CHAPTER 12

The Internet and Multimedia

In this chapter, you will learn how to:

- Discuss the origins of the Internet
- Define what a computer network is and how Internet domains, addresses, and interconnections work
- Discuss the current state of multimedia on the Internet and tools for the World Wide Web

THE material covered in this chapter is designed to give you an overview of the Internet while describing particular features that may be useful to you as a developer of multimedia for the World Wide Web. URLs and other pointers are also included here to lead you to information for obtaining, installing, and using these applications and utilities.

This chapter does *not* provide details about technology for connecting and using the Internet, about setting up servers and hosts, about installing and using applications, or what to do when you discover that you pressed the wrong key and have broadcast the intimate details of last night's hot date to 532 friends.

Embarrassing yourself on the stage of the civilized world can be avoided by education. Visit your local bookstore, where, along with the work you are now reading, you may discover as many as a hundred helpful volumes about all the simple and arcane aspects of the Internet. Buy one or two of these and dig in. Or, if you are already connected to the Internet, much of the documentation you may require can be found by surfing the Net itself. Use a search engine such as those listed here. Look particularly for documents called **FAQs (Frequently Asked Questions)**, because they contain answers.

AllTheWeb.com	*www.alltheweb.com*
AltaVista	*www.altavista.com*
AOL Search	*http://search.aol.com*
Ask	*www.ask.com*
Ask Jeeves	*www.askjeeves.com*
Bing	*www.bing.com*
Dogpile	*www.dogpile.com*
Gigablast	*www.gigablast.com*
Google	*www.google.com*
HotBot	*www.hotbot.com*

Lycos	*www.lycos.com*
Open Directory	*http://dmoz.org*
Yahoo	*www.yahoo.com*

Search engines on the World Wide Web

Internet History

The **Internet** began as a research network funded by the **Advanced Research Projects Agency (ARPA)** of the U.S. Defense Department, when the first node of the **ARPANET** was installed at the University of California at Los Angeles in September 1969. By the mid-1970s, the ARPANET "inter-network" embraced more than 30 universities, military sites, and government contractors, and its user base expanded to include the larger computer science research community. By 1983, the network still consisted of merely several hundred computers on only a few local area networks.

In 1985, the National Science Foundation (NSF) aligned with ARPA to support a collaboration of supercomputing centers and computer science researchers across the ARPANET. The NSF also funded a program for improving the backbone of the ARPANET, by increasing its bandwidth from 56 Kbps to T1 and then T3 (see "Connections" a little later in the chapter for more information) and branching out with links to international sites in Europe and the Far East.

In 1989, responsibility and management for the ARPANET was officially passed from military interests to the academically oriented NSF, and research organizations and universities (professors and students alike) became increasingly heavy users of this ever-growing " Internet." Much of the Internet's etiquette and rules for behavior (such as for sending e-mail and posting to newsgroups) was established during this time.

More and more private companies and organizations linked up to the Internet, and by the mid-1990s, the Internet included connections to more than 60 countries and more than 2 million host computers with more than 15 million users worldwide. Commercial and business use of the Internet was not permitted until 1992, but businesses have since become its driving force. By 2001 there were 109,574,429 domain hosts and 407.1 million users of the Internet, representing 6.71 percent of the world's population. By the beginning of 2010 (see Table 12-1), about one out of every four people around the world (26.6 percent) had access to the Internet, and more than 51 million domain names had been registered as "dot coms."

World Regions	Population (2009 Est.)	Internet Users Dec. 31, 2000	Internet Users Dec 31, 2009	Penetration (% Population)	Growth 2000–2009	Users% of Table
Africa	991,002,342	4,514,400	86,217,900	8.7%	1,809.8%	4.8%
Asia	3,808,070,503	114,304,000	764,435,900	20.1%	568.8%	42.4%
Europe	803,850,858	105,096,093	425,773,571	53.0%	305.1%	23.6%
Middle East	202,687,005	3,284,800	58,309,546	28.8%	1,675.1%	3.2%
North America	340,831,831	108,096,800	259,561,000	76.2%	140.1%	14.4%
Latin America/ Caribbean	586,662,468	18,068,919	186,922,050	31.9%	934.5%	10.4%
Oceania/ Australia	34,700,201	7,620,480	21,110,490	60.8%	177.0%	1.2%
WORLD TOTAL	6,767,805,208	360,985,492	1,802,330,457	26.6%	399.3%	100.0%

Table 12-1 World Internet Users and Population Stats (from www.internetworldstats.com)

First Person

When I was a kid, I took it for granted that you could see a million stars in the summer sky, and it wasn't until much later that I discovered the truth: only a paltry few thousand stars are actually visible to the naked eye from Earth. While "millions" is a perfect number for a ten-year-old's perception of an infinite universe, the term needs definition. For example, what does "133 million users on the Internet" really mean? The following exercise might help: Start counting to a million, incrementing by one every second: (One) (Two) (Three) . . . . In a minute, you will have counted to 60; in an hour, to 3,600. In 277.77 hours, you will reach a million—that's 11.57 24-hour days of nonstop counting, no pizza, no beer. You could try for the Guinness Book of Records, but you won't stay awake long enough!

Internetworking

In its simplest form, a **network** is a cluster of computers, with one computer acting as a **server** to provide network services such as file transfer, e-mail, and document printing to the **client** computers or users of that network. Using gateways and routers, a **local area network (LAN)** can be connected to other LANs to form a **wide area network (WAN)**. These LANs and WANs can also be connected to the Internet through a server that provides both the necessary software for the Internet and the physical data connection (usually a high-bandwidth telephone line, coaxial cable TV line, or wireless). Individual computers not permanently part of

a network (such as a home computer or a laptop) can connect to one of these Internet servers and, with proper identification and onboard client software, obtain an IP address on the Internet (see "IP Addresses and Data Packets" later in the chapter).

Internet Addresses

Let's say you get into a taxi at the train station in Trento, Italy, explain in English or Spanish or German or French that you wish to go to the Mozzi Hotel, and half an hour later you are let out of the car in a suburban wood—you have an address problem. You will quickly discover, as you return to the city in the back of a bricklayer's lorry to report your missing luggage and the cab driver, Mauro, who sped away in the rain, that you also have a serious language problem.

If you know how addresses work and understand the syntax or language of the Internet, you will likely not get lost and will save much time and expense during your adventures. You will also be able to employ shortcuts and workarounds.

Top-Level Domains

When the original ARPANET protocols for communicating among computers were remade into the current scheme of TCP/IP (Transmission Control Protocol/Internet Protocol) in 1983, the **Domain Name System (DNS)** was developed to rationally assign names and addresses to computers linked to the Internet. **Top-level domains (TLDs)** were established as categories to accommodate all users of the Internet:

com	Commercial entities
edu	Degree-granting colleges and universities (other schools register in the country domain)
gov	U.S. federal government agencies (state and local agencies register in the country domain)
int	Organizations established by international treaties and international databases
mil	U.S. military
net	Computers belonging to network providers
org	Miscellaneous and non-government organizations
Two-letter country codes	More than 240 countries and territories

In late 1998, the Internet Corporation for Assigned Names and Numbers (ICANN) was set up to oversee the technical coordination of the Domain Name System, which allows Internet addresses to be found

by easy-to-remember names instead of one of 4.3 billion individual IP numbers. In late 2000, ICANN approved seven additional TLDs:

aero	Air-transport industry
biz	Businesses
coop	Cooperatives
info	Unrestricted use
museum	Museums
name	For registration by individuals
pro	Accountants, lawyers, and physicians

Concerns about "rights" and "ownership" of domains are inappropriate. It is appropriate to be concerned about "responsibilities" and "service" to the community.

J. Postel, from the Network Working Group RFC 1591, March 1994

As a particular domain name is built up from the top-level domain, it consists of different levels separated by a period (spoken as "dot"). Since we read left to right, we tend to think first.second.third, left to right, but domain name levels are numbered right to left. Companies such as Microsoft, Apple, and IBM have second-level domain addresses that read microsoft.com, apple.com, and ibm.com—they are commercial (.com) operations with their second-level domain to the left of the top-level "com" domain. Government (.gov) agencies such as the Federal Bureau of Investigation, the Internal Revenue Service (a branch of the U.S. Treasury Department), and the White House have addresses that read fbi.gov, irs.ustreas.gov (note that the irs constitutes a third-level address), and whitehouse.gov.

Second-Level Domains

Many second-level domains contain huge numbers of computers and user accounts representing local, regional, and even international branches as well as various internal business and management functions. So the Internet addressing scheme provides for subdomains that can contain even more subdomains. Like a finely carved Russian matryoshka doll, individual workstations live at the epicenter of a cluster of domains.

Within the education (.edu) domain containing hundreds of universities and colleges, for example, is a second-level domain for Yale University called yale. At that university are many schools and departments (medicine, engineering, law, business, computer science, and so on), and each of these entities in turn has departments and possibly subdepartments and many users. These departments operate one or even several servers for managing traffic to

and from the many computers in their group and to the outside world. At Yale, the server for the Computing and Information Systems Department is named cis. It manages about 11,000 departmental accounts—so many accounts that a cluster of three subsidiary servers was installed to deal efficiently with the demand. These subsidiary servers are named minerva, morpheus, and mercury. Thus, minerva lives in the cis domain, which lives in the yale domain, which lives in the edu domain. Real people's computers are networked to minerva. Other real people are connected to the morpheus and mercury servers. To make things easy (exactly what computers are for), the mail system database at Yale maintains a master list of all of its people. So, as far as the outside world is concerned, a professor's e-mail address can be simply firstname.lastname@yale.edu; the database knows he or she is really connected to minerva so the mail is forwarded to that correct final address. In detailed e-mail headers, you may see the complete destination address listed as well as names of the computers through which your mail message may have been routed.

E-mail accounts are said to be "at" a domain (written with the @ sign). There are never any blank spaces in an Internet e-mail address, and while addresses on the Internet are normally case insensitive, conventional use dictates using all lowercase: the Internet will find tay@timestream.com, TAY@TIMESTREAM.COM, and Tay@Timestream.Com to be the same address.

The US Domain and Country Codes

The two-letter top-level US domain is based on political boundaries and is used by federal, state, and local government agencies, high schools, technical/vocational schools, private schools, elementary schools, libraries, fire and police departments, and regular citizens. Any computer in the United States can be in the US domain. Some fictitious examples are as follows:

fs.fed.us	Federal
senate.state.pa.us	State
assembly.state.ny.us	State
mwra.state.ma.us	State
ci.wayland.mi.us	City
co.alameda.ca.us	County
ccsf.cc.ca.us	Public community college
appleton.lib.me.us	Public library
pps.k12.or.us	Public school
perkins.pvt.k12.ma.us	Private school

NOTE *The Internet RFC 1480, http://rfc.net/rfc1480.html, describes the hierarchical rules for addresses in the US domain.*

Two-letter country codes, based on the International Organization for Standardization (ISO) document ISO-3166, are used in the addresses of all computers located outside the United States. Each country has an administrator who is responsible for organizing the naming hierarchy within that country's domain. Some countries use categories similar to com, edu, and org. Others base their naming hierarchies on political boundaries, as in the US country code.

schmidt@cage.rug.ac.be	Professor at University of Gent, Belgium
smythe@fiqus.unl.edu.ar	Student at L.C.S.A, Argentina
smith@iskratel.si	Commercial account, Slovenia
smith@laughs.co.uk	Commercial account, United Kingdom
smithe@idsc.gov.eg	Student at Cairo University, Egypt
smithy@udcf.gla.ac.uk	Researcher at University of Glasgow, Scotland
tsmith@library.usyd.edu.au	Scholar at University of Sydney, Australia

TIP *For a list of all the two-letter country codes, see: www.iana.org/cctld/cctld-whois.htm.*

IP Addresses and Data Packets

When a stream of data is sent over the Internet by your computer, it is first broken down into packets by the Transmission Control Protocol (TCP). Each packet includes the address of the receiving computer, a sequence number ("this is packet #5"), error correction information, and a small piece of your data. After a packet is created by TCP, the Internet Protocol (IP) then takes over and actually sends the packet to its destination along a route that may include many other computers acting as forwarders. **TCP/IP** is two important Internet protocols working in concert.

The 32-bit address included in a **data packet**, the **IP address**, is the "real" Internet address. It is made up of four numbers separated by periods, for example, 140.174.162.10. Some of these numbers are assigned by Internet authorities, and some may be dynamically assigned by an **Internet service provider (ISP)** when a computer logs on using a subscriber's account. There are domain name servers throughout the Internet whose sole job is to quickly look up text-based domain name addresses in large distributed databases, convert them into real IP addresses, and then return

them to you for insertion into your data packets. Every time you connect to http://www.google.com or send mail to president@whitehouse.gov, the domain name server is consulted and the destination address is converted to numbers.

TIP *IP addresses and domain names can be used interchangeably. Thus, kona .midcoast.com is the same Internet address as 69.39.100.10. There are occasional problems with the Internet's DNS servers, and by using the IP address, you may get connected immediately. With a Ping utility, or using the "whois" function in Unix, you can discover a domain's IP address.*

Connections

If your computer is connected to an existing network at an office or school, it is likely you are already connected to the Internet. If you are an individual working from home, you will need a telephone dial-up account or broadband cable, Digital Subscriber Line (DSL), or wireless equipment to connect to the **backbone** (the ultra-high-bandwidth underlying network operated by MCI, AT&T, Sprint, and other telecommunications companies) of the Internet through an Internet service provider (ISP).

The Bandwidth Bottleneck

Bandwidth is how much data, expressed in bits per second (bps), you can send from one computer to another in a given amount of time. The faster your transmissions (or the greater the bandwidth of your connection), the less time you will spend waiting for text, images, sounds, and animated illustrations to upload or download from computer to computer, and the more satisfaction you will have with your Internet experience. To think in bytes per second, divide the rate by eight. Table 12-2 lists the bandwidth of some common data transfer methods.

Type of Connection	Bandwidth (in bits per second) Without Compression	Comment
56K modem	56,000	Maximum analog modem speed for copper wires, (Dial-Up) data compressed using V91 standard. Actual is about 48 Kbps.
ISDN	56,000 to 128,000	Integrated Services Digital Network basic services (128,000 bps if no voice mixed in).
Frame relay	56,000 to 45,000,000	Dedicated service offered by long-distance phone companies.

Table 12-2 Bandwidth of Typical Internet and Computer Connections (For more information visit www.cis.eku.edu/loy/cis300/bandwidth.html.)

Type of Connection	Bandwidth (in bits per second) Without Compression	Comment
Ethernet-10	10,000,000	Networking hardware and protocol, commonly uses two twisted pairs of copper wire.
T-1 (DS-1 in North America)	1,544,000	Equal to 24 leased lines at 56 Kbps.
E-1 (DS-1 in Europe)	2,000,000	European equivalent of a T-1 connection.
DSL	1,500,000 to 9,000,000	Digital Subscriber Line service available in various technologies (HDSL, SDSL, ADSL, VDSL, and RDSL) with differing data rates, operating distances, and ratios between downstream and upstream speeds.
Cable Modem	3,000,000 upload; 7,000,000 download	Even though copper coaxial TV cable can be used in a bidirectional fashion, it was originally designed to carry limited signals in one direction.
Wireless (802.11)	3,000,000 to 54,000,000	Radio connection in the radio frequency (RF) bands of 2.4 GHz (WiFi) or 5.8 GHz.
T-3 (D-3 in North America)	45,000,000	Typical backbone speed of major ISPs in the United States (1996).
Fast Ethernet-100	100,000,000	Networking hardware and protocol, commonly uses two twisted pairs of copper wire.
OC-3	155,000,000	Upgrade for ISPs in the United States (1997).
Gigabit Ethernet	1,000,000,000	Used for local network backbones; standard in many computers.
OC-48	2,400,000,000 (2.4 gigabits per second)	Typical speed for intercity fiber-optic lines (called SONET or Synchronous Optical Network).
10 Gigabit Ethernet	10,000,000,000 (10 gigabits per second)	Used for local network backbones.
OC-255	13,210,000,000 (13.21 gigabits per second)	Really fast fiber-optic lines using SONET.

Table 12-2 Bandwidth of Typical Internet and Computer Connections (For more information visit www.cis.eku.edu/loy/cis300/bandwidth.html.) *(Continued)*

The bottleneck at a typical user's low-bandwidth modem connection is the most serious impediment to sending multimedia across the Internet. At low bandwidth, a page of text (3,000 bytes) can take less than a second to send, but an uncompressed 640 × 480, 8-bit/256-color image (about 300,000 bytes) can take a few minutes; an uncompressed 640 × 480, 24-bit/16 million-color image (about 900,000 bytes) can take many minutes to send. Occasionally also, even though you may have a high-speed connection, the server delivering your requested file or content may be "throttled down" while it manages many requests at once, and yours must wait its turn.

To work within the constraints of bandwidth bottlenecks, multimedia developers on the Internet have but a few options:

- Compress data as tightly as possible (into ZIP or SIT or TAR files) before transmitting.
- Require users to download data only once; then store the data in a local hard disk cache (this is automatically managed by most browsers).
- Design each multimedia element to be efficiently compact—don't use a greater color depth than is absolutely necessary or leave extra space around the edges.
- Design alternate low-bandwidth and high-bandwidth navigation paths to accommodate all users.
- Implement streaming methods that allow data to be transferred and displayed incrementally as it comes in (without waiting for the entire data file to arrive).

Internet Services

To many users, the Internet means the World Wide Web. But the Web is only the latest and most popular of services available today on the Internet. E-mail; file transfer; discussion groups and newsgroups; real-time chatting by text, voice, and video; and the ability to log into remote computers are common as well. Internet services are shown here.

Service	Purpose
ftp	For transferring files between computers; can be anonymous or password protected (from File Transfer Protocol)
gopher	For menus of material available on the Internet (seldom used)
http	For posting and reading documents (from the Hypertext Transfer Protocol used by the World Wide Web)
https	For posting and reading encrypted (secure) documents
imap	For receiving electronic mail (from Internet Message Access Protocol)
irc	For real-time text messaging (from Internet Relay Chat)
mud	For real-time game playing (from MultiUser Dimension)
pop	For receiving electronic mail (from Post Office Protocol)
rtsp	For streaming media control (from Real Time Streaming Protocol)
telnet	For logging on and working from remote computers
smtp	For sending mail (Simple Mail Transport Protocol)
usenet	For participating in discussion groups (from USErs NETwork)

Each Internet service is implemented on an Internet server by dedicated software known as a **daemon**. (Actually, daemons only exist on Unix/Linux systems—on other systems, such as Windows, the services may run as regular applications or background processes.) Daemons are

agent programs that run in the background, waiting to act on requests from the outside. In the case of the Internet, daemons support protocols such as the **Hypertext Transfer Protocol (HTTP)** for the World Wide Web, the **Post Office Protocol (POP)** for e-mail, or the **File Transfer Protocol (FTP)** for exchanging files. You have probably noticed that the first few letters of a **Uniform Resource Locator (URL)**—for example, http://www.timestream.com/index.html—notify a server as to which daemon to bring into play to satisfy a request. In many cases, the daemons for the Web, mail, news, and FTP may run on completely different servers, each isolated by a security firewall from other servers on a network.

FTP.ARL.MIL (Army Research Laboratory) Bandwidth Information

ARL has multiple high-speed connections to several significant networks, providing excellent performance for most document transfers:

- **DREN** This site is a primary node in the Defense Research and Engineering Network (DREN), which sports a variety of OC-12, OC-3, T-3, and T-1 communications links to other U.S. Government facilities.
- **The Internet Backbone** This site is connected to the Internet "backbone" at strategic locations via a "cloud" of OC-12 ATM paths provisioned over AT&T's nationwide DISC ATM network: MAE-East in Washington, DC; FIX-West and MAE-West in San Francisco. The Sprint NAP in Pensauken, NJ. The NAP in Chicago, IL. The "Giga-pop" in Washington state.
  NIPRNET (nee MILNET) This site is gatewayed to a NIPRNET military Packet Switching Node that has multiple T-1 trunks.

  Moving 1 MByte of data over an OC-3 link takes about 0.1 seconds, if your end is up to it.

  Moving 1 MByte of data over a T-3 link takes about 1 second, if your end is up to it.

  Moving 1 MByte of data over a T-1 link takes about 8 seconds.

  Moving 1 MByte of data over a 56 Kbps link takes about 3 minutes.

  Moving 1 MByte of data over a 28.8 Kbps modem takes about 5 minutes.

  Naturally, no single file transfer ever gets the full bandwidth of these communications lines, as they are a shared resource. These figures should help you make a lower-bound estimate on how much time large file transfers might take.

  You are free to transfer large files from this site at any time. Information seekers from all domains are welcome to view this data. It is important that our guests understand that this is an official U.S. Government System for unclassified use only. Use of this system constitutes consent to security testing and monitoring.

  webmaster@arl.army.mil

MIME-Types

To work with multimedia on the Internet, you must work within the requirements of the appropriate protocol, using recognizable documents and formats. A voice attachment to an e-mail message, for example, must be identified by the Post Office daemon for what it is, and then be transmitted with the correct coding to the receiving computer. The receiver must have the proper software (and hardware) for decoding the information and playing it back. To identify the nature of the data transmitted and, by inference, the purpose of that data, the Internet uses a standard list of filename extensions called **Multipurpose Internet Mail Extensions (MIME-types)**. Most browsers allow you to define MIME-types and map "helper apps" to the type for decoding and playing. For example, with Netscape Navigator you can define Adobe's Acrobat files (PDF files) as a MIME-type and select the Acrobat Reader as the player application.

These are not just used by the e-mail daemon but, by convention, by other Internet daemons, including the Web's HTTP daemon. Perhaps the most widely installed HTTP software for managing web pages is the open-source application called Apache (www.apache.org). Table 12-3 shows a list of common MIME-types and their uses. (Note that many come from the Unix world, where the Internet was born.) You can also visit www.file-ext.com for more information.

Extension	Type	Use
ai	application/postscript	PostScript program
aif	audio/x-aiff	Audio
aifc	audio/x-aiff	Audio
AIFF	audio/x-aiff	Audio
aiff	audio/x-aiff	Audio
au	audio/basic	ULAW audio data
avi	video/x-msvideo	Microsoft video
bin	application/octet-stream	Binary executable
cpio	application/x-cpio	Unix CPIO archive
csh	application/x-csh	C shell program
dcr	application/director	Shockwave animation
dvi	application/x-dvi	TeX DVI data
eps	application/postscript	PostScript program
exe	application/octet-stream	Binary executable

Table 12-3 Some Common MIME-Types Illustrate the Variety of Data Types and Formats Used on the Internet

Extension	Type	Use
fif	application/fractals	Fractal image format
flv	video/x-flv	Flash video
gif	image/gif	CompuServe image format
gtar	application/x-gtar	GNU tape archive
gz	encoding/x-gzip	GNU zip compressed data
hqx	application/mac-binhex40	Macintosh BinHex archive
htm	text/html	Hypertext Markup Language
html	text/html	Hypertext Markup Language
ief	image/ief	Image
jpe	image/jpeg	JPEG image
jpeg	image/jpeg	JPEG image
jpg	image/jpeg	JPEG image
latex	application/x-latex	LaTeX document
kmz	application/vnd.google-earth.kmz	Google Earth document
man	application/x-troff-man	Unix manual page
me	application/x-troff-me	TROFF document
mov	video/quicktime	QuickTime video
movie	video/x-sgi-movie	SGI video
mpe	video/mpeg	MPEG video
mpeg	video/mpeg	MPEG video
mpg	video/mpeg	MPEG video
ms	application/x-troff-ms	TROFF document
pbm	image/x-portable-bitmap	PBM image
pgm	image/x-portable-graymap	PGM image
pnm	image/x-portable-anymap	PBM image
ppm	image/x-portable-pixmap	PPM image
ps	application/postscript	PostScript program
qt	video/quicktime	QuickTime video
ra	audio/x-pn-realaudio	RealAudio sound
ram	audio/x-pn-realaudio	RealAudio sound
ras	image/x-cmu-raster	CMU raster image
rgb	image/x-rgb	RGB image
roff	application/x-troff	TROFF document
rtf	application/rtf	Rich Text Format

Table 12-3 Some Common MIME-Types Illustrate the Variety of Data Types and Formats Used on the Internet *(Continued)*

Extension	Type	Use
sh	application/x-sh	Bourne shell program
shar	application/x-shar	Unix shell archive
sit	application/x-stuffit	Macintosh archive
snd	audio/basic	ULAW audio data
swf	application/ x-shockwave-flash	Flash document
t	application/x-troff	TROFF document
tar	application/x-tar	Unix tape archive
tcl	application/x-tcl	TCL program
tex	application/x-tex	TeX document
texi	application/x-texinfo	GNU TeXinfo document
texinfo	application/x-texinfo	GNU TeXinfo document
text	text/plain	Plain text
tif	image/tiff	TIFF image
tiff	image/tiff	TIFF image
tr	application/x-troff	TROFF document
txt	text/plain	Plain text
vox	audio	VoxWare
wav	audio/x-wav	WAV audio
xbm	image/x-xbitmap	X bitmap
xpm	image/x-xpixmap	X pixmap
xwd	image/x-xwindowdump	X Window dump image
z	encoding/x-compress	Compressed data
zip	application/x-zip-compressed	Zip compressed data

Table 12-3 Some Common MIME-Types Illustrate the Variety of Data Types and Formats Used on the Internet *(Continued)*

Multimedia elements are typically saved and transmitted on the Internet in the appropriate MIME-type format and are named with the proper extension for that type. For example, Shockwave Flash animation files end in .swf; image files end in .jpg, .jpeg, .gif, or .png; sound files end in .au, .wav, .aif, .mp3, or another conforming format; QuickTime and other video clips end in .qt, .mov, mp4, or avi.

WARNING *Because some MIME-types for multimedia data are new or experimental, not all servers may recognize them. If you have problems with a multimedia file, check with your Internet service provider to be sure your server can serve "experimental" MIME-types. Some ISPs will not install the requisite, and often costly, server software for high-bandwidth streaming MIME-types.*

The Semantic Web is an extension of the current Web in which information is given well-defined meaning, enabling computers and people to work in better cooperation. The W3C Semantic Web Activity, in collaboration with a large number of researchers and industrial partners, is tasked with defining standards and technologies that allow data on the Web to be defined and linked in a way that it can be used for more effective discovery, automation, integration, and reuse across applications. The Web will reach its full potential when it becomes an environment where data can be shared and processed by automated tools as well as by people.

Tim Berners-Lee
and Eric Miller
from *The Semantic Web Lifts Off*

The World Wide Web and HTML

The World Wide Web (www.w3.org/) started in 1989 at the European Particle Physics Laboratory (CERN) as a "distributed collaborative hypermedia information system." It was designed by Tim Berners-Lee as a protocol for linking a multiplicity of documents located on computers anywhere within the Internet. This new Hypertext Transfer Protocol (HTTP) provided rules for a simple transaction between two computers on the Internet consisting of (1) establishing a connection, (2) requesting that a document be sent, (3) sending the document, and (4) closing the connection. It also required a simple document format called Hypertext Markup Language (HTML) for presenting structured text mixed with inline images.

An HTML document could contain hyperlinks or anchors that referred to other similar documents. With browser software, users could then click on designated areas of hot text in one document and jump to another, which itself might have more hot text pointing to yet other documents. Users could surf from document to document across the Web, with HTML as the underlying buoyant framework. Berners-Lee is currently developing the next evolution, the **Semantic Web**, which "provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries." Visit www.w3.org/2001/sw/ for more.

Dynamic Web Pages and XML

HTML is fine for building and delivering uncomplicated *static* web pages. But you will need other tools and programming know-how to deliver *dynamic* pages that are built on the fly from text, graphics, animations, and information contained in databases or documents. JavaScript and programs written in Java may be inserted into HTML pages to perform special functions and tasks that go beyond the vanilla abilities of HTML—for mouse rollovers, window control, and custom animations.

Cold Fusion and PHP are applications running side by side with a web server like Apache; they scan an outgoing web page for special commands and directives, usually embedded in special tags. If they find a special tag in the page, the software will do what the tag tells it to do, like "get today's date and put it into that table cell" or "search this database for all customers with balances greater than $100 and, after alphabetizing, put that list into a table on the web page being served." Working hand-in-hand with these application servers, Oracle, Sybase, and mySQL offer software to manage **Structured Query Language (SQL)** databases that may contain not only text but also graphics and multimedia resources like sounds and video clips. In concert with HTML, these tools provide

the power to do real work and perform real tasks within the context of the World Wide Web.

Flash animations, Director applications, and RunRev stacks can also be called from within HTML pages. These multimedia mini-applications, often programmed by Web developers, use a browser plug-in to display the action and perform tasks such as playing a sound, showing a video, or calculating a date. As with Cold Fusion and PHP, both use underlying programming languages. With the introduction of HTML5, browsers can play multimedia elements such as sound, animations, and video without requiring special plug-ins or software.

XML (Extensible Markup Language) goes beyond HTML—it is the next evolutionary step in the development of the Internet for formatting and delivering web pages using styles. Unlike HTML, you can create your own tags in XML to describe exactly what the data means, and you can get that data from anywhere on the Web. In XML, you can build a set of tags like

```
<fruit>
<type>Tomato</type>
<source>California</source>
<price>$.64</price>
</fruit>
```

and your XML document, according to your instructions, will find the information to put into the proper place on the web page in the formatting style you assign. For example, with XML styles, you can declare that all items within the <price> tag will be displayed in boldface Helvetica type.

TIP *For more information about XML see:*
www.xml.org
www.xml.com

In development as a technique to deliver more pleasing web experiences, AJAX (Asynchronous JavaScript and XML) uses a combination of XML, **CSS (Cascading Style Sheets)** for marking up and styling information), and JavaScript to generate dynamic displays and allow user interaction within a web browser.

Flash was created during the PC era—for PCs and mice. Flash is a successful business for Adobe, and we can understand why they want to push it beyond PCs. But the mobile era is about low power devices, touch interfaces and open web standards—all areas where Flash falls short.
The avalanche of media outlets offering their content for Apple's mobile devices demonstrates that Flash is no longer necessary to watch video or consume any kind of web content. And the 200,000 apps on Apple's App Store proves that Flash isn't necessary for tens of thousands of developers to create graphically rich applications, including games.
New open standards created in the mobile era, such as HTML5, will win on mobile devices (and PCs too). Perhaps Adobe should focus more on creating great HTML5 tools for the future, and less on criticizing Apple for leaving the past behind.

Steve Jobs,
CEO Apple Computer, Inc.,
April, 2010

The Web is becoming much more than a static library. Increasingly, users are accessing the Web for "web pages" that aren't actually on the shelves. Instead, the pages are generated dynamically from information available to the web server. That information can come from databases on the web server, from the site owner's enterprise databases, or even from other web sites.

Charles Goldfarb,
who invented SGML
(the parent language of HTML and XML)
and coined the term "markup language"

Multimedia on the Web

During the coming years, most multimedia experiences on the Internet will occur on the World Wide Web, programmed within the constraints of HTML, then stretched by the enhanced capabilities provided by XML, Java, JavaScript, AJAX, and special plug-ins like Flash and QuickTime to enable browsers to exceed their limits. These tools are used to build "**Web 2.0**" sites where there is collaboration and information sharing such as seen in blogs, on wikis, and at social networking sites such as Facebook and Twitter.

To design and make effective multimedia for this environment, developers need to understand not only how to create and edit the elements of multimedia, but also how to deliver it for HTML browsers and plug-in/player vehicles. Well-crafted, professionally rendered sites on the Web include text, images, audio, and animation presented in a user-friendly interface that balances the bandwidth deficit against user patience.

Inside the event horizon of the amazing World Wide Web explosion are many uncertainties and unsolved challenges. The bandwidth deficit will certainly be met with technology solutions that will reach the last mile into homes and businesses. There is a terrific need for high-quality, compelling content; multimedia developers and entrepreneurs will fill this creative void.

Tools for the World Wide Web

In the late 1990s, multimedia plug-ins and commercial tools aimed at the Web entered the marketplace at a furious pace, each competing for visibility and developer/user mind share in an increasingly noisy venue. In the few years since the birth of the first line-driven HTTP daemon in Switzerland, millions of web surfers had become hungry for "cool" enhancements to entertaining sites. Web site and page developers needed creative tools to feed the surfers, while surfers needed browsers and the plug-ins and players to make these cool multimedia enhancements work.

A combination of the explosion of these tools and user demand for performance stresses the orderly development of the core HTML standard. Unable to evolve fast enough to satisfy the demand for features (there are committees, international meetings, rational debates, comment periods, and votes in the standards process), the HTML language is constantly being extended de facto by commercial interests. These companies regularly release new versions of web browsers containing **tags** (HTML formatting elements) and features not yet formally approved. By the time (measured in weeks!) millions of users have become dependent upon the features of the new browser versions, the more carefully considered official specification has no choice but to incorporate them. By the time features are "official," of course—after more meetings, votes, and understated

demonstrations of power—still newer browser versions have been released with yet newer, unofficial features.

What keeps this cycle from being chaotic are the natural selection forces of the marketplace: developers strive toward a successful product that works better and satisfies more users without mutating so far from the core standard that there are no sales and the company collapses. Developers also complain about the contention among browser vendors because they must program workarounds that compensate for the performance differences among them, and they must test the performance of their site on all or as many as possible.

Browsers provide a method for third-party developers to "plug in" special tools that take over certain computational and display activities. They also support the **Java** and **JavaScript** languages by which programmers can create bits of programming script and Java **applets** to extend and customize a browser's basic HTML capabilities, especially into the multimedia realm. Java and JavaScript are only related by name. Java is a programming language much like C++ that must be compiled into machine code to be executed by a computer's operating system. JavaScript is a "scripting language" whose commands are executed at runtime by the browser itself. JavaScript code can be placed directly into HTML using <script> tags or referenced from a file with the ".js" extension.

Thus, while browsers provide the orchestrated foundation of HTML, third-party players and even nonprogrammers can create their own cadenzas to enhance browser performance or perform special tasks. It is often through these plug-ins and applets that multimedia reaches end users. Many of these tools are available as freeware and shareware while others, particularly server software packages, are expensive, though most any tool can be downloaded from the Internet in a trial version. Try it. If you like it or use it, buy it.

The stunning growth of the Internet as well as expansion of wireless mobile phone connectivity to the Internet has caused many multimedia developers to redirect their creative efforts toward providing software solutions for these arenas. This remains a new and lucrative frontier, and no developer wishes to be left behind.

Web Servers

The workings of the Web involve communication between two computers: a server and a client. The server delivers a file when a client asks for it. Because the playback or display performance of your multimedia content—particularly when it is a streaming MIME-type such as RealAudio or Shockwave/Flash or a QuickTime video—depends upon the speed and capabilities of the computer and software serving it (as well as the bandwidth and load factors of the Internet), you should know some basics.

A growing number of software vendors provide web servers of varying strength and capacity and for a variety of platforms, all of which meet the requirements of the Hypertext Transfer Protocol. A server is technically not the hardware, but the software—you should invest in server software that will stand up to your intended use and be supported by the vendor. Most vendors will also recommend hardware configurations. This combination of software and hardware is critical to your success and happiness if you wish to optimize response time (less than a second), your connections per second (as many as possible), and your throughput (plenty of room before your Internet connection is overwhelmed by traveling packets).

WARNING *If you do not develop growth predictions based upon sound business practices and install adequate server performance and load balancing, you may discover that those 6,000 hits you received in the hour after you and your brother held up the bedsheet boasting your company's URL during the Super Bowl are being served at 87 bytes per second or refused altogether. At least be sure your choice of server and its connection to the Internet backbone provide a sensible migration path for growth; people don't usually come back to an unsatisfying experience.*

Web Browsers

Your computer's performance is as important as the bandwidth of your connection to the Web. Web **browsers** are applications that run on a user's personal computer (on the client side on the Internet) to provide the interactive graphical interface for searching, finding, and viewing text documents, sounds, animations, and other multimedia resources on the Web. In 1996, as many as 50 browsers competed for market share, each boasting special or unique features, performance, and cost. Rich Santalesa, editor of *NetGuide* magazine, predicted even then that "the browser wars are over—it's a battle between Microsoft and Netscape, and everyone else is going to dry up and blow away." Indeed, by mid-2001, only two serious competitors remained: Netscape and Microsoft, and Netscape, despite more than 40 million registered users, was beginning a chameleon act. Purchased by AOL, then alloyed by a merger with Time Warner, Netscape was repositioned as a "media hub," not a software company, giving the new Netscape a chance to sell advertising across its many media properties and experiment with subscriptions rather than just free services within the AOL-Time Warner media empire (which includes properties such as *Fortune* and *Time* magazines and the 24-hour cable news network CNN). By 2006, Netscape was dead. From Netscape's ashes arose **Mozilla Firefox** as an open-source competitor to **Microsoft Internet Explorer**.

Despite the legal and financial seriousness of this competition, manifesting in very real congressional hearings and complicated multimillion-dollar antimonopoly lawsuits, some of those involved kept their sense of humor. Back in October 1997, late in the night after the gala announcement and rollout of Microsoft's new Explorer 4.0 in San Francisco, a group of Microsoft engineers drove 30 miles south to Netscape's headquarters and placed a truck-sized Explorer logo (the world-circling "e") on the front lawn of the competitor's headquarters, accented with a helium balloon saying "We Love You" and a greeting card with the message, "It's just not fair. Good people shouldn't have to feel bad. Best wishes, the IE team." By midmorning, Netscape's own engineers had crowned the Explorer logo with a giant dinosaur (their company mascot, named Mozilla), and nailed up a cardboard sign declaring, "Netscape 72, Microsoft 18" (the companies' market share at that time). Mozilla later spun out of Netscape in a free, open-source effort to standardize the browser's HTML engine.

Today, the majority of visitors to your web site will be using Microsoft's Internet Explorer, winner of the "browser wars" (see Table 12-4). In designing a web site, then, you should be certain that your documents and plug-ins work and look good using Internet Explorer.

Internet Explorer	Firefox	Chrome	Safari	Opera	Unknown
47%	31%	7%	5%	1%	9%

Table 12-4 Browser Market Penetration in 2010 (From www.w3counter.com)

Search Engines

You should become familiar with the operation of one or more search engines. They will ferret out information for you in seconds, information that would take months to find searching in a traditional library. Individualized personal search engines are available that can search the entire public Web, while enterprise search engines can search intranets, and mobile search engines can search PDAs and even cell phones.

Web Page Makers and Site Builders

To deliver multimedia on the Web today, you should know some HTML, meaning that you must place the proper tags and references into your documents to launch and control your multimedia. Many **HTML editors** and web page–making applications offer to shortcut your HTML learning curve and working effort. If you use one of these editors, enjoy its easing

your work effort, but do not shy away from learning the syntax and tags of the language. Often these "helpers" generate extremely complicated HTML code (described by some programmers as "garbage") with the idea that if this code is hidden "under the hood," who cares? As you yourself become more informed and better at HTML coding, you might discover that you are the person who cares!

HTML documents are simple ASCII text files saved to disk without any formatting at all—no bolding, underlining, special fonts, margins, or tabs. Professional web page developers often use only a word processor like BBEdit for the Mac (see Figure 12-1) or WordPad in Windows rather than a souped-up, drag-and-drop, HTML page builder, and they insert text and tags into their documents manually or with personalized shortcut keys and helper scripts. HTML currently includes about 50 tags, and once you understand their properties and uses, coding, or **marking up**, a document and saving it to your web site can be a straightforward process. Plain HTML may not be enough to create dynamic sites on the fly, sites based upon user preferences or that display "live" information pulled from databases or spreadsheets. To build these kinds of pages, you should be familiar with programming environments such as Microsoft's Active Server Pages (.asp); Adobe's ColdFusion (.cfm), which uses ColdFusion Markup Language (CFML); or the open-source and readily available PHP. For other powerful options beyond plain HTML, knowledge of Dynamic HTML (DHTML), **Extensible Markup Language (XML)**, and Cascading Style Sheets (CSS) will enhance your skill set.

Most web browsers allow you to read the HTML code behind the page you are viewing. In Firefox, click View:Page Source. The SeaMonkey browser, based on the Mozilla engine, includes a basic visual page editor, with several different modes, including a "Tag" mode, which visually shows the tags related to various elements on the page.

TIP *For tools that add power to HTML pages see:*
www.asp.net/
www.activeserverpage.org/
www.adobe.com/products/coldfusion/
www.php.net/

HTML translators are built into many word processing programs, so you can export a word-processed document with its text styles and layout converted to HTML tags for headers, bolding, underlining, indenting, and so on. Some are more powerful than others. These work well for simple text documents but tend to choke on powerful HTML features such as tables, forms, frames, and other extensions. Dedicated editors are

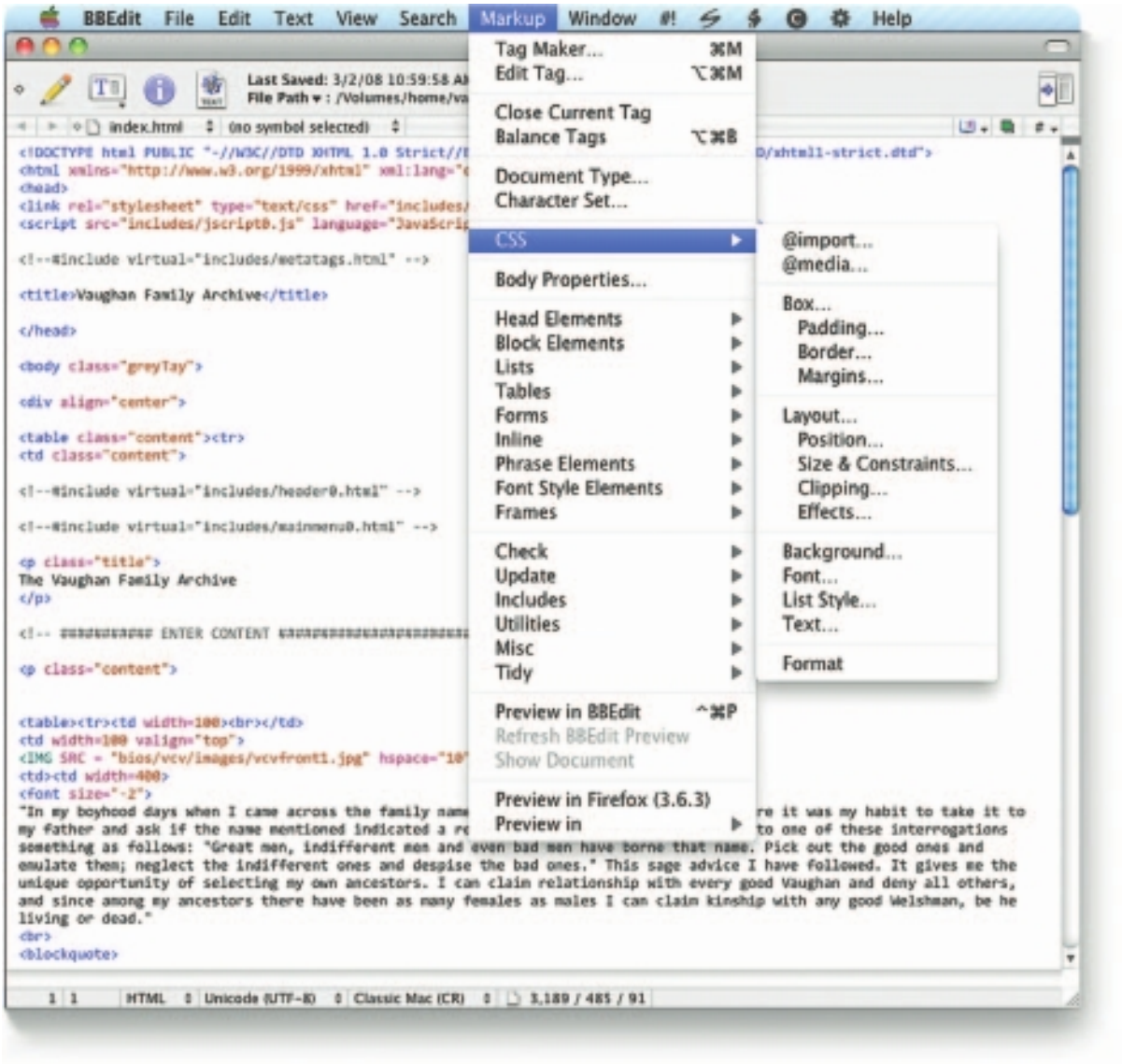


Figure 12-1 BBEdit is a professional programmer's text editor with dedicated features for web page development.

usually **WYSIWYG (What You See Is What You Get)** word processors, and they provide more power and more features specifically geared to exploiting HTML. Microsoft Word, for example, automatically opens web pages in a WYSIWYG view. On the downside, these "helpful" features may cause a page with many embedded graphics to load into the word processor very slowly while it interprets and lays out the page as a browser would, instead of just loading the text of the page's HTML code and letting you change a few tags or lines.

First Person

When I was 16, my grandmother loaned me $500 so I could buy my first car. It was a lovely, previously owned, British racing green 1950 MG-TD, happiest doing about 45 miles per hour on tree-lined summer roads in New England. When you hinged up the hood sideways, everything inside was simple and well defined; there was plenty of room to tweak the twin SU carburetors, adjust the distributor, and replace simple parts like the electric fuel pump. I even took the tiny four-cylinder engine entirely out and replaced the shell bearings on the crankshaft. A decade later, with my previously owned 1960 Ford pickup, it was the same—replacing the radiator or changing the starter motor was a piece of cake, and there was plenty of room to work on the engine. But then automobiles got complicated. It started with elaborate emission control systems, then electronic ignitions, then air conditioning, and finally, computers. Opening the hood of a car today, most of us can only stare dumbly at the myriad hoses and wires and color-coded containers for special fluids; and it's so compact a fit, you can't slip a screwdriver between the engine and the fire wall. When the "check engine" light comes on, an expert needs to "pull" the computer codes with a special, expensive reader to see what's wrong.

Writing HTML for the Web today is still simple. But unless you are an expert, you might be staring dumbly at the complex source code created by a new generation of high-powered, special web tools that will deliver mind-boggling multimedia pages built—no muss, no fuss—with simple drag and drop. But like my car today, which is happiest at 70 miles per hour and could cruise at twice that, these HTML engines won't let you do much under the hood without special tools and knowledge.

Among the many tools in this emerging marketplace, SharePoint Designer from Microsoft links to Microsoft Office and provides not only WYSIWYG support for many of the latest HTML formatting extensions, but also extensive web site management support through its extensions. InDesign from Adobe saves pages as HTML documents and as **Adobe Acrobat PDF** files. Corel's XMetal imports and converts files created in Word, WordPerfect, Ami PRO, and other word processors. It has a point-and-click interface for inserting valid HTML tags and elements and provides an enhanced URL editor to manage references and calls to other documents and files. **Adobe Dreamweaver** is a WYSIWYG editor that lets you create and edit text pages, import images, and link to other documents, and offers enhanced integration with Acrobat PDF files. Dreamweaver has become the most popular WYSIWYG HTML editor today.

Managing and maintaining a web site is a serious undertaking when the site contains many thousands of text documents, images, and other resources. Software and expert system tools for automated web page development, document management, and site activity analysis are becoming widely available. Combined with page builders and multimedia editors,

these applications will evolve into the ubiquitous "word processors" of the new information age, essential to every home and office with outreach to the Web, and able to integrate and present all the elements of multimedia.

Content Management Systems (CMSs) combine the power and flexibility of a database with the dynamic capabilities of a programming language. Most CMSs are built on a combination of mySQL, an open-source database, and PHP, a programming language designed to be included directly in web pages. Here's how it works: when a server receives a request for a web page, it looks through that page's code to see if there are any PHP directives to retrieve data from a database. If it finds such a request, it opens the proper database, grabs the data, and inserts it into the web page as programmed.

CMSs offer prepackaged templates of pages with PHP code built in. Open-source Joomla, one of the most popular CMSs, is used in thousands of web sites, large and small. Drupal is another popular, powerful CMS. Concrete5 combines powerful AJAX technology to allow a more interactive experience. CMSs let non-technical computer users add and edit the content of the pages and manage the presentation and ordering of pages at a web site.

Plug-ins and Delivery Vehicles

Plug-ins add the power of multimedia to web browsers by allowing users to view and interact with new types of documents and images. **Helper applications**, or **players**, also provide multimedia power by displaying or running files downloaded from the Internet by your browser, but helpers are not seamlessly integrated into the operation of the browser itself. When an unrecognized embedded MIME-type that can't be displayed within your browser is called from an HTML document (sounds, movies, unusual text or image files), most browsers will automatically launch a helper application (if it is specified in the browser's preferences) to view or run it. However, this helper starts up and runs separately from the browser.

Many plug-ins are designed to perform special tasks not available without the plug-in installed. If you land on a web page containing embedded, compressed images, for example, and the proper plug-in to decompress those images is not installed, you will not be able to view the images.

Designers work around this problem by including hyperlinks in their pages, which direct the user to the site where the missing plug-in may be found. Users must then download and install the required plug-in, and then restart their browser. This is all a bit cumbersome. Until the marketplace determines which plug-ins will become de facto standards for the Web, however, developers have no alternative. Because downloading and installing plug-ins is perceived as a hassle for the end user, many tool

developers use the Java and JavaScript capabilities already built into today's web browsers. To offer a plug-in's functionality to visitors at your own web site, you may need the addition of MIME-type information to a special setup file on your server that many plug-ins require. If you do not control or operate your own server, you should let your service provider know the MIME-types that you need to have supported. Setting up servers for some of the multimedia plug-ins is not a trivial task, and many Internet service providers will not support high-bandwidth data streams for fear of overwhelming their Internet connection by serving your streaming voice or video to the world. Indeed, while a plug-in or a player may be free and readily available to anyone who wishes it, the software to actually build, compress, manipulate, and serve the special data (such as for compressed images, compressed audio, streaming video, animations, and VRML worlds) may be difficult and expensive, since the company makes money from the development tool, not the client software.

Text

Text and document plug-ins such as the popular **Adobe Acrobat Reader** get you past the display limitations of HTML and web browsers, where fonts are dependent on end users' preferences and page layout is primitive. In file formats provided by Adobe Acrobat, for example, special fonts and graphic images are embedded as data into the file and travel with it, so what you see when you view that file is precisely what the document's maker intended.

Images

Browsers enabled for HTML5 will read and display bitmapped JPEG, GIF, and PNG image files as well as **Scalable Vector Graphics (SVG)** files. Vector files are a mathematical description of the lines, curves, fills, and patterns needed to draw a picture, and while they typically do not provide the rich detail found in bitmaps, they are smaller and can be scaled without image degradation. Plug-ins to enable viewing of vector formats (such as Flash) are useful, particularly when some provide high-octane compression schemes to dramatically shrink file size and shorten the time spent downloading and displaying them. File size and compression sound a recurring theme on the Internet, where data-rich images, movies, and sounds may take many seconds, minutes, or even longer to reach the end user.

Vector graphics are also **device-independent**, in that the image is always displayed at the correct size and with the maximum number of colors supported by the computer. Unlike bitmapped files, a single vector file

can be downloaded, cached, and then displayed multiple times at different scaled sizes on the same or a different web page.

Sound

Sound over the Web is managed in a few different ways. Digitized sound files in various common formats such as MP3, WAV, AIF, or AU may be sent to your computer and then played, either as they are being received (**streaming** playback) or once they are fully downloaded (using a player). MIDI files may also be received and played; as discussed in Chapter 4, these files are more compact, but they depend upon your computer's MIDI setup for quality. Speech files can be specially encoded into a **token language** (a "shorthand" description of the speech components) and sent at great speed to another computer to be un-tokenized and played back in a variety of voices. Sounds may be embedded into QuickTime, Windows Media, and MPEG movie files. Some sounds can be **multicast** (using the multicast IP protocols for the Internet specified in RFC 1112), so multiple users can simultaneously listen to the same data streams without duplication of data across the Internet. Web-based (VoIP, or Voice over Internet Protocol) telephones also transmit data packets containing sound information.

Animation, Video, and Presentation

The most data-intense multimedia elements to travel the Internet are **video streams** containing both images and synchronized sound, and commonly packaged as Apple's QuickTime, Microsoft's Video for Windows (AVI), and MPEG files. Also data rich are the files for proprietary formats such as Keynote, **Microsoft PowerPoint**, and other presentation applications. In all cases, the trade-offs between bandwidth and quality are constantly in your face when designing, developing, and delivering animations or motion video for the Web.

Beyond HTML

When an ingot of pure silicon is "pulled" from a furnace, the process begins with a "seed crystal," around which the ingot forms. HTML is the seed crystal that is shaping and forming the nature of multimedia on the World Wide Web as it extrudes itself onto the Internet's data highway. Within the latticework of HTML servers and browsers, tags such as <OBJECT> (browser-specific for Internet Explorer) or <EMBED> (browser-specific for Firefox) enable text, sound, images, animations, and motion video across the Web. Hooks for powerful platform-independent Java applets and JavaScripts are built into most browsers, so you can design local interaction and activities without a lot of communication between client- and

server-based **Common Gateway Interface (CGI)** programming. CGI is a standard for interfacing external applications with information servers, such as HTTP or web servers, and CGI programs can be written in C/C++, Fortran, PERL, TCL, a Unix shell, Visual Basic, or even AppleScript, as long as the language is supported by the server platform.

The following is an example of the combined <OBJECT> and <EMBED> tags used to display a Flash movie in both the Internet Explorer and Firefox browsers. A browser will only act on a tag it understands, ignoring tags it does not recognize. (See www.adobe.com/cfusion/knowledgebase/index.cfm?id=tn_4150.)

```
<OBJECT classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,40,0"
WIDTH="550" HEIGHT="400" id="myMovieName">
<PARAM NAME=movie VALUE="myFlashMovie.swf">
<PARAM NAME=quality VALUE=high>
<PARAM NAME=bgcolor VALUE=#FFFFFF>
<EMBED href="/support/flash/ts/documents/myFlashMovie.swf" quality=high bgcolor=#FFFFFF
WIDTH="550" HEIGHT="400"
NAME="myMovieName" ALIGN="" TYPE="application/x-shockwave-flash"
PLUGINSPAGE="http://www.macromedia.com/go/getflashplayer">
</EMBED>
</OBJECT>
```

3-D Worlds

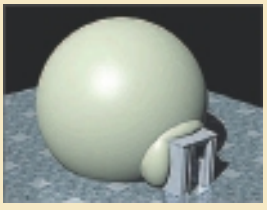
Three-dimensional environments and experiences on the Web are now possible with Intel's Internet 3-D Graphics Software using software such as Second Life, Papervision within Flash, Flash CS4, and **Adobe Director** for development and the Shockwave player for delivery. These have supplanted VRML (Virtual Reality Modeling Language) as an independent environment specifically designed to handle high-performance 3-D worlds containing 3-D text and images, textures, animations, morphs, multiple viewpoints, collision detection, gravity, sounds, and all the arcade elements associated with full-bore game action. With claims that well-executed interactive 3-D content can make nearly any web site more compelling and effective and can better attract, engage, and inform, Intel's algorithms and Adobe's delivery system allow 3-D content to be automatically scaled and tailored to each user's system and available bandwidth. Spinning off from a foundation of vector graphics and animation, 3-D renderings and creative whole worlds present only the latest of multimedia challenges and learning curves for web developers. As this 3-D technology becomes refined and end-user bandwidth increases during the coming years, the very shape of web pages will be altered forever.

TIP *See the following sites for more information about 3-D tools and technology:*
www.adobe.com/products/director/3d/3dservices
www.discreet.com
www.havok.com
www.maxon.net
www.newtek.com
www.nvidia.com
www.righthemisphere.com
www.softimage.com

First Person

When I received a press release from Alternate Realities Corporation, a small startup company spun out of a large research effort in North Carolina, I was intrigued. ARC's president, David Bennett, claimed, "We are redefining Virtual Reality!" He went on to describe "a new generation virtual environment that is a 3-D, immersive, full-color, interactive system enclosed in a 16-foot dome or sphere that can be either portable (inflatable or interlocking) or permanent. The system includes a 360-degree projection system with a 180-degree field of view. Imagine a 16-foot helmet that fits over 15 people at the same time and is nonrestrictive! Larger units (in the 24-foot and up range) are in the early development stage."

I knew I had to have one for my experiments with VRML! The 5-meter model, which fits in a 20 × 20–foot trade-show booth space, was available for $280,000, and the 7-meter model, perfect for my backyard, was only slightly more, at $340,000.



Chapter 12 Review

Chapter Summary

For your review, here is a summary of the important concepts discussed in this chapter.

Discuss the origins of the Internet

- The Internet began as a research network funded by the U.S. Defense Department in 1969.
- In 1989, the National Science Foundation took over its management, and research organizations and universities (professors and students alike) became increasingly heavy users of this ever-growing "Internet."
- Commercial and business use of the Internet was not permitted until 1992, but businesses have since become its driving force.

Define what a computer network is and how Internet domains, addresses, and interconnections work

- A network is a cluster of computers, with one computer acting as a server to provide services such as file transfer, e-mail, and document printing to the client computers.
- Using gateways and routers, a local area network (LAN) can be connected to other LANs to form a wide area network (WAN). These LANs and WANs can also be connected to the Internet through a server that provides both the necessary software for the Internet and the physical data connection.
- The Domain Name System (DNS) manages the names and addresses of computers linked to the Internet.
- Computers on the Internet manage names in subdomains that are encapsulated so that visitors from outside the local network need not worry about the subdomain names.
- When a stream of data is sent over the Internet by your computer, it is first broken down into packets by the Transmission Control Protocol (TCP).
- The IP (Internet Protocol) address is made up of four numbers between 0 and 255 separated by periods.
- Bandwidth is how much data, expressed in bits per second (bps), you can send from one computer to another in a given amount of time. The bottleneck at a typical user's low-bandwidth modem connection is the most serious impediment to sending multimedia across the Internet.
- When a server receives a request, it is handled by a specific application called a daemon that responds to the request based on the protocol.
- The first part of the URL (Uniform Resource Locator) identifies the protocol to use to handle the request.
- Multimedia elements are typically saved and transmitted on the Internet in the appropriate MIME-type (for Multipurpose Internet Mail Extensions) format and are named with the proper extension for that type.
- Hypertext Transfer Protocol (HTTP) provides rules for contacting, requesting, and sending documents encoded with the Hypertext Markup Language (HTML).
- HTML documents are simple ASCII text files. HTML currently includes about 50 tags.
- XML (Extensible Markup Language) allows you to create your own tags and import data from anywhere on the Web.

Discuss the current state of multimedia on the Internet and tools for the World Wide Web

- The explosion of tools and user demand for performance is stressing the orderly development of the core HTML standard. The marketplace keeps this cycle from being too chaotic.

- Interactions on the Web involve communication between two computers: a server and a client. The server delivers a file when a client asks for it.
- Browsers are the apps that run on a user's personal computer to provide the interface for downloading and viewing documents and multimedia.
- Professional web page developers often use only a word processor to edit their pages. Many HTML editors and web page–making applications offer to shortcut your HTML learning curve and working effort. Even if you use one of these editors, you should still understand the syntax and tags of the HTML language.
- Plug-ins allow users to view and interact with new types of documents and images. Requiring visitors to download plug-ins can be cumbersome, and servers must be set up to correctly handle requests for special data types.
- Web-enabled 3-D environments promise compelling and interactive multimedia experiences.

Key Terms

Adobe Acrobat PDF *(380)*
Adobe Acrobat Reader *(382)*
Adobe Director *(384)*
Adobe Dreamweaver *(380)*
Advanced Research Projects Agency (ARPA) *(359)*
applet *(375)*
ARPANET *(359)*
backbone *(365)*
bandwidth *(365)*
browser *(376)*
Cascading Style Sheets (CSS) *(373)*
client *(360)*
Common Gateway Interface (CGI) *(384)*
daemon *(367)*
data packet *(364)*
device-independent *(382)*
Domain Name System (DNS) *(361)*
Extensible Markup Language (XML) *(373)*
FAQ (Frequently Asked Questions) *(358)*
File Transfer Protocol (FTP) *(368)*
helper application *(381)*
HTML editor *(377)*
HTML translator *(378)*
Hypertext Transfer Protocol (HTTP) *(368)*
Internet *(359)*
Internet service provider (ISP) *(364)*
IP address *(364)*
Java *(375)*
JavaScript *(375)*
local area network (LAN) *(360)*
marking up *(378)*
Microsoft Internet Explorer *(376)*
Microsoft PowerPoint *(383)*
Mozilla Firefox *(376)*
multicast *(383)*
Multipurpose Internet Mail Extensions (MIME-types) *(369)*
network *(360)*
player *(381)*
plug-in *(381)*
Post Office Protocol (POP) *(368)*
Scalable Vector Graphics (SVG) *(382)*
Semantic Web *(372)*
server *(360)*
streaming *(383)*
Structured Query Language (SQL) *(372)*
tag *(374)*
TCP/IP *(364)*
token language *(383)*
top-level domain (TLD) *(361)*
Uniform Resource Locator (URL) *(368)*
video stream *(383)*
Web 2.0 *(374)*
wide area network (WAN) *(360)*
WYSIWYG (What You See Is What You Get) *(379)*
XML (Extensible Markup Language) *(373)*

Key Term Quiz

1. Many web sites include pages that have answers to common inquiries. These pages are known as _____.
2. A cluster of computers tied together to share files and communications is a(n) _____.
3. The set of four numbers separated by periods that points to a domain is a(n) _____.
4. How much data, expressed in bits per second (bps), you can send from one computer to another in a given amount of time is called _____.
5. Each Internet service is implemented on an Internet server by dedicated software known as a(n) _____.
6. HTML formatting elements in HTML-encoded pages are called _____.
7. Bits of Java programming code used to extend and customize a browser's basic HTML capabilities are called _____.
8. The applications that run on a user's personal computer and provide the interactive graphical interface for viewing web pages are called _____.
9. Web page editors that visually show how a page looks as you are editing are often called _____ editors.
10. Media that is played as it is being received is said to be _____.

Multiple-Choice Quiz

1. DNS stands for:
   a. Distributed Numbering System
   b. Device Nomenclature System
   c. Data Networking System
   d. Domain Name System
   e. Digital Neighborhood System

2. The levels of a domain name are separated by:
   a. a period
   b. the @ symbol
   c. forward slashes
   d. hyphens
   e. spaces

3. Which of these is not a top-level domain?
   a. com
   b. edu
   c. gov
   d. mil
   e. cis
   f. home.html

4. Which of the following is a valid IP address?
   a. 192.168.1.1
   b. www.apple.com
   c. activa@midcoast.com
   d. http://www.pages.net/index.html
   e. 12 Dreamcatcher Way, Hope, ME 04847

5. When a stream of data is sent over the Internet by your computer, it is first broken down into packets by the:
   a. Transmission Control Protocol
   b. Internet Protocol
   c. Post Office Protocol
   d. Simple Mail Transfer Protocol
   e. File Transfer Protocol

6. Perhaps the most widely installed HTTP software for managing web pages is the open-source application called:
   a. Apache
   b. Daemon
   c. ISP
   d. Acrobat
   e. Unix

7. Web pages are written in:
   a. MPEG
   b. HTML
   c. QuickTime
   d. TCP/IP
   e. MIME

8. One of the greatest benefits of XML is that:
   a. it allows you to create animated rollovers
   b. it compresses audio and video files, allowing larger files to be sent
   c. it connects local area networks with wide area networks
   d. it allows you to create your own tags for data
   e. it encapsulates data into packets for more reliable transmission

9. An IP address can be exchanged with a(n):
   a. MIME-type
   b. Point-to-Point Protocol
   c. domain name
   d. e-mail address
   e. usenet group

10. HTTP stands for:
    a. High-Technology Transmission Protocol
    b. Help Text Translation Protocol
    c. Hypertext Transfer Protocol
    d. Hardware Testing Tool Protocol
    e. How To Talk Protocol

11. The two most popular browsers in terms of number of users are Firefox and:
    a. Acrobat
    b. JavaScript
    c. Shockwave
    d. Internet Explorer
    e. Yahoo!

12. Helper applications run:
    a. on the client computer outside the browser
    b. on the client computer within the browser
    c. on the host computer outside the server
    d. on the host computer within the server
    e. on external devices such as an MP3 player or PDA

13. One criticism of visual page editors is that they:
    a. encrypt the code, making later editing impossible
    b. do not support features such as underlining and bold text
    c. use nonstandard HTML tags
    d. generate extremely complicated HTML code
    e. require the use of plug-ins

14. Which of these tags would most likely be used in HTML to view a multimedia element on the Web?
    a. <BLOCKQUOTE>
    b. <OBJECT>
    c. <OPEN>
    d. <MM>
    e. <FLASH>

15. Which audio file type's sound quality is dependent on the client's computer setup?
    a. AIF
    b. AU
    c. MIDI
    d. Shockwave
    e. WAV

Essay Quiz

1. List the most common web protocols and their uses.
2. Describe what the different parts of the URL http://www.secondLevel.topLevel/filename.filetype represent.
3. Briefly describe how a browser requests a URL, how the URL is handled, and how the server responds, in terms of the DNS system, encoding schemes, and data protocols.

4. Briefly describe the data rates of a 56K dial-up, ISDN, DSL, Cable Modem, and T1 connection. Roughly calculate how long it would take to download a web page consisting of a 5-kilobyte HTML page and 45 kilobytes of image files for each connection.
5. List the most common top-level domains, and describe what categories they are associated with.
6. You have been given the task of creating a new web site for your company. What tools will you use to create the pages? When might you use a word processor? When might you use a WYSIWYG tool? When might you use an HTML text editor? What are the strengths and weaknesses of each?

Lab Projects

Project 12.1

You are given the task of developing a new web site for a cooking magazine. Think about the capabilities of XML to allow you to define your own data tags. What data types would you include? How might you format them? What would be the benefits in this case of being able to define your own tags? Create an outline of how you might structure the data included in a recipe.

Project 12.2

Open a web browser. Locate the "helper applications" preferences panel in the browser. (Note: this option varies among browsers and operating systems, and may not be available for some.)

Find the following MIME-types: audio/x-aiff; application/postscript; application/x-gtar; text/html; image/jpeg; video/mpeg; image/tiff; video/quicktime; audio/x-pn-realaudio. List an application commonly used to read or edit each type.

Project 12.3

Open a word processor that exports to HTML. Create a page of formatted text. Be sure to use different text styles, numerous colors, indenting, tables, and other options. Print the page out from the word processing program.

Export the page to HTML, and then open the HTML page in a web browser. Print out the page from the browser. How does the browser-rendered page differ from the page in the word processor?

Project 12.4

View the source of the page you created in Project 12.4 in the browser. (Most browsers have a function that allows you to view the source HTML code of a web page.) Print out the source, either from the browser, or by copying the HTML code when you view the source, or by cutting and pasting. (Some browsers respond to the View Source command by automatically opening the HTML source in NotePad on the PC.)

Note how HTML treats various word processing features with its tags.

Project 12.5

Go online and compare web hosting packages from three different web hosts. Most hosting companies offer several options and prices. Select a basic, moderate, and advanced package from each host and describe that package's features and options. Do any of the host providers offer unique or unusual options?

Host: _____

Package	Basic	Moderate	Advanced
Storage space:			
Bandwidth:			
Streaming capabilities:			
Cost			

Comments: _____

_____

Host: _____

Package	Basic	Moderate	Advanced
Storage space:			
Bandwidth:			
Streaming capabilities:			
Cost			

Comments: _____

_____

Host: _____

Package	Basic	Moderate	Advanced
Storage space:			
Bandwidth:			
Streaming capabilities:			
Cost			

Comments: _____

_____

CHAPTER 13

Designing for the World Wide Web

In this chapter, you will learn how to:

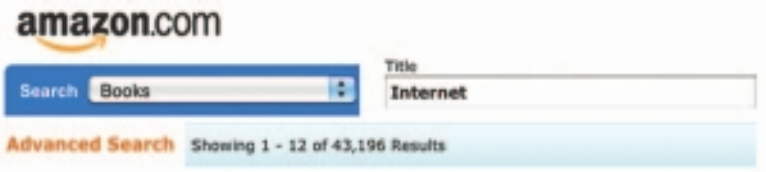
- Employ the basic methods for displaying elements of multimedia on a web page, including using HTML, CSS, and nibbling
- Manipulate the appearance of text on the Web
- Determine which graphics formats are best suited for different types of images and how they can be manipulated
- Play audio on a web page by embedding the sound within the site
- Include animation on a web page
- Include video on a web page with and without the use of plug-ins

LAUNCHED in 1989, the World Wide Web was not originally designed with multimedia in mind, but rather as a simple method for delivering text documents formatted in HTML, with occasional inline graphic illustrations and figures. By 1995, because it was operational, essentially free, and *good enough* to support traffic (see "Vaughan's Law of Multimedia Minimums" in Chapter 4), the Web had become a full-bore information highway of words and pictures with tens of millions of users cruising along it. The Doppler back-draft of passing travelers has exposed the gristle of an overwhelming number of disappointing audio and visual experiences on the Web: "This is my home page; here is a list of my favorite places; this is me with my dog . . ." To fill this vacuum of content and presentation, inventive multimedia solutions and enhancements now compete for mind share, stretching the capabilities of HTML, web browsers, PCs, and the very fabric of the Internet in order to bring multimedia power to this environment. Plain text and pictures are no longer enough for this highway!

WARNING *Powerful multimedia tools can be used to create totally vacuous web pages.*

Developing for the Web

This chapter investigates and illustrates some methods for developing and presenting the basic elements of multimedia within the constraints of HTML, **Cascading Style Sheets (CSS)**, and the World Wide Web. This chapter is not intended to substitute for a more complete library of HTML, CSS, web design, and Internet how-to texts, but to present basic examples that will get you started. In 2001, there were more than 2,000 published books with the word "Internet" in their title. In 2003 there were more than 6,000. In 2006, there were more than 10,000. In 2010, a search at Amazon.com showed 43,196 books with the word "Internet" in their title!



This chapter introduces you to basic ways you can put the elements of multimedia onto a web page. Learn how to use HTML tags with CSS styles rather than rely entirely on web page builders and WYSIWYG editors that never expect you to look under the hood.

All modern browsers allow you to examine the HTML code. Look for a menu item such as "View Source." Use this feature to dig around in the source HTML code of web sites to see how the page is laid out. As you explore, you will discover that some code is neat and clear, some has plenty of embedded descriptive comments, and some is a mess of what programmers call "spaghetti" code.

HTML Is a Markup Language

You should have a basic understanding of HTML and CSS before you begin developing multimedia for the Web. HTML-coded documents, which are the fundamental vehicles for all types of information delivered on the World Wide Web, are explained in Chapter 12, but for this chapter you need to understand the basics of how HTML works.

HTML stands for Hypertext Markup Language. The "Markup Language" part of the name means that tags are used to do such things as format text and embed media. The tags are enclosed by angled brackets: <>. Some tags are bounding tags, requiring both an opening tag and a closing tag. The closing tag is indicated by a leading forward slash inside the angled brackets. This example for bolded text illustrates the use of the two tags:

```
<strong>This text is emphasized</strong>
```

Other tags, such as the tag for inserting an inline image, stand by themselves:

```
<IMG src="grey_ball.gif">
```

Note that the tags may be written in either upper- or lowercase; some HTML text-editing programs have a switch allowing you to select the case in which you want the tags written in your document.

Tags listed in Table 13-1 are used by **HTML5**. These simple tags along with CSS elements are used to build web pages. **Deprecated** tags such as <FONT> and <CENTER> are no longer supported in the HTML standard, yet their use continues to be supported by most browsers.

HTML Tag	Description	HTML Tag	Description
<!-- -->	Defines a comment	<footer>*	Defines a footer for a section or page
<!DOCTYPE>	Defines the document type	<form>	Defines a form
<a>	Defines a hyperlink	<h1> to <h6>	Defines header 1 to header 6
<abbr>	Defines an abbreviation	<head>	Defines information about the document
<address>	Defines an address element	<header>*	Defines a header for a section or page
<area>	Defines an area inside an image map	<hgroup>*	Defines information about a section in a document
<article>*	Defines an article	<hr>	Defines a horizontal rule
<aside>*	Defines content aside from the page content	<html>	Defines an HTML document
<audio>*	Defines sound content	<i>	Defines italic text
<b>	Defines bold text	<iframe>	Defines an inline subwindow (frame)
<base>	Defines a base URL for all the links in a page	<img>	Defines an image
<bdo>	Defines the direction of text display	<input>	Defines an input field
<blockquote>	Defines a long quotation	<ins>	Defines inserted text
<body>	Defines the body element	<keygen>*	Defines a generated key in a form
 	Inserts a single line break	<kbd>	Defines keyboard text
<button>	Defines a push button	<label>	Defines a label for a form control
<canvas>*	Defines graphics	<legend>	Defines a title in a fieldset
<caption>	Defines a table caption		Defines a list item
<cite>	Defines a citation	<link>	Defines a resource reference
<code>	Defines computer code text	<map>	Defines an image map
<col>	Defines attributes for table columns	<mark>*	Defines marked text
<colgroup>	Defines groups of table columns	<menu>	Defines a menu list
<command>*	Defines a command button	<meta>	Defines meta information
<datalist>*	Defines a drop-down list	<meter>*	Defines measurement within a predefined range
<dd>	Defines a definition description	<nav>*	Defines navigation links
<del>	Defines deleted text	<noscript>	Defines a noscript section
<details>*	Defines details of an element	<object>	Defines an embedded object
<dfn>	Defines a definition term		Defines an ordered list
	Defines a section in a document	<optgroup>	Defines an option group
<dl>	Defines a definition list	<option>	Defines an option in a drop-down list
<dt>	Defines a definition term	<output>*	Defines some types of output
	Defines emphasized text	<p>	Defines a paragraph
<embed>*	Defines external interactive content or plug-in	<param>	Defines a parameter for an object
<fieldset>	Defines a fieldset	<pre>	Defines preformatted text
<figcaption>*	Defines the caption of a figure element	<progress>*	Defines progress of a task of any kind
<figure>*	Defines a group of media content, and their caption	<q>	Defines a short quotation

Table 13-1 Showing All HTML Tags Used for Building Web Pages. An Asterisk Marks Tags That Are New for HTML5 (see www.w3schools.com/html5/html5_reference.asp)

HTML Tag	Description	HTML Tag	Description
<rp>*	Used in ruby annotations to define what to show browsers that do not support the ruby element	<sup>	Defines superscripted text
<rt>*	Defines explanation to ruby annotations	<table>	Defines a table
<ruby>*	Defines ruby annotations	<tbody>	Defines a table body
<samp>	Defines sample computer code	<td>	Defines a table cell
<script>	Defines a script	<textarea>	Defines a text area
<section>*	Defines a section	<tfoot>	Defines a table footer
<select>	Defines a selectable list	<th>	Defines a table header
<small>	Defines small text	<thead>	Defines a table header
<source>*	Defines media resources	<time>*	Defines a date/time
	Defines a section in a document	<title>	Defines the document title
	Defines strong text	<tr>	Defines a table row
<style>	Defines a style definition		Defines an unordered list
<sub>	Defines subscripted text	<var>	Defines a variable
<summary>*	Defines the header of a "detail" element	<video>*	Defines a video

Table 13-1 Showing All HTML Tags Used for Building Web Pages. An Asterisk Marks Tags That Are New for HTML5 (see www.w3schools.com/html5/html5_reference.asp) *(Continued)*

HTML and Multimedia

HTML provides tags for inserting media into HTML documents: the **<IMG> tag** for inline images; the **<AUDIO>** and **<VIDEO> tags** for multimedia; and the **<EMBED>** and **<OBJECT> tags** for compound document embedding used to insert a "nonstandard" item such as a Java applet or Flash animation into an HTML document.

But it is not as simple as it seems. There is a difference between the way various versions of browsers handle multimedia elements and the plug-ins that play them. Some browsers that understand the <OBJECT> tag ignore the <EMBED> tag, and some browsers that cannot read the <OBJECT> tag need the <EMBED> tag. The **Object/Embed method** places an <EMBED> tag within the <OBJECT> tag to ensure that multimedia elements will play in all browsers. Thus the HTML code to play a flash animation might look something like this:

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" width="320" height="240" id="player1" name="player1">
   <param name="flashvars" value="file=playlist1.xml">
   <param name="movie" value="player.swf">
   <param name="allowfullscreen" value="true">
   <param name="allowscriptaccess" value="always">
   <embed id="player1"
          flashvars="file=playlist1.xml"
          name="player1"
```

```
            src="player.swf"
            width="320"
            height="240"
            allowscriptaccess="always"
            allowfullscreen="true"
    />
</object>
```

TIP *If you develop multimedia for the Internet, budget time and effort for keeping current in this rapidly changing environment—staying at the leading edge takes effort. It will be some years before multimedia delivery tools and techniques for the Web stabilize.*

The Desktop Workspace

Make your web pages look good on a 1024 × 768 display in true color (millions). Working at this resolution, you will satisfy more than 95 percent of all desktop viewers. Depending upon the browser and preferences set by the user, however, the area of the screen available for your web page, called the **viewport**, will always be less than the full display, and it is not controllable by the designer. Browser "**chrome**" (toolbars and other shiny stuff around the edges of your page's viewport) can be either hidden or shown by the user. If you want to maximize the browser active window size, in Internet Explorer press F11 (function key 11) and go back to regular mode by clicking the mouse; other browsers offer Full Screen toggle switches in the "View" options. So design your web page for a 1,024-pixel-wide display by using tables and images that do not exceed about 1,000 pixels across the page, and you will have room for browser scroll bars. Many designers choose a viewport workspace 960 pixels wide; a number divisible by 3, 4, 5, 6, 8, 10, 12, 15, and 16. This makes many logical "grid systems" of columns possible.

The Small-Device Workspace

Under the hood of many browsers is a layout engine for rendering pages. Versions of those browser engines have been customized to run on small devices such as tablets, e-readers, netbooks, PDAs, and smartphones, and they follow known rules when laying out web pages for smaller viewports. Microsoft Internet Explorer uses the Trident engine; Firefox uses the Gecko engine; Opera uses the Presto engine; and Apple's Safari and Google's Chrome use the WebKit engine.

Smartphones use various operating systems: Android, iPhone OS, Linux, Maemo, Palm WebOS, RIM's BlackBerry, Symbian OS, and Windows CE, with the most widely used being Symbian OS on handsets and devices manufactured by BenQ, Fujitsu, LG, Mitsubishi, Motorola, Nokia, Samsung, Sharp, and Sony Ericsson.

To deal with the multiplicity of viewport sizes in the small-device world (320 × 480, 240 × 320, 240 × 400, 854 × 480, etc.), the Android OS allows programmers to write one application that flexibly covers all display sizes by using virtual **density-independent pixels (dips)**:

> The density-independent pixel is equivalent to one physical pixel on a 160 dpi screen, the baseline density assumed by the platform.... At run time, the platform transparently handles any scaling of the dip units needed, based on the actual density of the screen in use. The conversion of dip units to screen pixels is simple: `pixels = dips * (density / 160)`. For example, on a 240 dpi screen, 1 dip would equal 1.5 physical pixels. Using dip units to define your application's UI is highly recommended, as a way of ensuring proper display of your UI on different screens.
>
> From the Android Developers Guide,
> http://developer.android.com/guide/practices/screens_support.html

While HTML and CSS do not provide for device-independence, if you expect that your project will be widely viewed on small devices, consider designing at 960 pixels to allow the device's browser the most flexibility. More, you should keep in mind that input events on small devices are different from the clicks and drags of a computer with mouse or touchpad: a **Double Tap** makes the browser zoom in and center on a document; a **Touch and Hold** will display an information bubble; a **Drag** will move the viewport or pan; a **Flick** will scroll up or down; and a **Pinch Open** or **Pinch Closed** will zoom in or out. There are no mouseOver events without a mouse.

Nibbling

A principle you must always keep in mind when designing and making multimedia elements for the Web and particularly for handheld devices should be called "nibbling." At a serious metal-working supply store you can buy a power tool called a nibbler—it devours the edges of sheet metal in an ear-damaging staccato of rapid tiny bites. You must apply this concept, for example, to the elegant bitmapped logo you created in Photoshop when you trim it from 24- to 8- to 4-bit color depth and resize it from 96 pixels square to 64 pixels square and create a transparent .png file. Nibble the audio clip of your client's theme song from 44.1 kHz to 11 kHz, and see if it's acceptable at an 8-bit sample size. Text as HTML is cheap: nibble your page design and throw away the pretty shadowed GIF graphic headers and image maps—re-create your text in HTML headers or emphasized text, and try coloring it. Put on your protective headgear—this compromising work is painful for you as the creator—and start nibbling, while

constantly seeking a balance between quality and the patience of a user who is downloading your material at 56 Kbps from home. Every choice you make should be tempered by bandwidth worry.

WARNING *For every image file referenced in an HTML document, a separate Internet HTTP connection must be made between your computer and that image's server before the image itself is downloaded; so using many* different *tiny images (such as various graphic images for bullets) may not be efficient. After a user has downloaded a file once, however, it should load more quickly from the user's local hard disk, where the browser stores them in a cache.*

Text for the Web

In addition to variations in the size of the viewport, viewers of your web site may not be displaying the same "preferred" font that you used to design your page because user preferences in the browser may alter the way text in your document looks and flows. To make the best of this uncertainty, many developers design their documents in Times Roman for the proportional serifed font, Verdana for proportional sans serif, and Courier as the monospaced font. These fonts readily move across platforms and are the default fonts users typically see if they do not set their own preferences. Although you can specify a font, and even alternate fonts, using CSS, browsers can only use a specified font if that font is installed on the end user's computer. Figure 2-3 in Chapter 2 shows a list of the most commonly installed fonts on Windows, Macintosh, and Linux computers.

NOTE *As with projects built for CD-ROM or DVD distribution using a multimedia authoring tool, if you wish to absolutely control the look of text on your web page, you must use a graphic bitmap rather than text in your HTML document. Adding images in place of text, however, increases the amount of time necessary to download your page. Embedding graphics into HTML documents is explained later in the chapter.*

You can tag text so that it is displayed as a header, strong, emphasized, or sub- or superscripted. Using CSS, you can specify your "preference" for font face and many text attributes (see Chapter 2), but the viewer's browser ultimately determines if and how these styles are displayed.

Making Columns of Text

The most powerful feature of HTML may be found in the **<TABLE> tag**. Study this tag and its attributes! To the right, you'll see how to organize your text into two columns, so it displays more like a newspaper or a magazine, using a table (see Figure 13-1).

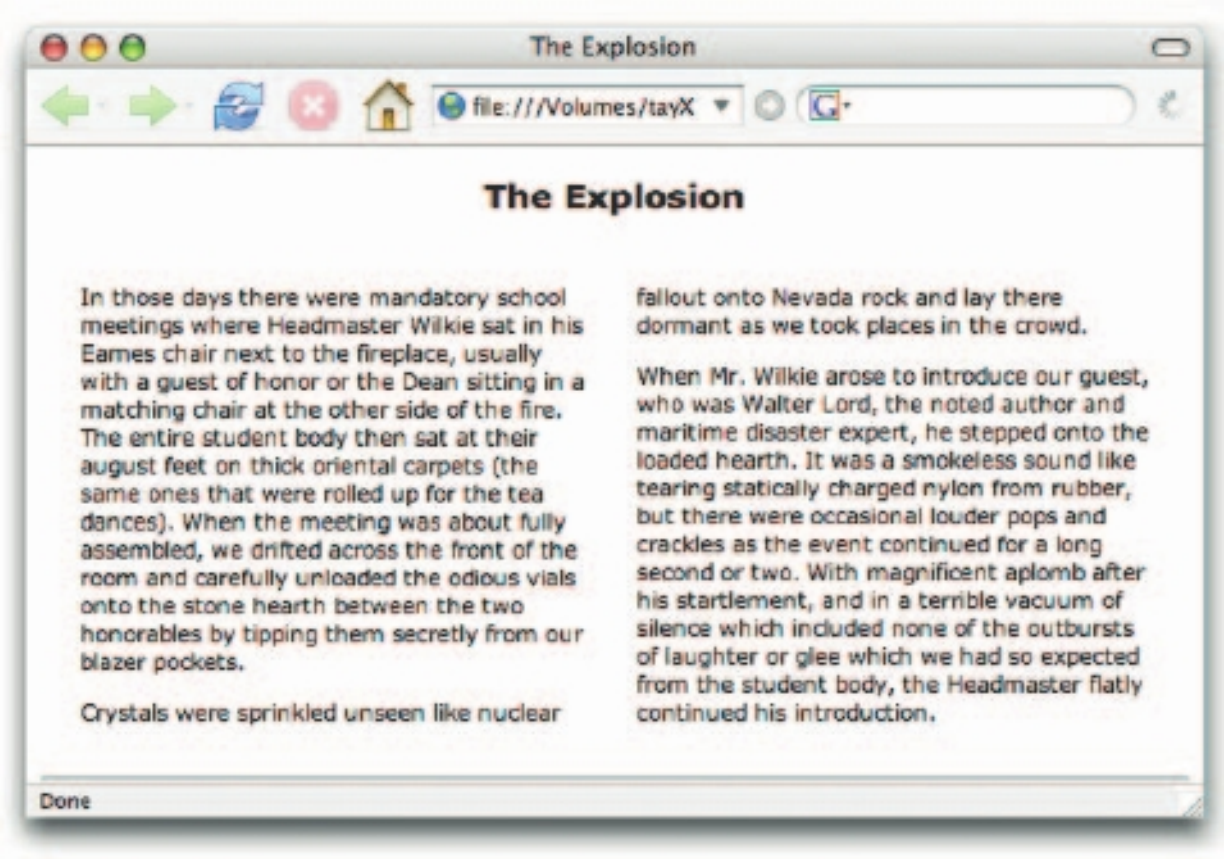


Figure 13-1 Using the <TABLE> tag, you can organize your text into columns.

```
<HTML>
<HEAD>
<TITLE>The Explosion</TITLE>
</HEAD>
<BODY>
<DIV ALIGN="center">
<H2>The Explosion</H2>
</DIV>
<TABLE BORDER="0" CELLSPACING="20">
<TR VALIGN="TOP">
<TD WIDTH="40%">
... text for Column 1 goes here ...
</TD>
<TD WIDTH="40%">
... text for Column 2 goes here ...
</TD>
</TR>
</TABLE>
<HR>
</BODY>
</HTML>
```

Flowing Text Around Images

As you can see in Figure 13-2, it is possible (and easy) to "flow" text around an image using the **ALIGN attribute** of the <IMG> tag. This is a quick and simple method for mixing text and images in a pleasing layout. Add a <BR CLEAR="left"> tag at the end of your text paragraph, so that if there is not enough text to fill the entire vertical height of the image, your next paragraph will begin on a new line, left-justified, and below the image. To add space around your image so it doesn't butt right up against the text, use the Horizontal Space (**HSPACE**) and Vertical Space (**VSPACE**) **attributes** of the <IMG> tag.

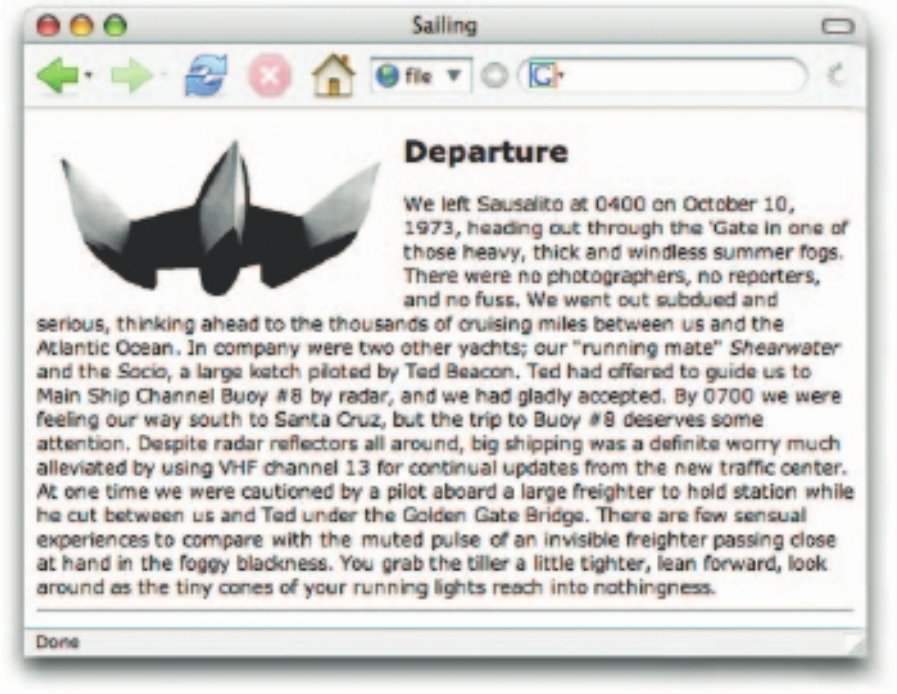


Figure 13-2 You can flow text around an image by using the ALIGN attribute of the <IMG> tag.

```
<HTML>
<HEAD>
<TITLE>Sailing</TITLE>
</HEAD>
<BODY>
<IMG SRC="gbsky.gif" ALIGN="left" HSPACE="15" VSPACE="5">
<H2>Departure</H2>
... text goes here ...
<BR CLEAR="left">
<hr>
</BODY>
</HTML>
```

The following HTML and CSS code sets up a more complicated screen with flowing text (see Figure 13-3). It also includes a background image, a portrait image, and an image map that is used for navigation. (Background images and image maps are described later in this chapter.) This document also contains the foreign language special character *ä*, which is called out in the document using HTML's escape sequence for special characters, in this case, "ä". An **escape sequence** begins with an ampersand and ends with a semicolon. Also, note the link to a separate style sheet file holding the CSS code. A MIDI file is embedded in this page to provide background music.

Figure 13-3 Images, text, and sound can be mixed in an HTML document. Note the use of escape sequences for special characters and an image map for navigation.

```
<html>
<head>
<title>Annan Lapsuus</title>
<link rel="stylesheet" type="text/css" href="anna.css">
</head>
<body>
<table><tr>
<td width="20"></td> <!-- blank spacer column for indent -->
<td width="180" align="center" valign="top"> <!-- Column to hold image -->
<img src="anna.jpg" align="left">
</td>
<td width="400" align="left" valign="top"> <!-- Column to hold text and nav button -->
<h1>Annan Lapsuus</h1>
<p class="annaText">
Min&auml; sain oman huoneen. Sen sein&auml;t on maalattu vihreiksi. Ja yhdelle sein&auml;lle on
maalattu maisema. Mutta joelle ei maalattu joutsenia, koska min&auml; en halunnut. Niihin voi
kyll&auml;sty&auml;niin helposti.
<br><br>
Isi on tehnyt minulle kirjahyllyn. Min&auml; j&auml;rjest&auml;n siihen kaikki tavarat.
Kiiltokuva-albumit ja kirjan. Sen nimi on "Tiina saa suukon". Vaikka on minulla muitakin
kirjoja, mutta en min&auml; en&auml;&auml; sellaisia lastenkirjoja lue.
<br><br>
"T&auml;st&auml; l&auml;htien minun huoneeni on aina hyv&auml;ss&auml; j&auml;rjestyksess&auml;",
sanoin isille.
<br><br>
<img src="navButton.gif" border="0" usemap="#thispagemap">
Isi hymyili.
</td></tr></table>
```

```
<!-- Use image map for click navigation -->
<map name="thispagemap">
<area shape="circle" coords="48,48,12" href="fhelp.htm">
<area shape="polygon" coords="50,50,0,0,100,0" href="fnavmap.htm">
<area shape="polygon" coords="50,50,0,100,100,100" href="f03.htm">
</map>
<!-- play MIDI file on this page -->
<EMBED SRC="03/pianobg.mid" width="0" height="2" autostart="true">
</body>
</html>
```

And the accompanying CSS code:

```
body {
    background-image: url(earth.jpg);
    }
H1 {
    font-family: "Lucida Grande", "Trebuchet MS",  Verdana, Helvetica, sans-serif;
    color: #FFFFFF;
    font-size : 24px;
    font-weight: bold;
    }
p.annaText {
    font-family: "Lucida Grande", "Trebuchet MS",  Verdana, Helvetica, sans-serif;
    color: #FFFFFF;
    font-size : 14px;
    font-weight : bold;
    line-height : 18px;
    text-align: left;
    padding-right: 20px;
    }
```

Images for the Web

Theoretically, the Web can support any graphics format the client and server have in common. Practically, even though web standards do not specify a graphics format you must use, browsers recognize four image formats—GIF, PNG, JPEG, and SVG—without resorting to special plug-ins. These formats use built-in **compression algorithms** to reduce file size. (Graphic image formats are described in detail in Chapter 3.) For other graphics formats, such as CGM, CMX, DXF, and fractal- and wavelet-compressed images, special proprietary creation software and browser plug-ins may be required.

GIF and PNG Images

GIF images (Graphic Interchange File, also discussed in Chapter 3) are limited to 8 bits of color depth (256 colors). This is a commercial image format developed by CompuServe Information Services, an online

company once owned by Unisys and currently folded into America Online. In late 1994, Unisys announced a patent fee charge to all software developers who use the GIF format. In an angry, industry-wide response, **PNG** (for Portable Network Graphics Specification) was developed as a new "open" format (not requiring fees) to replace GIF. By allowing transparency by single pixel or by alpha channel mask and a 24-bit indexed palette, the PNG format is an improvement on the GIF format it was intended to replace. But it does not support animation. And because it only uses the RGB color model (not CMYK), PNG images may not print well.

First Person

A few years ago somebody told me about an interesting web survey: how does the world pronounce GIF? The results turned out about 50/50 on the hard/soft question, my colleague claimed. Then I spent considerable time using that word (softly) in Europe before realizing everybody was being smirkingly polite about my outlandish pronunciation. In the San Francisco Bay Area, a world center for multimedia development, GIF has the soft "g" of "ginger," "gin," and "gybe." In New York, where little is soft, and in Europe, GIF has a more cutting, hard pronunciation, as in "giggling," " gingham," "girdled," "guilty," or "girls." The real question is whether the written word requires a prefixed dot.

JPEG Images

JPEG (Joint Photographic Experts Group) images may contain 24 bits of color depth (millions of colors). JPEG uses a powerful but **lossy** compression method that produces files as much as ten times more compressed than GIF. Lossy means that information in the original image is lost in the compression process and cannot be retrieved. A **lossless** compression method does not irretrievably discard the original data.

WARNING *Do not edit and reedit files that are in JPEG format. Every time you open a JPEG image and edit it, then recompress and save it as a compressed JPEG, the image degrades. After a few editing/saving cycles, you will be very disappointed. Edit and archive your images in a 24-bit lossless graphic format (such as TIFF, BMP, or PSD), then convert to JPEG (if you need to).*

The JPEG compression scheme compresses about 20:1 before visible image degradation occurs. Test the amount of compression acceptable for your JPEG image; stay inside the "threshold of visible error." To compress an image with JPEG, the image is divided into 8 × 8–pixel blocks, and the resulting 64 pixels (called a "search range") are mathematically described

Figure 13-4 Both images at the top were saved in the JPEG format, which compresses image data and trades image quality for small file size. The resulting compressed images at the bottom show the "lossy" and "blocky" nature of compressed JPEGs. The photo at top left is 71K in size when saved as a GIF (not shown) and only 27K saved as a JPEG (bottom left). The drawing at top right is 17K when saved as a GIF (not shown) and 46K as a JPEG (bottom right).

relative to the characteristic of the pixel in the upper-left corner. The binary description of this relationship requires far less than 64 pixels, so more information can be squeezed into less space. JPEG compresses slowly, about one to three seconds for a 1MB image, depending upon computer speed, but JPEG can compress images as much as 75:1, with loss.

GIF or JPEG?

Use JPEG for photo-realistic images containing many colors, and avoid using it for images already forced into a 256-color palette or for line drawings or 1-bit black-and-white images. GIF compresses drawings and cartoons that have only a few colors in them much better than JPEG, which may introduce visible defects—sharp edges and lines that blur—especially with small-size text. Figures 13-4 and 13-5 show the "blocky" and "lossy" nature of the compressed JPEG images. For the Web, use the JPEG format for photo-realistic images that are busy with color; use the GIF format for line art and drawings where there are large areas of the same color.

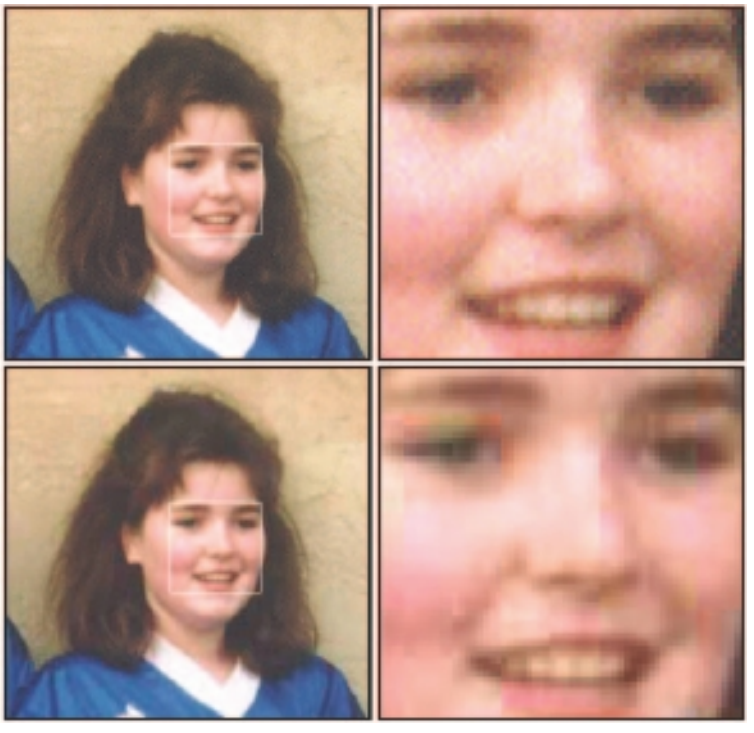


Figure 13-5 Lossy compression schemes save disk space but can also degrade an image. For the Web, line art is often better saved in GIF, PNG, or SVG format than in JPEG.

Using Photoshop

Adobe's Photoshop is the "tool of choice" for most graphic artists, so it is worth taking some time to provide a few suggestions for creating images for use on the World Wide Web. If you use a different image-editing application, follow the same logic and use the commands appropriate for that application. Always work in native Photoshop format using PSD files—these images are typically in RGB mode and use the maximum color depth. They are larger, but they contain more information that can be usefully processed when resizing and dithering, and you will get better final results. PSD files also contain layers, a very useful application feature. When creating images for display on a web page, use 72 pixels per inch resolution, which is the resolution of most monitors. When you convert a 24-bit RGB image to an 8-bit indexed image (change its mode), you lose huge amounts of color information that cannot be retrieved, meaning that the fine data is gone forever. So you should follow two practices in order to protect your original image. One is to save the original image in a 24-bit lossless image format (such as PSD, TIFF, or BMP). The other is to do all of your image manipulation (such as resizing, sharpening, and hue adjustments) in RGB mode. Next, save this source image in RGB mode as a PSD file, before reducing the color palette by saving it as a GIF or using a lossy compression like JPEG. By saving the high-quality original and saving the manipulated image in the program's native format, you can return to them if you need to make changes later.

TIP *When you scan an image, the scanner will often default to print resolution of 300 dpi. When displayed on a 72 dpi resolution monitor, the picture will be displayed more than three times bigger than the original. Never fix this problem by changing the height and width attributes of the IMG tag. Even though this will display the image at the size you want, you still have a huge image file that will slow down the downloading and display of your page. Instead, use Photoshop or another image-editing program to resample the image at a 72 dpi resolution, and use that new image on your page.*

When you are satisfied with your image and ready to save it as a GIF, PNG, or JPEG file, archive it as described earlier. If you make any mistakes while converting modes or saving, you will still have the original, complete with any layers you might have used. To be very safe, duplicate the original file and open the copy before saving to other formats.

Saving as JPEG Files

To save your image as a JPEG file, you do not need to change Photoshop's mode from RGB, but if you are using layers, you will need to "flatten" the image, merging all layers into a single bitmap. Once an image is flattened and you have edited or saved it, its layers cannot be remade without a great

deal of difficult cutting and pasting—so again, archive your original file! You must name your file with the extension .jpg or .jpeg if you will use it on the Web. Then click Save, and choose Maximum, High, Medium, or Low-quality compression in the dialog box that appears. Your file is ready for the Web.

Saving as GIF Files

To save a GIF file using Photoshop, you must first set the mode of your image to Indexed Color, converting it to the best 8-bit palette (256 colors) that will represent the image and be displayed well by web browsers. Note that the option of saving a Photoshop 24-bit RGB file in GIF format will not be available in Windows, and it will be grayed out on the Macintosh menu until you have converted your image to 8-bit mode: GIF is only for 8-bit images.

TIP *Use GIF files for line art and images that contain large areas of the same color (that can be easily compressed). Use JPEG for photo-realistic images.*

Palettes When you change the mode to Indexed Color, you must specify the color depth of the converted image, the color palette to be used, and whether the colors of your image should be dithered (Diffusion or Pattern) or not (None). Figure 13-6 shows the mode changing dialog box from Photoshop, where the custom Netscape Navigator palette for Windows has been selected.

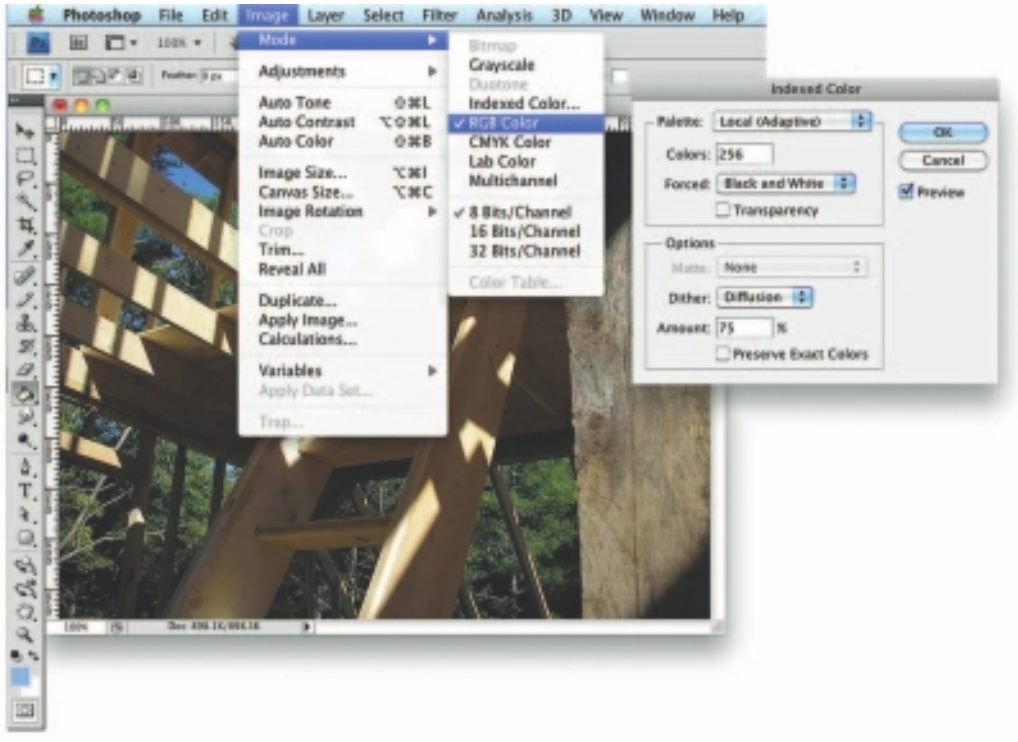


Figure 13-6
In Photoshop, changing the mode of your image from RGB Color to Indexed Color changes the color depth of your image.

Interlaced and Progressive Scans Both GIF and JPEG images can be saved so that when your browser displays the image as it is being downloaded, you can immediately see a chunky approximation of the final image, with resolution improving as more and more data comes in. While in baseline, or normal configuration, image data is stored as a single top-to-bottom scan; in **interlaced** GIF and **progressive** JPEG files, the data is organized in a different sequence within the file. An interlaced GIF file, for example, is arranged into a series of four passes:

Pass 1 : Every 8th row, starting with row 0

Pass 2 : Every 8th row, starting with row 4

Pass 3 : Every 4th row, starting with row 2

Pass 4 : Every 2nd row, starting with row 1

Figure 13-7 shows Photoshop's Save for Web & Devices dialog for saving an image as interlaced, and four increasingly resolved images.

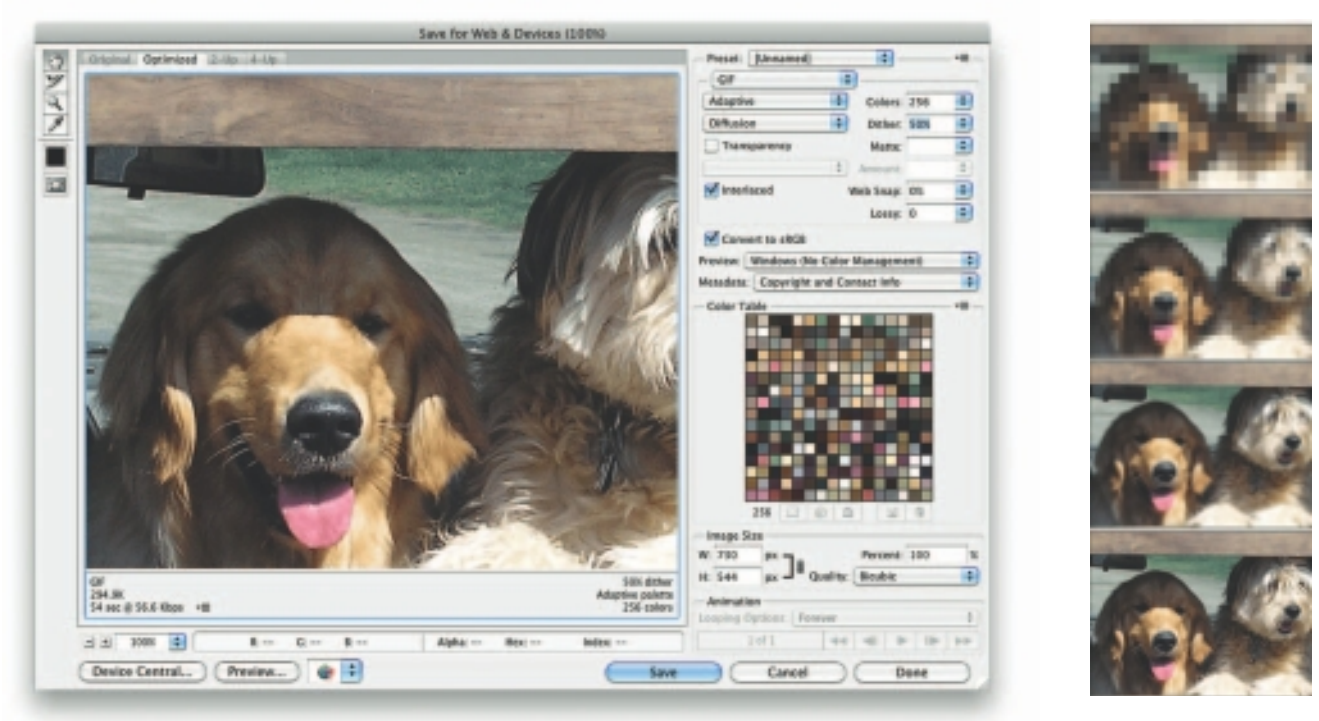


Figure 13-7 Interlacing settings when exporting a GIF89a file from Photoshop. With interlacing, the image incrementally improves its resolution as it downloads.

Transparency The GIF89a and PNG specifications allow for **transparency**: you can save your file with instructions to a browser to use a specific color or palette of colors (with PNG) as your selected

transparency color. In many cases, such as for company logos and inline illustrations, it is attractive to let an image float on top of the browser's background.

Images on web pages are displayed as rectangles. The area outside of the circle in Figure 13-8 is filled with a wash of color and would (without transparency) be displayed as a rectangle showing those colors to its edges. To make the part surrounding the circle transparent so that the circle floats on your web page, fill the area outside the circle with a single color, and then save the file, selecting that fill color to be transparent. While white is often used as the transparency color, in this example it would not work because there are white pixels inside of the circle that would also become transparent. Use a fill color not in the area you wish to show; in this case red works. Most image-editing tools provide a palette from which you can select the transparency index color. You cannot make a JPEG file transparent.

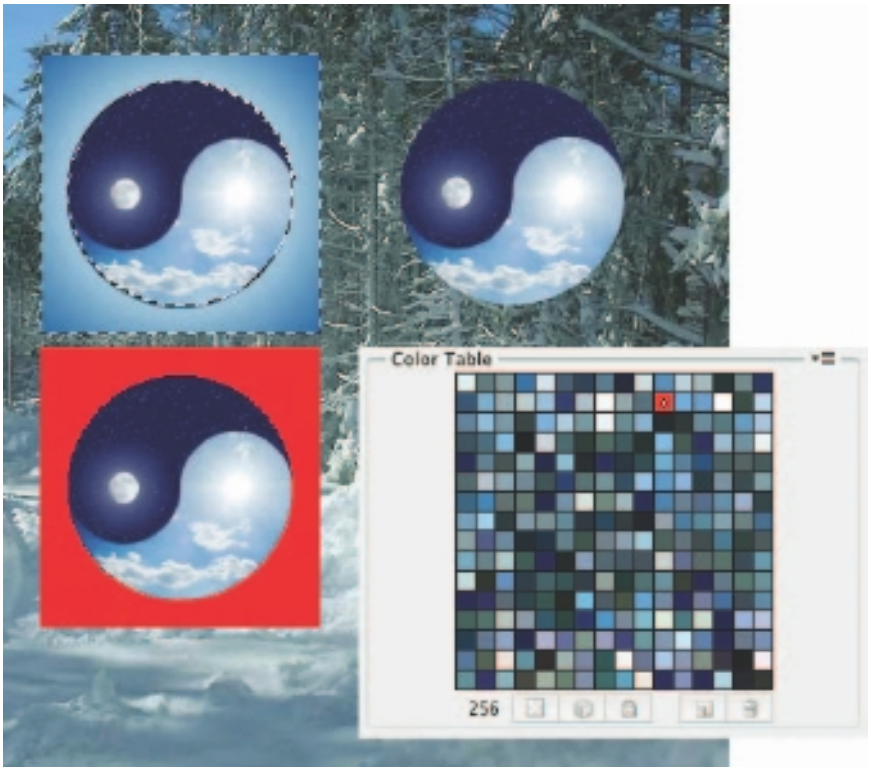


Figure 13-8 Use a transparent GIF or PNG to float a circle or other image on a web page: select the area outside of the circle (upper left), fill it with a single indexed color (red, lower left), choose that color to be transparent (lower right), and save the image as a GIF file. The circle will float on your page (upper right).

Backgrounds

Most browsers allow you to specify an image or color to place in the background of your page or into table cells. Text and images will float on top of this layer.

Background Coloring

You can choose colors for backgrounds, text, and anchors to URL links. Color controls for the entire page are attributes of the **<BODY> tag** and are set using CSS:

```
body {background-color: #0000FF;}
```

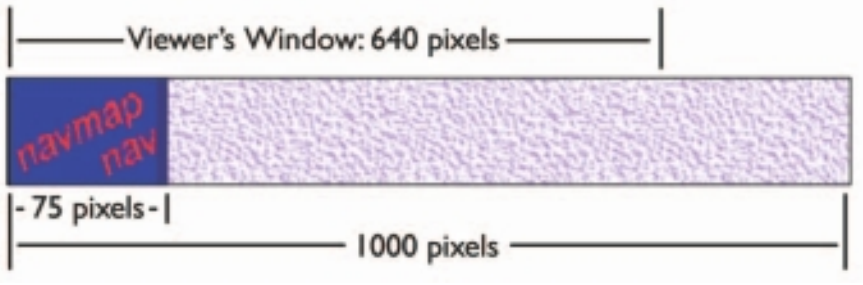
where "#0000FF" is a hexadecimal red-green-blue triplet used to specify the background color, in this case, blue. See Chapter 3 for an explanation of red-green-blue triplets.

Once you have chosen a background color, you will then want to set the color of your text and establish proper contrasts. Red on green shimmers, black on black is invisible. By setting styles in the <BODY> tag, you set default styles for the entire document. For white text on a blue background, the CSS code would be:

```
body {color: #FFFFFF;}
```

Background Images

Background images are by default tiled, or repeated, across and down the page until the page or page element is filled, so a randomly distributed "sandy" background image (see Figure 13-9) can easily be made from a very small source image.



Load a background image into a document by specifying its URL (if it is available somewhere on the Web) or its relative file path (if it is on the same server as the page) in the CSS attributes for the <BODY> tag, for example:

```
body {background-image: url('paper.gif');}
```

TIP *It is a good idea to specify a background color similar to the prevailing color of the image being used for a background. If the user viewing your page has Image Loading turned off, or if your background image cannot be found for some reason, the page may still look close to the way you designed it. If the image you specify as a background has transparent areas, the background color will show through.*

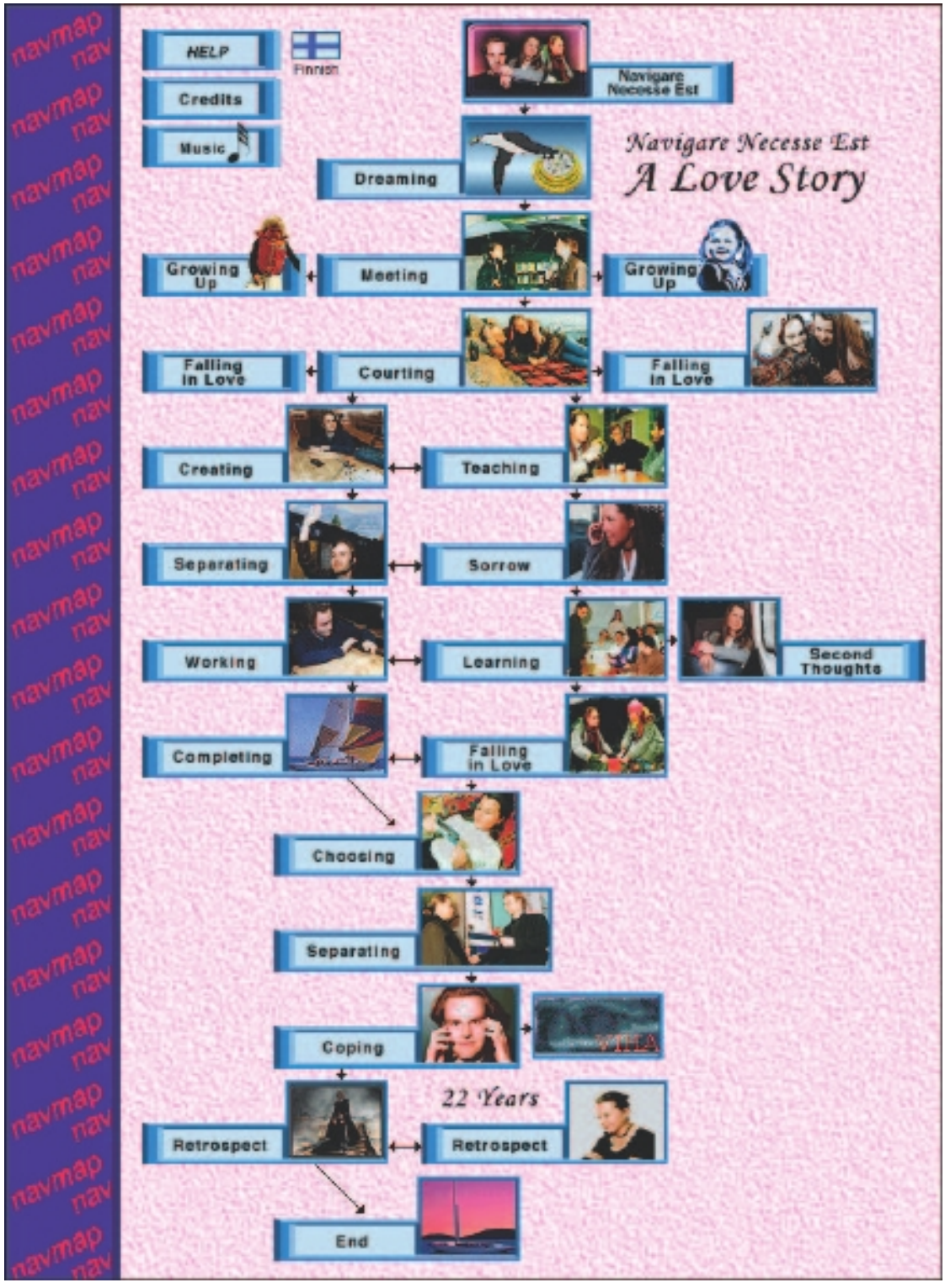


Figure 13-9 A simplified navigation map not only provides an overview of a multimedia project but also contains active links to documents using "hot" areas of the graphic. This is the navmap page from Navigare Necesse Est, a student-built love story.

Sidebars

In the navigation map shown in Figure 13-9, a commonly seen graphic layout was used: a vertical bar containing the word "navmap" is displayed at the left of the screen and in the background. When users scroll up or down, this bar remains stationary. Make the graphic bar at the left as wide as you wish (say 75 pixels); then set the full width of your image to 1,000 pixels. Fill the space to the right of your bar with plain color or a texture. When this background image repeats itself (tiles), it will repeat to the right only if the user widens the viewing window to more than 1,000 pixels; but the image will tile vertically in increments of its height until it reaches the bottom of the window. With CSS you can force the browser to repeat only vertically, only horizontally, or not at all. In this example, adding *background-repeat: repeat-y;* to your CSS code will allow repeats only vertically, even when the window is made wider than 1,000 pixels.

Clickable Buttons

To make a graphic image "clickable" so that it links to another document, simply include the image tag inside the bounding tags of an HTML anchor that points to that document's URL:

```
<a href="documentToGoTo.html">
<img src="greenButton.gif" border="0">
</a>
```

You can also use the **<A> tag** to provide a link to a larger graphic or even to a video clip from a small, thumbnail-sized image:

```
<a href="bigPicture.jpg"><img src="thumbnail.gif" border="0"></a>
```

Be sure to include the **BORDER attribute** (border="0") in the <IMG> tag if you wish to avoid showing a border around the button image (sometimes an ugly two blue pixels wide).

Client-Side Image Maps

Image maps are pictures with defined hot spots that link to other documents when a user clicks on them. Browsers support client-side image maps so that mouse coordinates and their associated document URLs can be included in an HTML document. This is managed by the **<MAP> tag** and the **USEMAP attribute** of the <IMG> tag.

To make a client-side image map with USEMAP, you need three things: an image, a list of coordinates designating hot spots on the image, and the document URL associated with each hot spot. To program the image map into your HTML document, you use the USEMAP attribute

of the <IMG> tag. Here is the HTML code for the navigation button in Figure 13-3 and detailed in Figure 13-10:

```
<img src="compas.gif" hspace="5" vspace="50" border="0" usemap="#compass">
<MAP name="compass">
  <area shape="circle" coords="60,60,10" href="help.htm">
  <area shape="polygon" coords="60,60,0,0,120,0" href="back.htm">
  <area shape="polygon" coords="60,60,0,120,120,120" href="forward.htm">
  <area shape="polygon" coords="60,60,0,0,0,120" href="navmap.htm">
</map>
```

Compas.gif is the transparent image, the hspace="5" and vspace="50" attributes provide space between the image and the text around it, and the border="0" attribute makes the image borderless. The usemap="#compass" attribute points to the <map> extension tag that contains the coordinates and URLs. (The pound sign means the <MAP> tag is located in this same document.) A <MAP> segment may be placed anywhere in the body of the HTML document and is related to the correct image by the name="xxxxxxx" attribute of the <MAP> tag. You can have more than one image map in an HTML document, but they must have different names.

Within the <MAP> tag, the <AREA> tag defines the shape of the hot spot (as a circle, polygon, or rectangle) and anchors or links it to a URL. Areas are defined by x,y coordinates of the pixels in your bitmap: a circle by the x,y coordinates of its center location and radius (60,60,10), a polygon by a sequence of sets of x,y locations that close automatically (60,60,0,0,120,0 defines a triangle), or a rectangle (two x,y locations defining top left and bottom right).

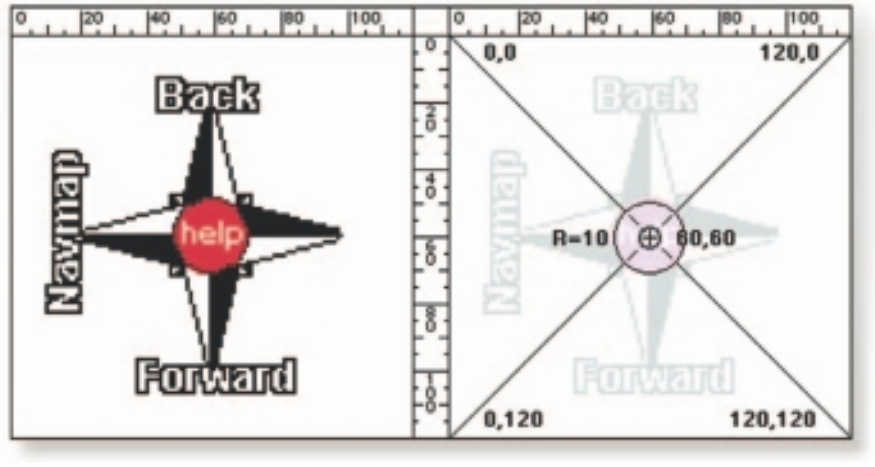
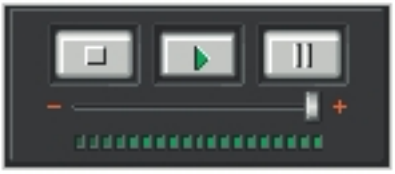


Figure 13-10 This enlarged image illustrates the coordinates used to define hot spots for image maps (the ruler is marked in pixels).

Sound for the Web

In the beginning, when the Internet was primarily a collection of Unix machines, sound files were sent from machine to machine in **AU format** and, when downloaded, were played back using a sound application. As the Web has developed, sound has become more important, and most browsers allow embedding of sounds into documents using the <AUDIO> tag. Inside this tag, the **autoplay** attribute, if present, starts the audio playing as soon as it is ready. If **controls** is present, a play/pause and other controls will be displayed. When **preload** is present, the audio will load when the page does and be ready to run. Text can be included in the tag that will be ignored unless the user's browser cannot understand the <AUDIO> tag:



```
<audio src="LizLaugh.aiff" preload autoplay controls loop>
Sorry, your browser does not support the HTML audio element.
</audio>
```

The <A> anchor tag and <EMBED> tag can also be used to play sound files:

```
Click <a href="LizLaugh.aiff">here</a> to play sound file.
<embed src="Mozart.mid" autostart="true" loop="false" width="120" height="50" hidden></embed>
```

Chapter 4 describes designing and making MIDI and digitized sound files in detail.

TIP *Making sound for the Web requires the basic tools and techniques described in Chapter 4. Always nibble at your sound elements and reduce them to the lowest file sizes that will play acceptably. Remember, they will move across the Internet and may be downloaded or played on machines with low-bandwidth connections.*

Animation for the Web

HTML makes no provision for animation, by itself delivering only a static page of text and graphics. Boring, many people said, and programmers went to work devising methods to liven up the view. JavaScript can dynamically change a web page without needing to reload it. JavaScript with XML features, combined into **Ajax** (Asynchronous JavaScript and XML), is used for powerful interactive applications such as Google's "Office." The Flash plug-in for browsers offers animation and interaction.

GIF89a

Browsers implement a little-known animation feature in the final 1989 revision "a" of the GIF file format specification. It is possible to make

simple animations by putting multiple images, or frames, into a single GIF89a file and display them with programmable delays (in 100ths of a second) between them.

When you use the <IMG> tag to embed a GIF89a **multiframe image**, the browser downloads the file and stores it in the cache folder of your local hard disk. Once fully downloaded, the browser plays each frame of the image quickly and smoothly. Limit animated GIFs to small images, and use a more capable plug-in like Flash for animations over larger areas.

Read Chapter 5 to learn the basics of animation. Pick a tool or method and start creating. Lokki, the Shockwaved seagull, was created by a beginner and was flying in just a few hours:

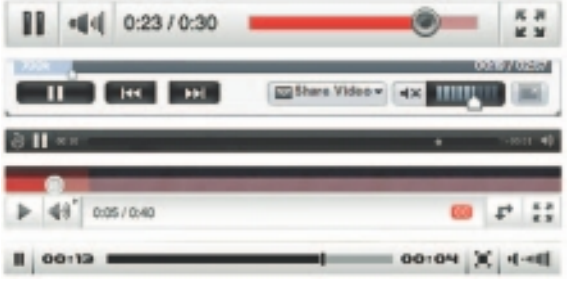
Animation software includes Swish (www.swishzone.com), Flash, Director, After Effects, DHTML, and animated GIF files built using shareware and freeware. Designers must be careful how they use animation though: too much motion and too many flashy colors can cheapen a web site. Subtle animation, however, enhances a site's content and messages.

http://webstyleguide.com/wsg2/multimedia/animation.html
http://website.needbeyond.com/templates2.html?flash_intro
www.animationtipsandtricks.com
www.compuphase.com/animtips.htm

For animation styles and tips

Video for the Web

In the past (and still today), playing video on a web page requires special plug-ins like Flash, QuickTime, RealVideo, Windows Media Player, or other proprietary software. These plug-ins use many different **codecs** (compressor/decompressors) and many different streaming and storage container file formats, each with its own interface and custom options, as shown here.



To reduce reliance on these plug-ins and offer a standard method for preparing and delivering video to the Web, the HTML5 specification provides a <VIDEO> tag, meaning that HTML5-capable browsers such as Internet Explorer

9, Firefox 3.5, Safari 4, Chrome, and Opera must contain within themselves the programming code required to recognize a video file, read and decompress both its audio and video components, and play that video on a screen—where and how you, as the designer, specify.

```
<video src="myVideo.mpg" preload autoplay controls width="320" height="240">
Sorry, your browser does not support the HTML video element.
</video>
```

There are more than 250 file formats that contain video elements (see www.fileinfo.com/filetypes/video), and there are more than 25 codecs from which to choose. And there are many methods and options to capture, compress, edit, store, and distribute video. Introduction of the HTML5 <VIDEO> tag is a push in the direction of a standardized few technologies and methodologies that will work for most everyone on the Internet.

The most commonly used codecs are H.264, Theora, and VP8 within MP4, Ogg, and WebM containers. Unfortunately, no one of these will necessarily play in every HTML5-compliant browser. To guarantee playability by all browsers, you may need to encode four separate versions of your video file, including a Flash .flv format as a fallback, and program your HTML <VIDEO> tag with all four. The browser will play the first file in the list that it can:

```
<video width="160" height="120" controls autoplay>
<source src="myVideo.mp4" type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>
<source src="myVideo.webm" type='video/webm; codecs="vp8, vorbis"'>
<source src="myVideo.ogv" type='video/ogg; codecs="theora, vorbis"'>
<object type="application/x-shockwave-flash" width="160" height="120" wmode="transparent"
data="flvplayer.swf?file=myVideo.flv">
    <param name="movie" value="flvplayer.swf?file=myVideo.flv" />
    <param name="wmode" value="transparent" />
</object>
</video>
```

For more about making and editing video files, codecs, and distribution methods, see Chapter 6.

Plug-ins and Players

Prior to Adobe's acquisition, when Macromedia introduced Shockwave to allow the animation and interactivity of its flagship tool Director to be embedded into pages viewed, real animation and programmable power became available to web page developers. Later, they added Flash to their animation armory, which also uses Shockwave to create an .swf (Shockwave Flash) version of the native .fla file in order to make it displayable on a web page. **Players** and **plug-ins** became available for other multimedia tools with animation capabilities (for example, RunRev), and the

view came alive as long as the person viewing your page had installed the necessary plug-in on his or her machine. The QuickTime movie format includes the ability to create Virtual Reality (VR) files, also displayed on a web page via a player. Flash and proprietary viewers can be used to present panoramas. Figure 13-11 shows a real estate sales panorama—when you drag the mouse across this player's window, the scene tracks and rotates in a 360-degree panorama. You can see adjacent rooms, too, by panning the image in a circle.

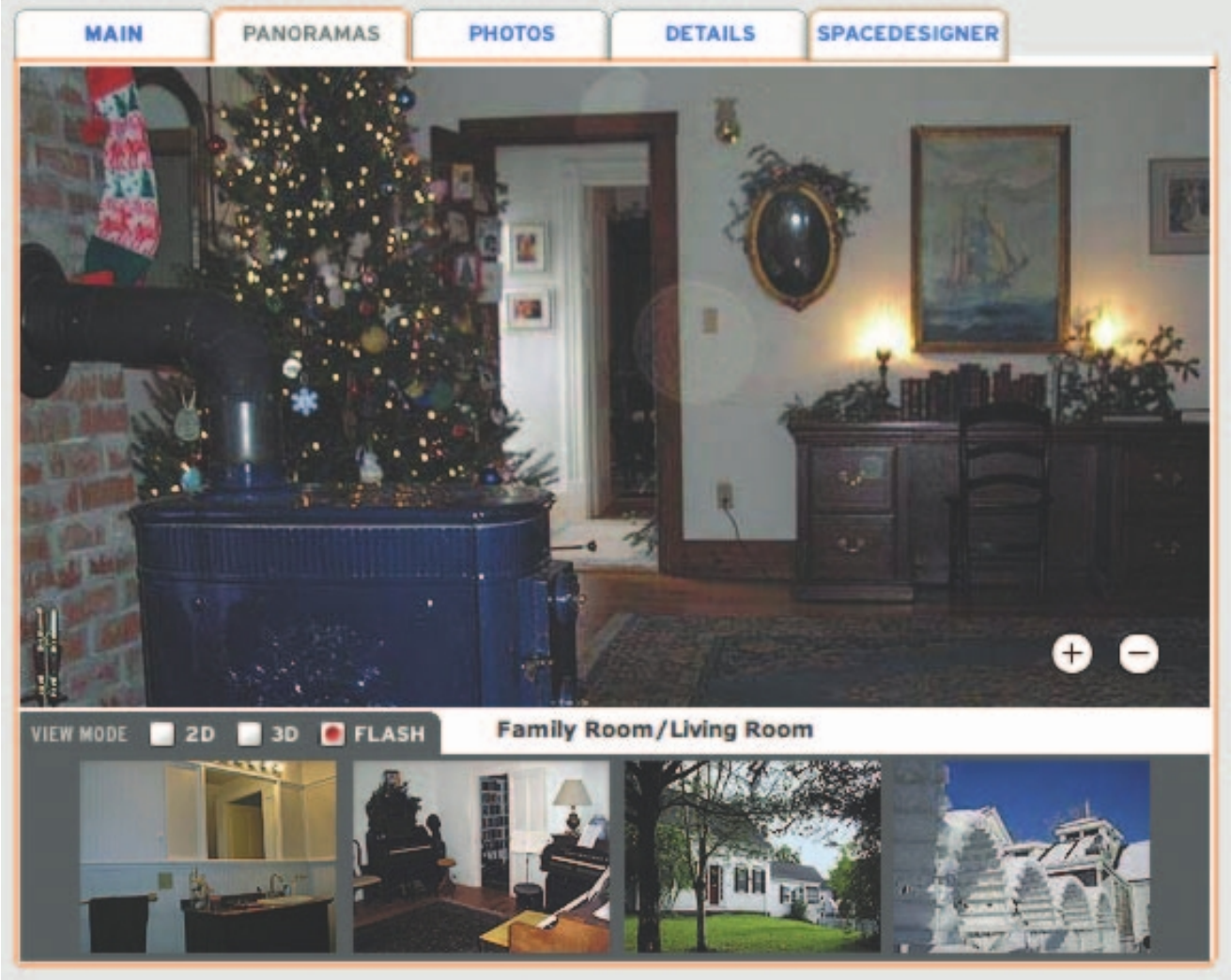


Figure 13-11 Useful multimedia tools can enhance commercial web sites.