted effort on the part of linguists, engineers, and information professionals from many well-known computer companies has been focused on a 16-bit architecture for multilingual text and character encoding. Called **Unicode**, the original standard accommodated up to about 65,000 characters to include the characters from all known languages and alphabets in the world.

Figure 2-5 Writing systems currently in use around the world. Unicode provides a consistent methodology for encoding the characters of any alphabet.

> While 65,000 characters are sufficient for encoding most of the many thousands of characters used in major languages of the world, the Unicode standard and ISO/IEC 10646 now support three encoding forms that use a common repertoire of characters but allow for encoding as many as a million more characters. This is sufficient for all known character encoding requirements, including full coverage of all historic scripts of the world, as well as common notational systems.
>
> *The Unicode® Standard: A Technical Introduction (www.unicode.org/unicode/standard/principles.html)*

Where several languages share a set of symbols that have a historically related derivation, the shared symbols of each language are unified into collections of symbols (called **scripts**). A single script can work for tens or even hundreds of languages (for example, the Latin script used for English and most European languages). Sometimes, however, only one script will work for a language (such as the Korean Hangul). Figure 2-5 shows a map of writing systems used in the world today.

The Unicode standard includes more than 18,000 Han characters (ideographs for Japanese, Chinese, and Korean) as well as obsolete alphabets such as cuneiform, hieroglyphs, and ancient Han characters. In addition, character space is reserved for users and publishers to create their own scripts, designed especially for their own applications. For example, a carpenter might develop a script that included a character meaning "half-inch Sheetrock," another character meaning "three-quarter-inch plywood," and so forth. HTML allows access to the Unicode characters by numeric reference. Thus 水 (in hexadecimal) represents the Chinese character for water:

水

Mapping Text Across Platforms

If you build your multimedia project on a Windows platform and play it back on a Macintosh platform (or vice versa), there will be subtle (and sometimes not-so-subtle) differences. Fonts are perhaps the greatest cross-platform concern, because they must be mapped to the other machine. If a specified font doesn't exist on the target machine, a substitute must be provided that does exist on the target. This is **font substitution**. In many cross-platform-savvy applications, you can explicitly define the **font mapping**. Table 2-3 shows some typical mappings when crossing platforms.

Mac→Win	Win→Mac
Mac:Chicago→Win:System	Win:Arial→Mac:Helvetica
Mac:Courier→Win:Courier New	Win:Courier→Mac:Courier
Mac:Geneva→Win:MS Sans Serif	Win:Courier New→Mac:Courier
Mac:Helvetica→Win:Arial	Win:MS Serif→Mac:New York
Mac:Monaco→Win:Terminal	Win:MS Sans Serif→Mac:Geneva
Mac:New York→Win:MS Serif	Win:Symbol→Mac:Symbol Map None
Mac:Symbol→Win:Symbol Map None	Win:System→Mac:Chicago
Mac:Times→Win:Times New Roman (sizes: 14→12, 18→14, 24→18, 30→24)	Win:Terminal→Mac:Monaco
Mac:Palatino→Win:Times New Roman	Win:Times New Roman→Mac:Times (sizes: 12→14, 14→18, 18→24, 24→30)

Table 2-3 Typical Mappings for Common Macintosh and Windows Fonts

TIP *Never assume that the fonts you have installed on your computer will also be installed on another person's computer. Pay attention to the way you include fonts in a project so that you never face the nightmare of your carefully picked fonts being replaced by an ill-suited default font like Courier (see the next "First Person"). If your work is being distributed to sites that may not have the fonts you are using, or if you do not license these fonts for distribution with your work, be sure to bitmap the special font text you use for titles, headlines, buttons, and so forth. For text to be entered by users, it is safest to stay with the installed Windows or Macintosh fonts, because you know they are universally available on that platform. In Windows, use the TrueType fonts installed during the Setup procedure.*

First Person

We had a short break between sessions to install the software for a panel discussion about multimedia. Four of us brought media with discussion material. Our moderator installed her presentation first, and we heard her wail, "Something's wrong with my fonts!" We all looked at her ugly 48-point Courier and felt sorry for her; we knew her mistake. The beautiful fonts she had installed on her home system were not installed in the system of the computer being used for the big-screen projector, and she had failed to bring the fonts along. By then, it was too late anyway.

Always be sure your fonts travel with your application when you are delivering software to run on a hardware platform other than the one you used to create the application. To avoid many font display problems, particularly for menus and headlines, you may wish to snap a picture of your text with a screen capture utility and use this image, or bitmap, instead of text that you type into a text field. (Chapter 3 describes bitmaps and how to capture and edit images.) This will ensure that the screen always looks right, regardless of what hardware platform you use or what fonts are installed.

It is not just fonts that are problematic; characters, too, must be mapped across platforms. Character mapping allows bullets, accented characters, and other curious characters that are part of the extended character set on one platform to appear correctly when text is moved to the other platform. Curly quotation marks, for example, rarely, if ever, map successfully across platforms.

Languages in the World of Computers

In modern Western languages, words are made up of symbols or letters strung together, representing as a whole the sounds of a spoken word. This is not so for Eastern languages such as Chinese, Japanese, and Korean (and the ancient languages of Sumeria, Egypt, and Mesopotamia). In these languages, an entire concept might be represented by a single word symbol that is unrelated to a specific phonetic sound.

The letters or symbols of a language are its alphabet. In English, the alphabet consists of 26 Roman or Latin letters; in Japanese, the kanji alphabet comprises more than 3,000 kanas, or whole words. The Russian alphabet, made up of Cyrillic characters based on the ancient Greek alphabet, has about the same number of letters as the Roman alphabet. All languages, from Navajo to Hebrew, have their own unique alphabets.

display the full-screen working area of your project or presentation and still have space to put your tools and other menus. This is particularly important in an authoring system such as Flash or Director, where the edits and changes you make in one window are immediately visible in the presentation window—provided the presentation window is not obscured by your editing tool! During development there is a lot of cutting and pasting among windows and among various applications, and with an extra monitor, you can open many windows at once and spread them out. Both Macintosh and Windows operating systems support this extra hardware.

NOTE *A few weeks of having to repeatedly bring windows to the front, and then hide them again to see the results of your editing, will probably convince you to invest in a second monitor.*

TIP *Your operating system has keyboard shortcuts for moving among windows and applications, accessing your desktop, and for standard commands such as cut, copy, paste, and undo. Learning to use these commands—and using them consistently—will enable you to work efficiently.*

Making Still Images

Still images may be small or large, or even full screen. They may be colored, placed at random on the screen, evenly geometric, or oddly shaped. Still images may be a single tree on a wintry hillside; stacked boxes of text against a gray, tartan, or Italian marble background; an engineering drawing; a snapshot of your department manager's new BMW. Whatever their form, still images are generated by the computer in two ways: as **bitmaps** (or paint graphics) and as **vector-drawn** (or just plain "drawn") graphics. Bitmaps may also be called "raster" images. Likewise, bitmap editors are sometimes called "painting" programs. And vector editors are sometimes called "drawing" programs.

Bitmaps are used for photo-realistic images and for complex drawings requiring fine detail. Vector-drawn objects are used for lines, boxes, circles, polygons, and other graphic shapes that can be mathematically expressed in angles, coordinates, and distances. A drawn object can be filled with color and patterns, and you can select it as a single object. The appearance of both types of images depends on the display resolution and capabilities of your computer's graphics hardware and monitor. Both types of images are stored in various file formats and can be translated from one application to another or from one computer platform to another. Typically, image files are compressed to save memory and disk space; many bitmap image file formats already use compression within the file itself—for example, **GIF**, **JPEG**, and **PNG**.

Still images may be the most important element of your multimedia project or web site. If you are designing multimedia by yourself, put yourself in the role of graphic artist and layout designer. Take the time necessary to discover all the tricks you can learn about your drawing software. Competent, computer-literate skills in graphic art and design are vital to the success of your project. Remember—more than anything else, the user's judgment of your work will be heavily influenced by the work's visual impact.

Bitmaps

A **bit** is the simplest element in the digital world, an electronic digit that is either on or off, black or white, or true (1) or false (0). This is referred to as **binary**, since only two states (on or off) are available. A map is a two-dimensional matrix of these bits. A bitmap, then, is a simple matrix of the tiny dots that form an image and are displayed on a computer screen or printed.

First Person

A few years ago a large corporation asked us and one other multimedia developer to bid on a long-term contract for computer-based training. Though busy with other active projects, we didn't want this possibly lucrative opportunity to slip by, so we spent a few days hastily putting together a demonstration of our technical skills for building nifty databases, designing tricky telecommunications systems, and integrating live video. We even "wire-framed" a bit of a working multimedia database with real data we got from the corporation.

We showed our demo to about a dozen management and training executives, in a fancy boardroom that had a built-in projector and sound system with mixers and light dimmers—a place where we could knock the socks off anybody. But within 30 seconds, the disaster bells started tinkling: most of our presentation was going way over their heads. Afterward, there were one or two vague questions and some thank-you's.

Our competitor's presentation, on the other hand, provided a slick series of finely rendered bitmapped screen images and elegant visuals. It was heavy on pretty menu screens and very light on how-it-is-done technology. We later learned that one of their graphic artists had worked for two solid weeks on the color bitmaps for that demo. In the follow-up phone call, we were told by our potential clients that the competition's "incredible artwork" had won out over our "excellent technology demonstration."

To cover our disappointment, we mumbled something to ourselves about not wanting to work with computer illiterates, anyway—people who could be taken to the cleaners by fresh paint. But we knew we'd missed a hefty piece of contract work because we hadn't invested serious graphic art talent in our demonstration. We decided that's why the real peas in the can are never the same bright green as the ones on the label. So we learned a marketing lesson.

A one-dimensional matrix (1-bit depth) is used to display monochrome images—a bitmap where each bit is most commonly set to black or white. Depending upon your software, any two colors that represent the on and off (1 or 0) states may be used. More information is required to describe shades of gray or the more than 16 million colors that each picture element might have in a color image, as illustrated in Figure 3-1. These picture elements (known as **pels** or, more commonly, **pixels**) can be either on or off, as in the 1-bit bitmap, or, by using more bits to describe them, can represent varying shades of color (4 bits for 16 colors; 8 bits for 256 colors; 15 bits for 32,768 colors; 16 bits for 65,536 colors; 24 bits for 16,772,216 colors). Thus, with 2 bits, for example, the available zeros and ones can be combined in only four possible ways and can, then, describe only four possible colors:

Bit Depth	Number of Colors Possible	Available Binary Combinations for Describing a Color
1-bit	2	0, 1
2-bit	4	00, 01, 10, 11
4-bit	16	0000, 0001, 0011, 0111, 1111, 0010, 0100, 1000, 0110, 1100, 1010, 0101, 1110, 1101, 1001, 1011

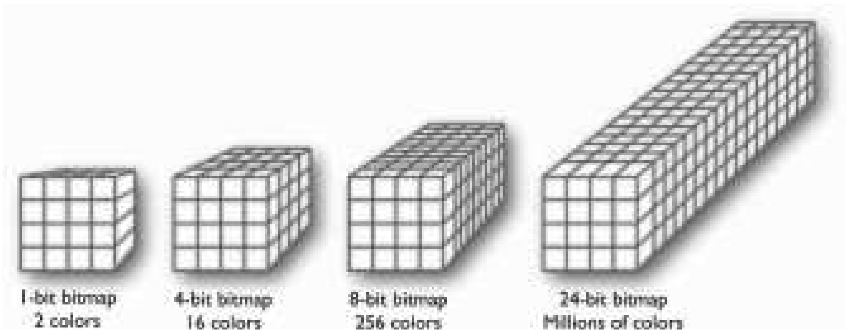


Figure 3-1 A bitmap is a data matrix that describes the characteristics of all the pixels making up an image. Here, each cube represents the data required to display a 4 × 4–pixel image (the face of the cube) at various color depths (with each cube extending behind the face indicating the number of bits—zeros or ones—used to represent the color for that pixel).

Together, the state of all the pixels on a computer screen make up the image seen by the viewer, whether in combinations of black and white or colored pixels in a line of text, a photograph-like picture, or a simple background pattern. Figure 3-2 demonstrates various color depths and compression formats. Image 1 is 24 bits deep (millions of colors); Image 2 is dithered to 8 bits using an adaptive palette (the best 256 colors to represent the image); Image 3 is also dithered to 8 bits, but uses the Macintosh system palette (an optimized standard mix of 256 colors). Image 4 is dithered to 4 bits (any 16 colors); Image 5 is dithered to 8-bit gray-scale (256 shades of gray); Image 6 is dithered to 4-bit gray-scale (16 shades of gray); and Image 7 is dithered to 1 bit (two colors—in this case, black and white).

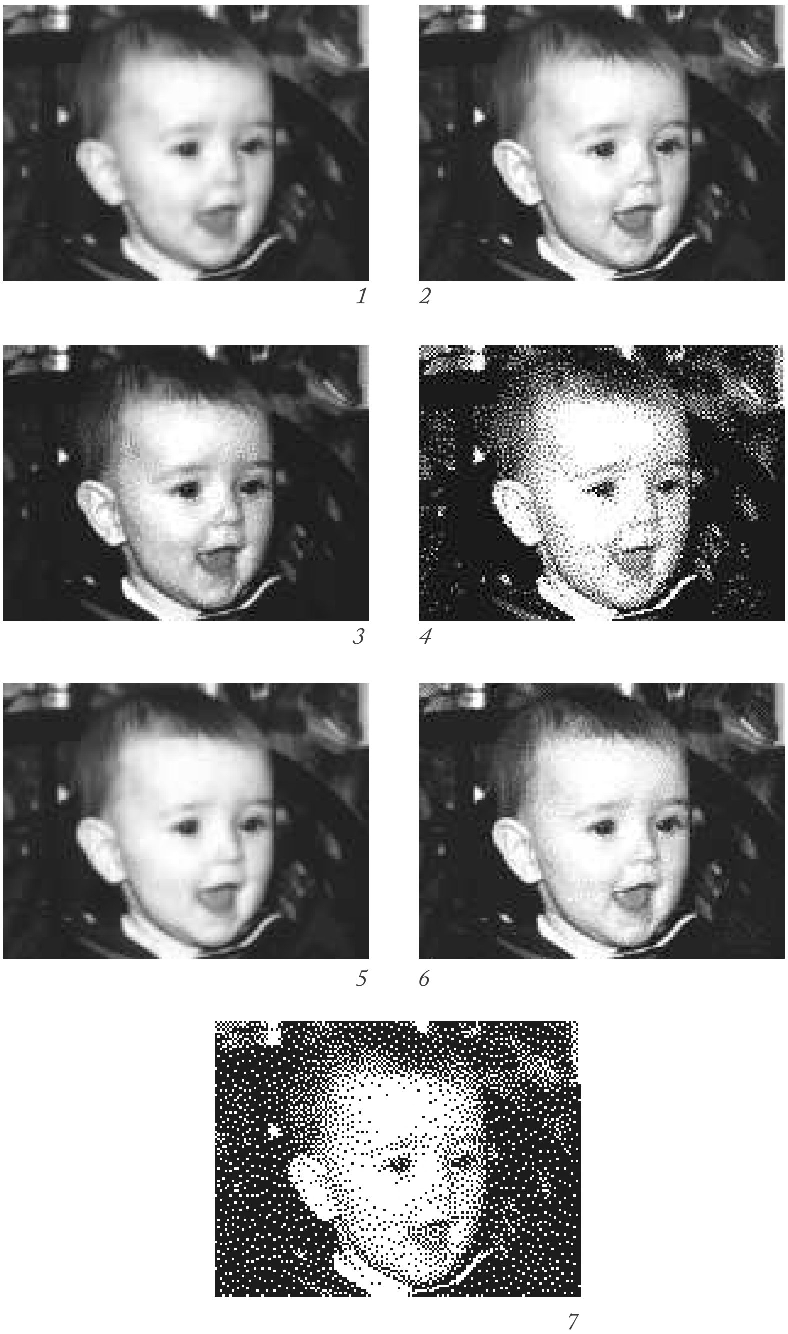


Figure 3-2 These images show the color depth of bitmaps as described in Figure 3-1. Note that Images 4 and 6 require the same memory (same file size), but the gray-scale image is superior. If file size (download time) is important, you can dither GIF bitmap files to the lowest color depth that will still provide an acceptable image.

Is there a colour scheme that will appear coloured or at least solid black for the colour-impaired?
If you're gathering empirical evidence, I have something called red-green colour blindness (it is quite common in males). It doesn't mean that you don't know which traffic light is showing! What it means mainly is that the *tone* of red-type colours doesn't seem so different to the tone of greens—the obvious case is a poppy field. I can see the poppies as red OK if I look carefully or they are pointed out to me, but other people see them kind of exploding out of the green... For people like me, a vibrant yellow always works. I read somewhere that black on yellow is a reliable "strong" combination. Certainly it is used by one of the motoring organisations in the UK for special diversion notices and the like.

Graham Samuel, Educational Software Developer, The Living Fossil Co., London

Bitmap Sources

Where do bitmaps come from? How are they made? You can do the following:

- Make a bitmap from scratch with a paint or drawing program.
- Grab a bitmap from an active computer screen with a screen capture program, and then paste it into a paint program or your application.
- Capture a bitmap from a photo or other artwork using a scanner to digitize the image.
- Once made, a bitmap can be copied, altered, e-mailed, and otherwise used in many creative ways.

If you do not want to make your own, you can get bitmaps from suppliers of clip art, and from photograph suppliers who have already digitized the images for you. Libraries of clip art are available on CD-ROMs and downloadable through online services. Many graphics applications are shipped with clip art and useful graphics. A clip art collection may contain a random assortment of images, or it may contain a series of graphics, photographs, sound, and video related to a single topic. Some 3-D modeling programs incorporate libraries of pre-made 3-D models into the application, allowing you to drag and drop common objects into a scene.

You can also download an image bitmap from a web site: in most browsers right-click over the image to see a menu of options. Choose "Download image to disk," "Copy Image," or "Save picture as…." Regardless of the source of the image, you should be aware of who owns the copyright to the image you wish to use and what is required to reproduce the image legally.

WARNING *To avoid legal problems, always assume that an image on the Web is protected by copyright, even if there is no copyright notice shown. Just because you can easily download an image from a web site, doesn't mean that you can reuse that image in your own work without permission or paying a license fee. See Chapter 11 for more about copyright protection.*

Legal rights protecting use of images from clip libraries fall into three basic groupings. Public domain images were either never protected by a copyright or their copyright protection has ended. Generally these can be freely used without obtaining permission or paying a license fee, though there still may be an ownership issue for a particular work of art (such as a painting owned by an art gallery). Royalty-free images are purchased and then used without paying additional license fees. Rights-managed images require that you negotiate with the rights holder regarding terms for using the image and how much you will pay for that use.

Figure 3-3 shows a page of thumbnails describing a commercially available resource of royalty-free images called Photodisc, a part of Getty

Images (www.gettyimages.com). The Photodisc collections contain high-resolution bitmaps with a license for their "unlimited use." But you should note that "unlimited use" often contains caveats: in many cases there is an upper limit to the number of "units" of your own product that you may distribute without paying more, so you need to read the fine print. These additional fees are usually reasonable, however, and affect only commercial multimedia publishers.

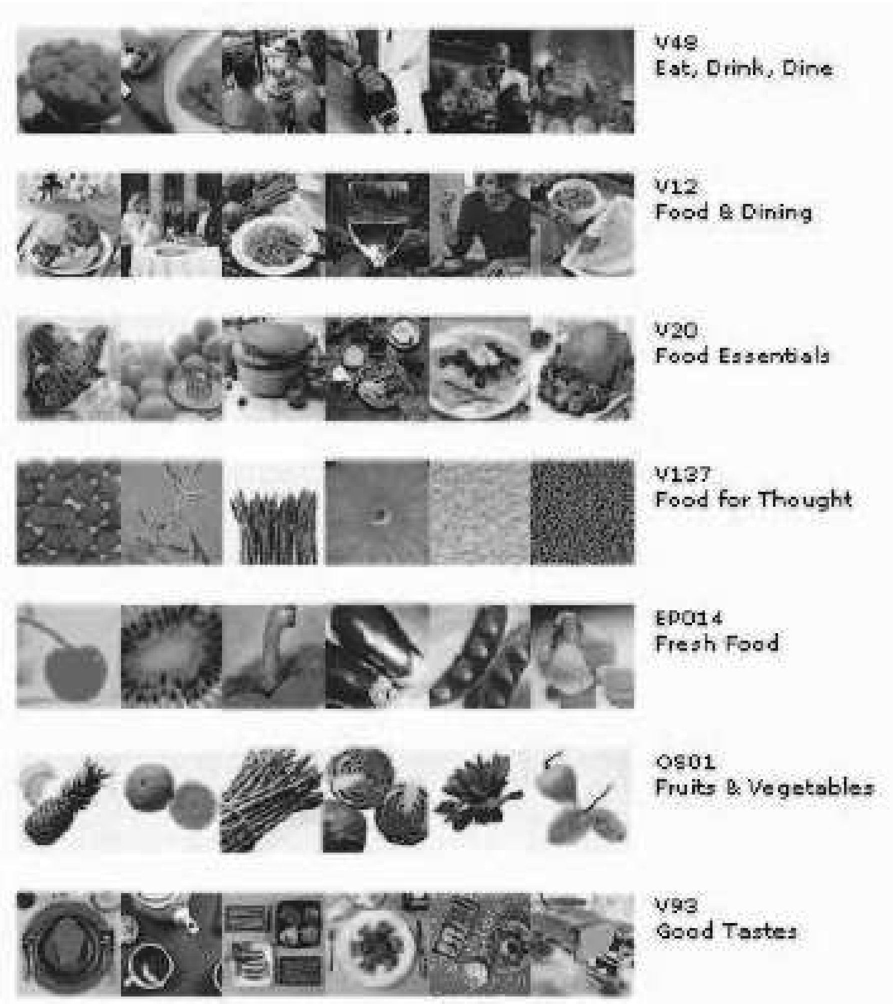


Figure 3-3 A page of thumbnails showing the content of various royalty-free Photodisc collections from Getty Images

Regardless of the source, once you have a bitmap, you can manipulate and adjust many of its properties (such as brightness, contrast, color depth, hue, and size). You can also cut and paste among many bitmaps using an image-editing program. If the clip art image is high resolution (aimed at 300 or 600 dpi printers, not 72 dpi monitors), you may discover that you can grab just a tiny portion of the high-res image—say, a sheep in the far corner of a farmyard or a car in a parking lot—and it will look great when displayed at monitor resolution.

Bitmap Software

The abilities and features of painting and image-editing programs range from simple to complex. The best programs are available in versions that work the same on both Windows and Mac platforms, and the graphics files you make can be saved in many formats, readable across platforms.

Macintosh computers do not ship with a painting tool, and Windows provides only a rudimentary Paint program, so you will need to acquire this very important software separately. Many multimedia authoring tools offer built-in bitmap editing features. Director, for example, includes a powerful image editor that provides advanced tools such as "onion-skinning" and image filtering using common plug-ins. Adobe's Photoshop, however, remains the most widely used image-editing tool among designers worldwide; it is available without some bells and whistles in a less-expensive version, Photoshop Elements, which may have all the features you need for your projects.

Many designers also use a vector-based drawing program such as Adobe's Illustrator, CorelDRAW, or InDesign to create curvy and complicated looks that they then convert to a bitmap. You can use your image-editing software to create original images, such as cartoons, symbols, buttons, bitmapped text, and abstract images that have a refined "graphic" look, but it is virtually impossible to create a realistic-looking photo from scratch using an image-editing program. The artistic painting tools offered by Corel's Painter (www.corel.com/painter) include hundreds of brushes, sprays, watercolors, inks, and textures to mimic the output of natural media in a bitmap (see Figure 3-4). There are also many open source and free bitmap editors available—just type "graphics editors" in a search engine. Regardless of your program of choice, learning to use a high-powered paint program and image editor is a necessary investment in your multimedia future.

Figure 3-4 Painter is used for creating original artwork; for book, medical, and architectural illustration; to transform photographs into realistic-looking paintings; to build seamless patterns for fabrics; and for story-boarding scene concepts and costumes for movies and theater.

Capturing and Editing Images The image you see on your monitor is a digital bitmap stored in video memory, updated about every 1/60 of a second. As you assemble images for your multimedia project, you may often need to capture and store an image directly from your screen. The simplest way to capture what you see on the screen at any given moment is to press the proper keys on your computer keyboard. This causes a conversion from the screen buffer to a format that you can use.

- Both the Macintosh and Windows environments have a **clipboard**—an area of memory where data such as text and images is temporarily stored when you cut or copy them within an application. In Windows, when you press PRINT SCREEN, a copy of your screen's image goes to the clipboard. From the clipboard, you can then paste the captured bitmap into an application (such as Paint, which comes with Windows).
- On the Macintosh, the keystroke combination COMMAND-SHIFT-3 creates a readable PNG-format file named Picture and places it on your desktop. You can then import this file's image into your multimedia authoring system or paint program. You can also press COMMAND-CONTROL-SHIFT-4 to drag a rectangle on your screen and capture what is inside the rectangle onto the clipboard, ready for pasting.

The way to get more creative power when manipulating bitmaps is to use an image-editing program, likely one of the programs named previously. These are the king-of-the-mountain programs that let you not only retouch the blemishes and details of photo images, but also do tricks like placing an image of your own face at the helm of a square-rigger or right on the sideline at last year's Super Bowl. Figure 3-5 shows just such a composite image, made from two photographs. It was created by graphic artist Frank Zurbano and shows his fiancée, Brandy Rowell, chasing after wedding gifts on the lawn where they will be married. Isolating and extracting parts of an image is an essential skill in multimedia production. Most bitmap editors have "lasso" type tools that select areas by drawing a path. This selection can be "feathered," or made to include partially transparent pixels outside the selected area.

In addition to letting you enhance and make composite images, image-editing tools allow you to alter and distort images. A color photograph of a red rose can be changed into a purple rose, or blue if you prefer. A small child standing next to her older brother can be "stretched" to tower over him. **Morphing** is another effect that can be used to manipulate still images or to create interesting and often bizarre animated

Figure 3-5 Image-editing programs let you add and delete elements in layers.

transformations. Morphing (see Figure 3-6) allows you to smoothly blend two images so that one image seems to melt into the next, often producing some amusing results.

Figure 3-6 Morphing software was used to seamlessly transform the images of 16 kindergartners. When a sound track of music and voices was added to the four-minute piece, it made a compelling video about how similar children are to each other.

Image-editing programs may, indeed, represent the single most significant advance in computer image processing during the late 1980s, bringing truly amazing power to PC desktops. Such tools are indispensable for excellent multimedia production.

> We have to keep saturation in mind *all* the time when doing our web pages… viewing the graphics on both Macs and PCs before actually using them. For instance, when doing our Halloween pages, we used a very cool pumpkin background that was beautifully saturated on the Mac side. On Windows, though, it was way too dark, and you couldn't read the overlying text. We had to lighten the GIF on the Mac side a few times before using it cross-platform.
>
> Rich Santalesa, Editor, *NetGuide* Magazine

NOTE *When you import a color or gray-scale bitmap from the Macintosh to Windows, the colors will seem darker and richer, even though they have precisely the same red, green, and blue (RGB) values. In some cases, this may improve the look of your image, but in other cases you will want to first lighten (increase the brightness and possibly lower the contrast) of the Macintosh bitmap before bringing it into Windows.*

Scanning Images After poring through countless clip art collections, you still haven't found the unusual background you want for a screen about gardening. Sometimes when you search for something too hard, you don't realize that it's right in front of you. Everyday objects can be scanned and manipulated using image-editing tools, such as those described in the preceding section, to create unusual, attention-getting effects. For example, to

enliven a screen with a gardening motif, scan a mixture of seeds, some fall foliage, or grass-stained garden gloves. Open the scan in an image-editing program and experiment with different filters, the contrast, and various special effects. Be creative, and don't be afraid to try strange combinations—sometimes mistakes yield the most intriguing results.

Another alternative to computer-generated graphics is to create artwork using traditional methods: watercolors, pastels, and even crayons. You can then scan the image, make necessary alterations, and tweak pixels on the computer. Too many designers have fallen into the trap of trying to draw detailed sketches using a mouse or drawing tablet, when a pencil or pen on paper would have produced better results quicker. In Chapter 10, Figure 10-7 shows a web page that uses a large image map of a seacoast village for navigation. The picture of the village was drawn on a large sheet of paper by artist Carolyn Brown using a fine pen. Then it was digitized in sections because the original drawing was too large for the scanner top. Four scans were stitched together into a single image using Photoshop layers, and the image was resized to fit the web page. Finally, it was colorized to look "old" and reduced in color depth to 4 bits so that it would load quickly on the Internet as a GIF.

Powerful filters and plug-ins are offered by most image-editing programs (see illustration to right) to manipulate bitmaps in many different ways. Experiment with your filters and plug-ins. Alien Skin's Exposure, for example, brings the creative tools of film photography to the world of digital editing with presets for many looks: discontinued films, darkroom tricks, lo-fi camera quirks like Holga and Lomo, vintage looks like Technicolor movie film and old Kodachrome that are distressed with dust, scratches, and lens blur, warped vignettes, and funky colors from cross-processing (see Figure 3-7).

Figure 3-7 Exposure from Alien Skin, offering photography effects, is one of hundreds of commercial plug-ins and filters available for manipulating bitmapped images. Here a digital color image has been processed to look like it came from a photographer's darkroom.

Vector Drawing

Most multimedia authoring systems provide for use of vector-drawn objects such as lines, rectangles, ovals, polygons, complex drawings created from those objects, and text.

- Computer-aided design (CAD) programs have traditionally used vector-drawn object systems for creating the highly complex and geometric renderings needed by architects and engineers.
- Graphic artists designing for print media use vector-drawn objects because the same mathematics that put a rectangle on your screen can also place that rectangle (or the fancy curves of a good line-art illustration) on paper without jaggies. This requires the higher resolution of the printer, using a page description format such as Portable Document Format (PDF).
- Programs for 3-D animation also use vector-drawn graphics. For example, the various changes of position, rotation, and shading of light required to spin an extruded corporate logo must be calculated mathematically. (Animation is discussed in Chapter 5.)

How Vector Drawing Works

A **vector** is a line that is described by the location of its two endpoints. Vector drawing uses **Cartesian coordinates** where a pair of numbers describes a point in two-dimensional space as the intersection of horizontal and vertical lines (the x and y axes). The numbers are always listed in the order x,y. In three-dimensional space, a third dimension—depth—is described by a z axis (x,y,z). This coordinate system is named for the French philosopher and mathematician, René Descartes. So a line might be simply

```
<line x1="0" y1="0" x2="200" y2="100">
```

where x1 and y1 define the starting point (in the upper-left corner of the viewing box) and x2 and y2 define the end point.

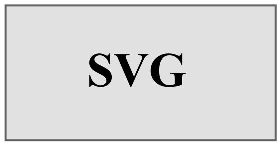
A simple rectangle is computed from starting point and size: your software will draw a rectangle (rect) starting at the upper-left corner of your viewing area (0,0) and going 200 pixels horizontally to the right and 100 pixels downward to mark the opposite corner. Add color information like

```
<rect x="0" y="0" width="200" height="100" fill="#FFFFFF" stroke="#FF0000"/>
```

and your software will draw the rectangle with a red boundary line and fill it with the color white. You can, of course, add other parameters to describe a fill pattern or the width of the boundary line. Circles are defined by a location and a radius:

<circle cx="50" cy="50" r="10" fill="none" stroke="#000000" />

Type the following code into a text editor and save it as plain text with a .svg extension. This is a **Scalable Vector Graphics** file. Open it in an HTML5-capable browser (File:Open File…) and you will see:




```
<svg xmlns="http://www.w3.org/2000/svg"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    width="200"
    height="200"
    viewBox="-100 -100 300 300">
<rect x="0" y="0" fill="yellow" stroke="red" width="200" height="100"/>
<text transform="matrix(1 0 0 1 60 60)" font-family="'TimesNewRomanPS-BoldMT'" font-size="36">SVG</text>
</svg>
```

Because these SVG files can be saved in a small amount of memory and because they are scalable without distortion (try changing the width and height of the view box in the preceding code), SVG (Tiny) is supported by browsers on most mobile phones and PDAs. The SVG specification also includes time-based changes or animations that can be embedded within the image code (see www.w3.org/TR/SVG11/animate.html#AnimationElements). Figure 3-8 shows Adobe Illustrator saving a file in SVG format. Vector drawing tools use **Bézier** curves or paths to mathematically represent a curve. In practical terms, editing software shows you points on the path, each point having a "handle." Changing the location of the handle changes the shape of the curve. Mastering Bézier curves is an important skill: these curves not only create graphic shapes but represent motion paths when creating animations.

Figure 3-8 Drawing software such as Adobe Illustrator can save vector graphics in SVG format.

Vector-Drawn Objects vs. Bitmaps

Vector-drawn objects are described and drawn to the computer screen using a fraction of the memory space required to describe and store the same object in bitmap form. The file containing the vector-drawn colored rectangle described in the preceding section is less than 698 bytes of alphanumeric data (even less—468 bytes—when the description is tokenized or compressed as .svgz). On the other hand, the same rectangle saved as a .gif image with a 64-color palette takes 1,100 bytes.

Because of this file size advantage, web pages that use vector graphics as SVG files or in plug-ins such as Flash download faster and, when used for animation, draw faster than pages displaying bitmaps. It is only when you draw many hundreds of objects on your screen that you may experience a slowdown while you wait for the screen to be refreshed—the size, location, and other properties for each of the objects must be computed. Thus, a single image made up of 500 individual line and rectangle objects, for example, may take longer for the computer to process and place on the screen than an image consisting of just a few drawn circle objects.

A vector-drawn object is created "on the fly," that is, the computer draws the image from the instructions it has been given, rather than displaying a precreated image. This means that vector objects are easily scalable without loss of resolution or image quality. A large drawn image can be shrunk to the size of a postage stamp, and while it may not look good on a computer monitor at 72 dpi, it may look great when printed at 300 dpi to a color printer. Resizing a bitmapped image requires either duplicating pixels (creating a blocky, jagged look called **pixelation**) or throwing pixels away (eliminating details). Because vector images are drawn from instructions on the fly, a rescaled image retains the quality of the original.

TIP *Using a single bitmap for a complicated image may give you faster screen-refresh performance than using a large number of vector-drawn objects to make that same screen.*

Converting Between Bitmaps and Drawn Images

Most drawing programs offer several file formats for saving your work, and, if you wish, you can convert a drawing that consists of several vector-drawn objects into a bitmap when you save the drawing. You can also grab a bitmapped screen image of your drawn objects with a screen capture program.

Converting bitmaps to drawn objects is more difficult. There are, however, programs and utilities that will compute the bounds of a bitmapped image or the shapes of colors within an image and then derive the polygon object that describes the image. This procedure is called **autotracing** and is available in vector drawing applications such as Illustrator or Freehand. Flash has a Trace Bitmap menu option that converts a bitmapped image into a vector image. Be cautious: the size of your Flash file may actually balloon because the bitmapped image is replaced by hundreds or even thousands of tiny vector-drawn objects, leading to slow processing and display.

WARNING *Some bitmap applications allow vector images to be pasted into them. Be careful to save your vector drawing separately because you will not be able to edit the curves when they are bitmapped.*

3-D Drawing and Rendering

Drawing in perspective or in 3-D on a two-dimensional surface takes special skill and talent. Creating objects in three dimensions on a computer screen can be difficult for designers comfortable with squares, circles, and other x (width) and y (height) geometries on a two-dimensional screen. Dedicated software is available to help you render three-dimensional scenes, complete with directional lighting and special effects, but be prepared for late nights and steep learning curves as you become familiar with nurbs, deformations, mesh generations, and skinning! From making 3-D text to creating detailed walkthroughs of 3-D space, each application will demand study and practice before you are efficient and comfortable with its feature set and power.

The production values of multimedia projects have increased dramatically, and as the production bar has risen, end users' expectations have also ratcheted upward. The multimedia production bar moves like a high jump or pole vault contest—as each new project improves on the last, competitors must jump to meet the new, higher standard. Flat and colorless 2-D screens are no longer sufficient for a successful commercial multimedia project. 3-D-rendered graphic art and animation has become commonplace since the late 1980s, providing more lifelike substance and feel to projects. Luckily, in an arena where only high-powered workstations could supply the raw computing horsepower for effective 3-D designing, inexpensive desktop PCs and excellent software have made 3-D modeling attainable by most multimedia developers.

Today many products—including Daz3D (www.daz3d.com) and form•Z (www.formz.com)—are touted as essential tools for illustration, animation, and multimedia production. NewTek's Lightwave (www.newtek.com/lightwave) and Autodesk's Maya (www.autodesk.com/Maya) are industry-standard, high-end animation programs used for everything from multimedia programs and game designs to special effects in films and even feature-length movies. For experimenting with 3-D, Google's SketchUp (sketchup.google.com) provides a simple (and free) cross-platform tool. To delve deeply into 3-D, the open-source Blender (www.blender.org) is a powerful tool—but its complex interface presents a steep learning curve.

Form•Z, the 3-D form synthesizer, is above all a 3-D modeling program, even though it also includes drafting, rendering and animation. Additional photo-realistic rendering is offered by form•Z RenderZone Plus. It combines solids and surface modeling. It also combines faceted (boundary) representations with parametric spline representations, NURBS, patches, and metaballs. This unique mixture of modeling personalities allows you to create any form, existing or imaginary, while working in a single package.

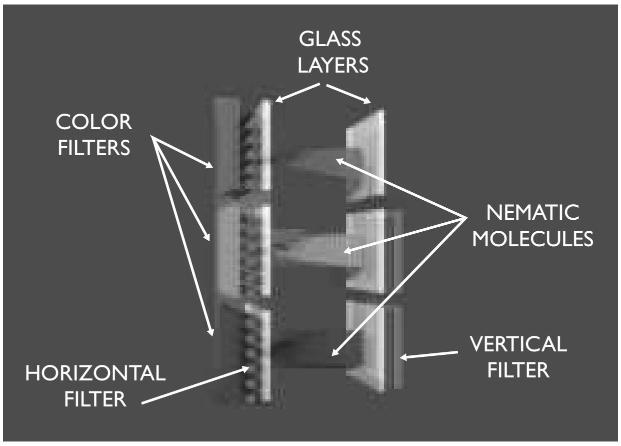
Marketing literature from auto•des•sys, Inc. *(www.formz.com)*

Computerized Color

Because the eye's receptors are sensitive to red, green, and blue light, by adjusting combinations of these three colors, the eye and brain will interpolate the combinations of colors in between. This is the psychology, not the physics, of color: what you perceive as orange on a computer monitor is a combination of two frequencies of green and red light, not the actual spectral frequency you see when you look at that namesake fruit, an orange, in sunlight. Various color models are illustrated in Figure 3-14. Although the eye perceives colors based upon red, green, and blue, there are actually two basic methods of making color: additive and subtractive.

Additive Color

In the **additive color** method, a color is created by combining colored light sources in three primary colors: red, green, and blue (**RGB**). This is the process used for cathode ray tube (CRT), liquid crystal (LCD), and plasma displays. On the back of the glass face of a CRT are thousands of phosphorescing chemical dots. These dots are each about 0.30mm or less in diameter (the **dot pitch**), and are positioned very carefully and very close together, arranged in triads of red, green, and blue. These dots are bombarded by electrons that "paint" the screen at high speeds (about 60 times a second). The red, green, and blue dots light up when hit by the electron beam. Your eye sees the combination of red, green, and blue light and interpolates it to create all other colors. Like CRTs, LCD and plasma screens utilize tiny red, green, and blue elements energized through tiny transparent conductors and organized in a Cartesian grid as illustrated by Marvin Raaijmakers and Angelo La Spina:



Subtractive Color

In the **subtractive color** method, color is created by combining colored media such as paints or ink that absorb (or subtract) some parts of the color spectrum of light and reflect the others back to the eye. Subtractive color is the process used to create color in printing. The printed page is made up of tiny halftone dots of three primary colors: cyan, magenta, and yellow (designated as CMY). Four-color printing includes black (which is technically not a color but, rather, the absence of color). Since the letter B is already used for blue, black is designated with a K (so four-color printing is designated as **CMYK**). The color remaining in the reflected part of the light that reaches your eye from the printed page is the color you perceive.

All these factors make computerized color pretty tricky to manage. The fact that a paint program uses RGB to create the colors on your monitor, while your printer uses CMYK to print out your image, explains the problem of matching what you see on the screen with your printout. High-end image-editing programs such as Photoshop deal with this problem by allowing you to calibrate your monitor with your printer.

The following chart shows the three primary additive colors and how, when one of the primary colors is subtracted from this RGB mix, the subtractive primary color is perceived. The numbers in parentheses indicate the amount of red, green, and blue (in that order) used to create each of the colors in 24-bit color, which is described in the next section. A zero indicates a lack of that primary color, while 255 is the maximum amount of that color.

RGB Combination (R,G,B)	Perceived Color
Red only (255,0,0)	Red
Green only (0,255,0)	Green
Blue only (0,0,255)	Blue
Red and green (blue subtracted) (255,255,0)	Yellow
Red and blue (green subtracted) (255,0,255)	Magenta
Green and blue (red subtracted) (0,255,255)	Cyan
Red, green, and blue (255,255,255)	White
None (0,0,0)	Black

Computer Color Models

Models or methodologies used to specify colors in computer terms are RGB, **HSB**, **HSL**, CMYK, **CIE**, and others. Using the 24-bit RGB (red, green, blue) model, you specify a color by setting each amount of red, green, and blue to a value in a range of 256 choices, from 0 to 255. Eight bits of memory are required to define those 256 possible choices, and that has to be done for each of the three primary colors; a total of 24 bits of memory (8 + 8 + 8 = 24) are therefore needed to describe the exact color, which is one of "millions" (256 × 256 × 256 = 16,777,216). When web browsers were first developed, the software engineers chose to represent the color amounts for each color channel in a hexadecimal pair. Rather than using one number between 0 and 255, two **hexadecimal** numbers, written in a scale of 16 numbers and letters in the range "0123456789ABCDEF" represent the required 8 bits (16 × 16 = 256) needed to specify the intensity of red, green, and blue. Thus, in HTML, you can specify pure green as #00FF00, where there is no red (first pair is #00), there is maximum green (second pair is #FF), and there is no blue (last pair is #00). The number sign (#) specifies the value as hexadecimal.

Red	Green	Blue	Color
255 (#FF)	255 (#FF)	255 (#FF)	White (#FFFFFF)
255 (#FF)	255 (#FF)	0 (#00)	Yellow (#FFFF00)
255 (#FF)	0 (#00)	255 (#FF)	Magenta (#FF00FF)
0 (#00)	255 (#FF)	255 (#FF)	Cyan (#00FFFF)
255 (#FF)	0 (#00)	0 (#00)	Red (#FF0000)
0 (#00)	255 (#FF)	0 (#00)	Green (#00FF00)
0 (#00)	0 (#00)	255 (#FF)	Blue (#0000FF)
0 (#00)	0 (#00)	0 (#00)	Black (#000000)

In the HSB (hue, saturation, brightness) and HSL (hue, saturation, lightness) models, you specify hue or color as an angle from 0 to 360 degrees on a color wheel, and saturation, brightness, and lightness as percentages. Saturation is the intensity of a color. At 100 percent saturation a color is pure; at 0 percent saturation, the color is white, black, or gray. Lightness or brightness is the percentage of black or white that is mixed with a color. A lightness of 100 percent will yield a white color; 0 percent is black; the pure color has a 50 percent lightness.

The CMYK color model is less applicable to multimedia production. It is used primarily in the printing trade where cyan, magenta, yellow, and black are used to print process color separations.

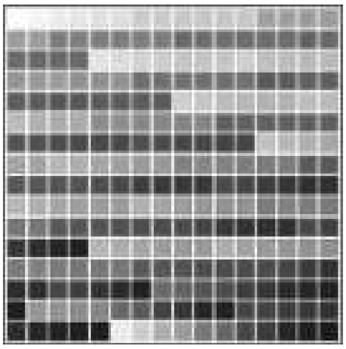
Color	Degrees
Red	0°
Yellow	60°
Green	120°
Cyan	180°
Blue	240°
Magenta	300°

There are many valid color names for HTML and CSS listed by The World Wide Web Consortium (W3C) that can be used in place of #Hex values: for example, aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow. Check out www.w3.org/TR/css3-color/#svg-color for a full list.

Other color models include CIE, **YIQ**, **YUV**, and **YCC**. CIE describes color values in terms of frequency, saturation, and illuminance (blue/yellow or red/green, which in turn corresponds to the color receptors in the cones of the eye). CIE more closely resembles how human beings perceive color, but certain devices such as scanners are unable to replicate the process.

YIQ and YUV were developed for broadcast TV (composite NTSC). They are based on luminance and chrominance expressed as the amplitude of a wave and the phase of the wave relative to some reference. Detail is carried by luminance (black and white), so reduction in color does not result in the loss of image definition detail. This analog process can be translated to a number value so that the computer can use a palette to assign a color to a pixel.

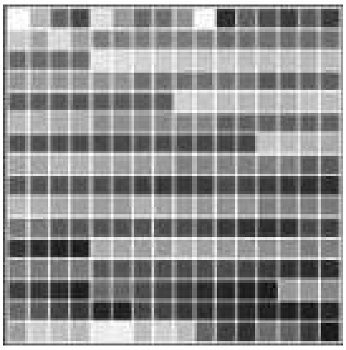
The Photo YCC model has been developed by Kodak to provide a definition that enables consistent representation of digital color images from negatives, slides, and other high-quality input. YCC is used for PhotoCD images.



Macintosh System

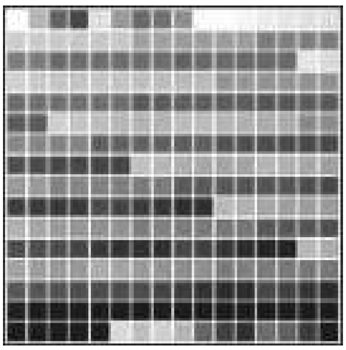
Color Palettes

Palettes are mathematical tables that define the color of a pixel displayed on the screen. The most common palettes are 1, 4, 8, 16, and 24 bits deep:



Windows System

Color Depth	Colors Available
1-bit	Black and white (or any two colors)
4-bit	16 colors
8-bit	256 colors (good enough for color images)
16-bit	Thousands of colors (65,536; excellent for color images)
24-bit	More than 16 million colors (16,777,216; totally photo-realistic)



Web-Safe

When color monitors became available for computers, managing the computations for displaying colors severely taxed the hardware and memory available at the time. 256-color, 8-bit images using a color lookup table or palette were the best a computer could do. 256 default system colors were statistically selected by Apple and Microsoft engineers (working independently) to be the colors and shades that are most “popular” in photographic images; their two system palettes are, of course, different. Web authorities also decided on a palette of 216 “web-safe” colors that would allow browsers to display images properly on both Macintosh and Windows computers.

GIF files using 256-color palettes are saved in a lossless format. The PNG format also uses palettes (24-bits or 32 bits if an "alpha" mask is included for transparency), and is lossless. It was developed for the Internet (it supports only the RGB color space) to expand GIF's limited 256 colors to millions of colors.

In 24-bit color systems, your computer works with three channels of 256 discrete shades of each color (red, green, and blue) represented as the three axes of a cube. This allows a total of 16,777,216 colors (256 × 256 × 256). Just as the 44.1 kHz sampled-sound standard for CD music on compact discs that is discussed in Chapter 4 covers the range of human hearing, the color range offered by 24-bit systems covers what the human eye can sense.

Dithering

If you start out with a 24-bit scanned image that contains millions of colors and need to reduce it to an 8-bit, 256-color image, you get the best replication of the original image by **dithering** the colors in the image. Dithering is a process whereby the color value of each pixel is changed to the closest matching color value in the target palette, using a mathematical algorithm. Often the adjacent pixels are also examined, and patterns of different colors are created in the more limited palette to best represent the original colors. Since there are now only 256 colors available to represent the thousands or even millions of colors in the original image, pixels using the 256 remaining colors are intermixed and the eye perceives a color not in the palette, created by blending the colors mixed together. Thus any given pixel might not be mapped to its closest palette entry, but instead to the average over some area of the image; this average will be closer to the correct color than a substitute color would be. How well the dithered image renders a good approximation of the original depends upon the algorithm used and whether you allow the image-editing program to select the best set of 256 colors from the original image (called an adaptive palette) or force it to use a predetermined set of 256 colors (as, for example, with a System palette or the browser-safe web palette). Figure 3-15 compares the same scanned image dithered from millions of colors to 256 colors, 16 colors, 16 grays, and black and white.

Dithering concepts are important to understand when you are working with bitmaps derived from RGB information or based upon different palettes. The palette for the image of a rose, for example, may contain mostly shades of red with a number of greens thrown in for the stem and leaves. The image of your pretty Delft vase, into which you want to electronically place the rose, may be mostly blues and grays. Your software will

> To generate a palette which is best for representing a particular image, we support Heckbert's median cut algorithm. This algorithm first builds a three-dimensional table (a histogram cube) indicating how popular any given colour in the RGB cube is in the image being converted. It then proceeds to subdivide this histogram cube (by dividing boxes in half) until it has created as many boxes as there are palette entries.
> The decision as to where to divide a box is based on the distribution of colours within the box. This algorithm attempts to create boxes which have approximately equal popularity in the image. Palette entries are then assigned to represent each box. There are other methods of generating a palette from an image, but Heckbert's algorithm is generally regarded as the best trade-off between speed and quality.
>
> Allan Hessenflow of HandMade Software, makers of Image Alchemy, describing how an 8-bit palette is made

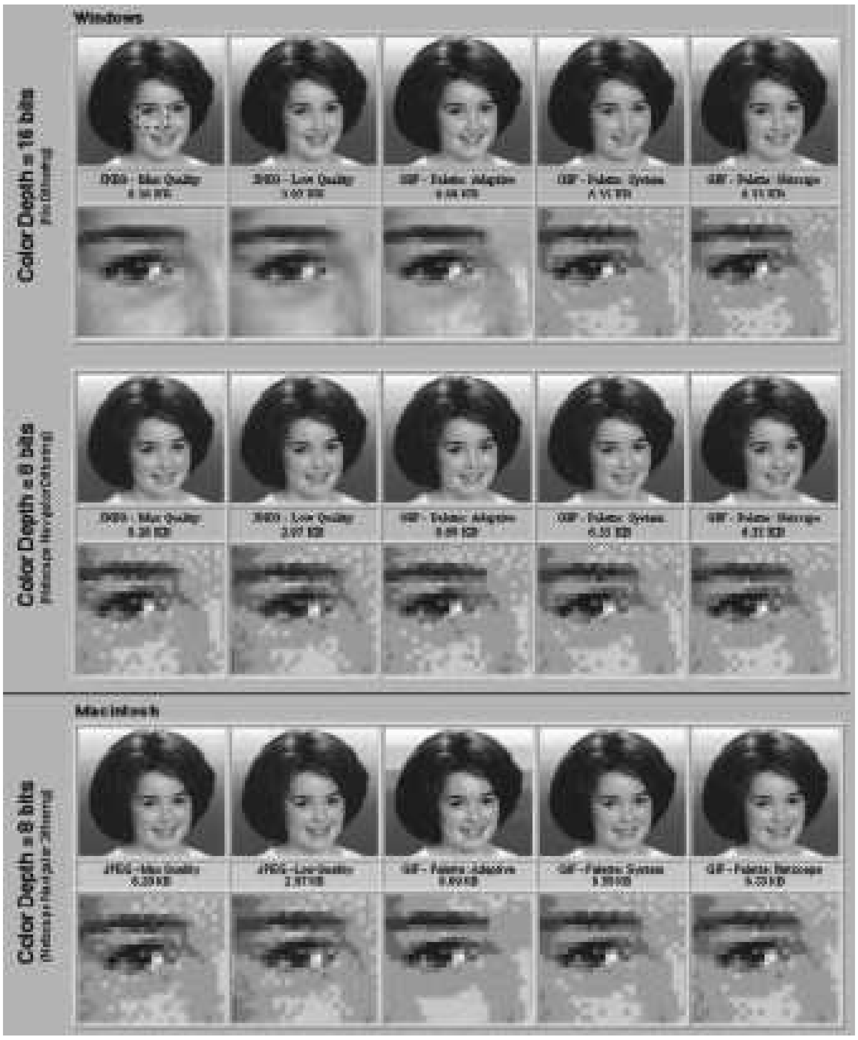


Figure 3-15 These images were dithered in Photoshop to best fit the 8-bit palettes of GIF files (Adaptive, System, or Custom 216 Netscape). Also shown are JPEG files compressed with highest and lowest quality and their actual file sizes. The files were then displayed using a browser at 16-bit and then 8-bit color depth. Note the subtle differences among palettes and systems, especially in the gradient blue background. Gradients do not usually dither well into 8-bit palettes.

> Multimedia is just another way to transform ambiguity. There were so many ambiguous colors in this scan, I decided to make them unambiguous. How do you like the purple?
>
> Lars Hidde, explaining why he dithered a perfectly fine 256-color image into a 16-color default palette

use a dithering algorithm to find the 256 color shades that best represent both images, generating a new palette in the process.

Dithering software is usually built into image-editing programs and is also available in many multimedia authoring systems as part of the application's palette management suite of tools.

Image File Formats

Most applications on any operating system can manage JPEG, GIF, PNG, and TIFF image formats. An older format used on the Macintosh, **PICT**, is a complicated but versatile format developed by Apple where both bitmaps and vector-drawn objects can live side by side. The **device-independent bitmap (DIB)**, also known as a **BMP**, is a common Windows palette–based image file format similar to PNG. **PCX** files were originally developed for use in Z-Soft MS-DOS paint packages; these files can be opened and saved by almost all MS-DOS paint software and desktop publishing software. **TIFF**, or Tagged Interchange File Format, was designed to be a universal bitmapped image format and is also used extensively in desktop publishing packages. Often, applications use a proprietary file format to store their images. Adobe creates a PSD file for Photoshop and an AI file for Illustrator; Corel creates a CDR file. **DXF** was developed by AutoDesk as an ASCII-based drawing interchange file for AutoCAD, but the format is used today by many computer-aided design applications. **IGS** (or **IGES**, for **Initial Graphics Exchange Standard**) was developed by an industry committee as a broader standard for transferring CAD drawings. These formats are also used in 3-D rendering and animation programs.

JPEG, PNG, and GIF images are the most common bitmap formats used on the Web and may be considered cross-platform, as all browsers will display them. Adobe's popular PDF (Portable Document File) file manages both bitmaps and drawn art (as well as text and other multimedia content), and is commonly used to deliver a "finished product" that contains multiple assets.

First Person

I needed to get about 40 bitmap files from the Macintosh to the Sun SPARCstation. "Piece of cake," I said. "Give me a few minutes." The network hadn't gone down in three days, and we were connected at broadband speeds. Well, the files had been saved in native Photoshop format on the Macintosh. So I launched Photoshop, opened each file, and then saved it in PICT format. The translator program I wanted to use to convert Macintosh PICT files to Sun raster files was an MS-DOS application, so I renamed all the Macintosh files to fit the DOS eight-plus-three-character filename convention. Then I cranked up the PC, launched the translator, and batch-processed all of the files into RAS files using the network. The 40 new files were now on the Macintosh, mixed in with the original PICTs. I collected the needed raster files into a single folder on the Macintosh and then sent the whole thing over to the Sun.

A few minutes? The process kept three chairs warm for about two hours.