CHAPTER 1 INTRODUCTION TO HTML AND JAVASCRIPT

- Tools You Need to Get Started
- What You Need to Know to Get Started

JavaScript the Language

Good news! You are going to love JavaScript. Why? Because, except for HTML, it's the easiest thing a state-of-the-art programmer can learn. If you have been around programming languages for a while, you are very aware that high-level language compilers have exploded from their single floppy editions (i.e., Borland Inter-national's original Turbo Pascal) to 200+-megabyte monsters like Microsoft's Visual C++.

Programming fundamentals also have escalated from simple procedural, top-down, structured programming to object-oriented, event-driven, multitasking, multimedia, multimigraine formulas. Unfortunately, you cannot increase your mental storage capacity as easily as adding a terabyte hard disk! Well, good news. JavaScript is easy to master and is fun and powerful.

Introduction

If you are reading this, then you have already discovered that HTML provides little more than sophisticated Web page formatting. While HTML pages can be as exciting to view as today's state-of-the-art $3-million Hollywood production, in all honesty, to a serious programmer, they are little more than monitor-sized flashy billboards.

You have purchased this book because you need to go beyond a "pretty face" and need to have your Web page meaningfully interact with the user.

JavaScript allows you to use Web page accessories such as counters, catalogs, calendars, maps, scrolling messages, and games. With Java-Script you can control your system's audio, video, and virtual reality plug-ins. By the time you finish this book, you'll have a thorough understanding of the JavaScript language and how to design effective scripts for input validation, forms, searches, calculations, games, and more.

Plug-ins

Allow program modules to dynamically extend the capabilities of most state-of-the-art browsers. These extensions include the handling of new data types and information, along with Java and JavaScript that facilitate actual Web page programming.

Tools You Need to Get Started

You basically need two tools to begin writing and testing JavaScript programs. The first is a good source code editor, such as Microsoft WordPad. The second is a copy of the latest version of Netscape or Internet Explorer. The latter is necessary in order for you to develop and test program code. If you need to download a current version of Netscape, you can get it from Netscape's home page at http://home.netscape.com, or from Netscape's FTP (File Transfer Protocol) server at ftp://ftp.netscape.com. For the latest at Microsoft, surf on over to http://www.microsoft.com.

What You Need to Know to Get Started

If one were to equate HTML (HyperText Markup Language) to the shell of an automobile, then JavaScript would be the vehicle's engine. One without the other is useless. While this text is not a comprehensive manual on HTML, Chapters 1 through 13 discuss the fundamental HTML terms and methodology necessary to author in-depth Web pages with sophisticated front ends. Chapters 10 through 23 will then teach you how to add programmable functionality.

JavaScript Ancestry

JavaScript is not a totally new programming language but instead could be considered a dialect, just as C++ is not a totally new language but an evolution of C. JavaScript's ancestry begins at Netscape, which originally developed a language called LiveScript. LiveScript provided a basic scripting capability to both Netscape Navigator and Internet Explorer.

Starting With Netscape 2.0, LiveScript was given the new name JavaScript.

JavaScript supports both Web browser and Web server scripting. Browser scripts are used to create dynamic Web pages that are more interactive, more responsive, and more tightly integrated with plug-ins and Java applets. JavaScript supports these features by providing special programming capabilities, such as the ability to dynamically generate HTML and to define custom event-handling functions.

You incorporate a JavaScript into an HTML document via the HTML <SCRIPT> tag element. When a JavaScript-enabled browser loads an HTML document containing scripts, it evaluates the scripts as they are encountered. The scripts may be used to create HTML elements that are added to the displayed document or to define functions, called event handlers, that respond to user actions, such as mouse clicks and keyboard entries. JavaScript also controls Java applets and plug-ins.

From a Web server viewpoint, JavaScripts can process form data, perform database searches, and implement custom Web applications. Server-side JavaScripts are more tightly integrated with the Web server than traditional Common Graphical Interface (CGI) programs.

JavaScript the Language

In late November 1995, Netscape's LiveScript officially became JavaScript in a joint announcement by Netscape and Sun Microsystems, Inc. With such a symbiotic relationship, there is little doubt that JavaScript will continue to resemble Java and that it will become the choice tool for gluing Java applets into Web pages (at least from Netscape's point of view).

If you compare the number of Netscape Navigator (Internet Explorer) users versus those Navigator users who actually write JavaScripts, you'll soon discover that JavaScript is one of the least-used components of Netscape's suite of interactivity tools. Yet the ease of using JavaScript, by simply inserting the appropriate code into an HTML document with the benefits of quick

access to browser properties, makes for an enticing programming scenario.

Navigator or Internet Explorer can report back to a JavaScript meaningful information such as history lists, currently loaded documents, frames, forms, and links to the programmer. Capturing user events such as changing form values or pointing at links are second nature for JavaScript.

Unlike many current state-of-the-art languages, such as C and C++ which are compiled languages, and the unique niche created by the Java language, which is both compiled and interpreted, JavaScript is an interpreted language. This means that your JavaScript source code is executed directly at runtime. Interpreted languages offer several advantages as well as several disadvantages.

Interpreted languages are generally simpler than compiled languages and are easy to learn. Frequently, it is easier to develop, change, and trouble-shoot programs because the need to recompile with each change is removed. On the down side, the need to interpret commands as the program is run can produce a performance hit with some interpreted languages. JavaScripts are precompiled to bytecode format, similar to Java source code, as the script is downloaded and interpreted.

JavaScript is similar in syntax, basic structure, flow constructs, and security validation checking. Note however, that Netscape Navigator, Dynamic HTML 4.0, and JavaScript are all under intense development and are constantly evolving. For all intents and purposes, this means you are trying to hit a moving target.

What Is a Scripting Language?

Scripting languages have been around for a long time, predating the Web. UNIX used scripts to perform repetitive system administration tasks and to automate many tasks for less computer-literate users. Scripting languages are the fundamental building block for much of the CGI programming that is currently used to add a limited form of interactivity to Web pages (for example the

Perl language). JavaScript makes it possible to program responses to user events such as mouse clicks and data entry in forms because of the way in which JavaScript is integrated into the browser and its direct interaction with HTML documents.

JavaScript Is Object-Oriented

The simplest way to define an object to a procedural programmer is to say that an object is a syntactic bundling of data members with functions that work exclusively on those data members. Therefore, when you pass "data" around, you also pass the "horsepower" that works on that data, all in one neat tidy code package. Note, however, that the object doesn't give you any additional capabilities, just a new protected syntax. And so that you understand object-oriented terminology, in a procedural language one would say, "I'm declaring a variable," but in an object-oriented language one would say, "I'm instantiating an object." Also, an object-oriented programmer uses the term `method` or member function to distinguish a subroutine syntactically bundled to a specific object, from regular standalone functions not specific to any object.

JavaScript provides several built-in objects that report back information about currently loaded Web pages and contents. A sample object hierarchy would look like Window, Location, History, Document-Forms/Anchors. Other built-in objects, such as the strings Math and Date, allow you to manipulate strings, perform trigonometric calculations, and retrieve current date information.

JavaScript Is Platform-Independent

Since you have no way of knowing an end-user's system configuration, any program you write, distributed across the Web, must necessarily be hardware- or platform-independent, in order to execute properly. JavaScript is platform-independent. The same program code can be used on any platform for which Netscape Navigator or Internet Explorer is available.

Source Code Plagiarism

From a code security point of view, the one downside to JavaScript is that there is no code security! Since an entire JavaScript is embedded within an HTML document, it is visible to all who care to view. This means that any signature-specific algorithm you design necessarily becomes public domain.

Quick Get Me Started - What Is HTML?

Since JavaScripts are hosted by HTML documents, it is necessary for you to understand HTML. The beginning chapters of this book review the necessary fundamentals. If you are already thoroughly familiar with HTML tag elements, you can skip directly to Part Three, "JavaScript Fundamentals." Note, however, that HTML is also a moving target. By skipping Parts One and Two you may inadvertently miss discussions of the newer dynamic HTML 4.0 features.

From IBM to grandmothers, everybody's getting into WWW (World Wide Web, a hypertext-based system of presenting information over the Internet) page development, the visually exciting

way to say to the world, "I've arrived!" Undoubtedly the most exciting aspect of Web page construction is the easy to learn protocol of HTML. But let's not put the cart before the horse. HTML has a very interesting history.

HTML Ancestry

You see, it all began at the high-energy physics laboratory in Geneva, Switzerland, named CERN. The simple problem encountered by the scientists involved the time delay in disseminating research papers and other documents. And this time delay wasn't restricted to the nucleus of buildings on the CERN campus, since their vital statistics were shared throughout the world. It is Tim Berners-Lee who is credited with designing the system that would allow scientists to share fairly complex materials easily, using a simple set of protocols over the Internet (the term used to describe all the world-wide interconnected TCP/IP networks).

Tim broke his solution down into two parts: HTTP, the HyperText Transfer Protocol definition, which provided a simple way for users to request and receives files over the Internet, and the more familiar HTML or HyperText Markup Language. Unlike HTTP, which defines how information is sent or received, HTML defines the visual presentation of the material on the receiving end.

TCP/IP

An abbreviation for Transport Control Protocol/Internet Protocol, a set of protocols that applications use for communicating across networks or over the Internet. These protocols specify how packets of data should be constructed, addressed, checked for errors, and so on.

Needless to say, as originally designed, HTML was never intended to be used for the variety of display potentials presented by today's multitasking object-oriented operating systems like UNIX and Microsoft Windows. Nor was it ever designed to create wild multimedia sites that incorporated graphics and animation. The fledgling Internet was seen more as a library than as a Virtual Reality Mall. As such, the original definition of HTML included as much output display control as would be needed by the typical scientific journal article.

Because HTML's protocols were succinct and complete, they were immediately accepted by the scientific community which adopted it as its electronic typesetter. Scientists were particularly excited about HTML's ability to create links to other pages of information, making the documents much more alive than a static piece of paper. Unfortunately, this forward-only hotlink capability left something to be desired.

SGML

Actually, HTML is a subset of an even larger page display protocol definition known as SGML, or Standard Generalized Markup Language. The scientists at CERN used SGML for highly technical and legal documentation. SGML is still used by large organizations that maintain libraries of frequently referenced documentation.

SGML flavoring lingers to this day within the formal HTML standard with its insistence on document structure, the separation of content from formatting, and the use of logical tags (tags are the special control characters that separate HTML markup from ordinary text, namely the left and right angle brackets: < >).

DTD

All documents that can be marked up with the same hierarchy of elements are said to belong to a certain document type. Rather than describe a set of tools to mark up documents, SGML defines the structure of a particular type of document in what is called

Document Type Definition, or DTD. HTML 4.0, code-named Cougar, is an example of an SGML DTD.

SGML in HTML

The most prominent throwbacks to SGML are the <H1> through <H6> tags used to generate a six-level outline format. Today's users employ these tags as a quick way to format headings with varying degrees of emphasis. Some tags like <B> for bold and <EM> for emphasis may appear to be two different ways of saying the same thing, and indeed some browsers treat them as such. However, technically, <B> is font-specific, whereas <EM> provides more leeway in formatting the document.

Browsers

> Programs that give you the ability to browse the World Wide Web such as Netscape and Internet Explorer. Some browsers offer additional functionality, such as FTP and e-mail support.

With the explosion in popularity of the Internet, the browser market skyrocketed in 1995 and so too did the availability of new HTML tags. However, the SGML throwbacks are still the quickest and easiest way to create tables of contents and indices automatically.

Realizing the tremendous profit potential of this new electronic media, Netscape and Microsoft locked into a perpetual battle over trying to give their customers the de facto standard in Web browser capabilities. Fortunately the W3C (World Wide Web Consortium—http://www.w3.org/—which technically oversees HTML revisions) stepped in to maintain some sort of compatibility across platforms. One of the exciting protocols to come out of these meetings is the cascading style sheet. Cascading style sheets make it possible to specify particular fonts, point sizes, and text placement for a wider variety of HTML elements.

What Is an Element?

An element is any HTML tag or tag pair that is parsed and treated as a unique item on a page. An element differs from an object in that an element is an object only if it is named with an ID attribute. Some examples of elements are headers, paragraphs, lists of items, or tables. Elements can contain plain text, or other elements, or both.

More on Tags

HTML tags, as mentioned earlier, mark the elements of a file for your browser. Tags are usually paired, for example <H1> for a level-one header definition followed by a closing </H1> end tag instruction. End tags look just like beginning tags except for the addition of a forward slash / symbol immediately after the left angle bracket <.

Some elements may include an attribute, which is additional information that is included inside the start tag. For example, files in GIF or JPEG format can use top-, middle-, or bottom-edge alignment by including the appropriate attribute with the image source HTML code. One final note. HTML tags are not case sensitive so <H1> = <h1> (well most of the time—the few exceptions will be documented throughout the book).

What Is an HTML Document?

Technically, an HTML document is nothing more than a combination of HTML tags placed inside a standard ASCII text file. This means that you can create an HTML file using Windows Notepad, or Emacs or vi on UNIX architectures, and even BBEdit on a Macintosh plat-

form. If you are running the latest editions of many popular software suites such as Corel Office, or Microsoft Professional, then you own a WYSIWYG (What You See Is What You Get) word processor capable of generating and visually displaying HTML documents!

So If It's So Easy, What's the Catch?

There's only one snag to your actually taking a completed HTML document and flashing it on every monitor from here to Moscow—money (unless you are a student). You see, you need someone (a host server, and for students, well, let the institution take it out of your tuition) willing to store your multimedia punched front page, willing to pay for the electricity running the system, willing to pay for the hardware connection to the Internet, and willing to pay a tech support person to keep everything static free and running clean.

The great news is that some communities operate what is called FreeNet, a community-based network that provides free Internet access. If all else fails, well, you'll have to subscribe to a service such as AOL (America Online) or a local service provider. These services usually charge two separate rates, one rate for personal and a separate rate for business Web page accounts.

OK, Here's a Quick HTML Document

Whether you are using Assembly Language, Ada, Java, Perl, or C++, every language, beyond its formal description, has an industrywide acceptable list of language-specific dos and don'ts—

HTML is no exception. Even a minimal HTML document should contain certain standard HTML tags, mainly a head and body text. The head contains the title, while the body encapsulates the actual text to be displayed. The body can contain multiple paragraphs, lists, tables, graphics, and other elements.

Browsers, similar to compilers, expect an HTML document to be in a certain form. This standard form comes directly from the SGML and its sibling HTML specifications. The simplest, properly formed HTML document you can create looks very similar to the following code listing:

```
<HTML>
<HEAD>
<TITLE>My First HTML Document</TITLE>
</HEAD>
<BODY>
<H1>Hello World!</H1>
<P>This is my first paragraph letting the world
know that I am ready to begin my cottage industry.
<P>OH, and make my first MILLION!
</BODY>
</HTML>
```

<HTML>

The first tag in the listing, <HTML>, flags the browser to the fact that it is parsing an HTML encoded file format. HTML files always contain an *.html file extension (or for systems using the eight-letter filename, three-letter file extension format, *.htm extension).

<HEAD>

The <HEAD> tag identifies the first part of your HTML encoded document that contains the document's title. The title is shown as part of your browser's window.

<TITLE>

The <TITLE> tag defines your document's title and broadcasts its content in a global context. Your browser displays the title in an area separate from the document text. Choosing your document's <TITLE> is a little more important than it may initially appear since it is this line of text that is displayed on someone's hotlist or bookmark list. It is your document's title that is referenced when using search engines. For this reason, you should always choose a <TITLE> that is unique, descriptive, and pulls those Web surfers right down onto your Web page.

<BODY>

The largest portion of your HTML document is encapsulated within the <BODY>, </BODY> tag pairs and contains the information displayed by the browser. The following tags are used within the <BODY>, </BODY> tags:

<H1> - LEVEL HEADER

As mentioned earlier, HTML has a six-level throwback to its SGML proginator. These tags are <H1> through <H6> and </H1> through </H6>, respectively. Each level header is displayed using a larger and/or bolder font than is used by the browser for displaying normal text.

<P> - PARAGRAPH

Browsers ignore carriage returns embedded within the <BODY> of an HTML document. This means that you do not have to concern yourself with how long each of your text lines is since any text formatting you want must be accomplished with an appropriate HTML tag. So, for example, the two lines in the source file

```
<P>This is my first paragraph letting the world
know that I am ready to begin my cottage industry.
```

are seen as one continuous line and will be formatted based on the browser's current screen width. From the browser's point of view, the source file could have been entered as follows:

```
<HTML> <HEAD> <TITLE>My First HTML Document</TITLE>
</HEAD> <BODY> <H1>Hello World!</H1> <P>This is my
first paragraph letting the world know that I am ready
to begin my cottage industry. <P>OH, and make my first
MILLION! </BODY> </HTML>
```

with identical visual display interpretations. Browsers also condense multiple spaces embedded within an HTML document down to a single space.

You may be wondering why the sample HTML document contains no </P> end tags. This is because browsers understand that a start <P> tag indirectly indicates that a previous paragraph is terminating.

Note however, there is a </P> end tag, but it is traditionally used only when the encapsulated paragraph (<P>, </P>) requires a certain attribute, such as centering, as in this reworked example:

```
<ALIGN=CENTER>
<P>This is my first paragraph letting the world
know that I am ready to begin my cottage industry.
</P>
```

A Word about HTML Document Formatting

Since browsers ignore carriage returns and extra spaces, you can use these to visually format your HTML document. For example, you should separate headings from their associated bodies with a blank line. Use extra blank lines to separate document sections. All of this will go a long way to streamlining any edits or modifications the document may need over time.

Some Common File Types and Their Extensions

Web pages are never going to be the same with the spectacular multimedia enhancements possible with HTML 4.0. Throughout this book you will learn how to add images in their various formats, sounds, even animation. Table 1.1 enumerates these file types by name and file extension.

Table 1.1: Common File Types and Their Extensions

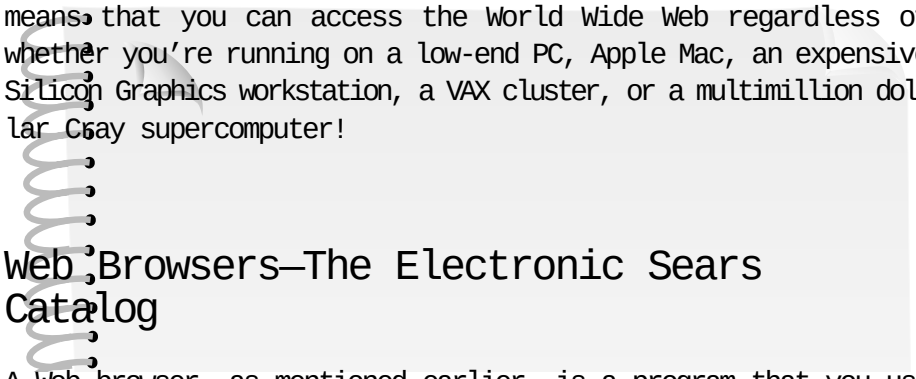
File Type	File Extension
Plain ASCII text	.txt
HTML document	.html or for 8.3 formats, .htm
GIF image	.gif
TIFF image	.tiff or for 8.3 formats, .tif
X Bitmap image	.xbm
JPEG image	.jpeg or for 8.3 formats, .jpg
PostScript file	.ps
AIFF sound file	.aiff or for 8.3 formats, .aif
AU sound file	.au
WAV sound file	.wav
QuickTime movie	.mov
MPEG movie	.mpeg or for 8.3 formats, .mpg

Quick Tell Me—How Does the Web Work?

The World Wide Web (WWW) can be viewed in three parts. The first is the server that is hosting (storing/connecting to the Internet) your HTML document. The second is the surfer or end user viewing your HTML document. The third is the protocol making the bidirectional communication possible.

It Allows Every Type of Computer World Access!

But most importantly, the Web is platform-independent. This means that you can access the World Wide Web regardless of whether you're running on a low-end PC, Apple Mac, an expensive Silicon Graphics workstation, a VAX cluster, or a multimillion dollar Cray supercomputer!



Web Browsers—The Electronic Sears Catalog

A Web browser, as mentioned earlier, is a program that you use to view pages on the World Wide Web, sometimes called *Web clients*. A vast diversity of Web browsers are available for just about every type of architecture you can imagine, most importantly Graphical-User-Interface–based systems or GUI systems such as X11, Windows, and Mac platforms. There are even text-only browsers available for simple dial-up UNIX connections.

Full-Color Shopping at Your Fingertips

One of the key features of Web browsers is their ability to display both text and graphics in full color on the same page, and all of this with a simple URL address followed very often by nothing more than consecutive mouse clicks. If you are just jumping onto your Internet surfboard for the first time, you may not be aware that in its fledgling state, the Internet was accessed by nonstandard, confusing, command-line, text-only protocols. Of course today's state-of-the-art rendition incorporates sound, and even streaming video. Even 3D virtual reality simulations are possible with VRML, Virtual Reality Markup Language.

VRML

You can find a very interesting World Wide Web Virtual Library at www.w3.org/pub/DataSources/by Subject/Overview.html/, supported by individuals interested in promoting/sharing information on this extremely exciting outgrowth of HTML.

Info, Info Everywhere

Of course the very name, World Wide Web, indicates that the information you are downloading is distributed throughout the entire globe. Since the information you are accessing occupies vast amounts of disk storage, particularly when you include images, multimedia, and streaming video, there hasn't been a computer built to date that could house this bit explosion in one physical location.

Actually, this distributed diversity of data storage repositories is to your advantage. Were this information stored in a single location, imagine the chaos generated by a downed mainframe! The Web is successful in providing so much information because that information is distributed globally across thousands of Web sites. And the best part about the interconnection is that if one leg of the information route is interrupted, for any reason, an alternate Web link takes over.

Provides Full Bidirectional Communication

An exciting aspect to Web interaction is its provisions for you to "talk back." Take, for example, a radio or television broadcast. This is one-directional output.

The exciting news is that with today's evolution of HTML your document's display instructions are not limited to text only, but now include graphical and auditory elements.

Web Browser Characteristics

URL

A Universal Resource Locator consists of a protocol name, a colon :, two forward slash characters //, a machine name, and a path to a resource using a single forward slash as a separator. URLs can also specify more than just Web page addresses. For example, you can retrieve a document by preceding the URL with ftp://, or File Transfer Protocol, instead of http://,

MIME

Designed as a means for embedding complex binary documents within an e-mail message. Browsers take the MIME protocol deciphering the document's type and subtype. At this point the browser decides how it wishes to handle the document type. It may choose to process it internally, or invoke an external program to decipher the information. MIME types consist of a main type and subtype. For example, plain text

While there is only one HTML 4.0 standard, it is not true that there is only one visual interpretation of this standard among all the various browsers. Each Web browser displays HTML elements differently. Matters degrade even more when you find out that not all browsers support all of the HTML 4.0 code standards!

Fortunately, any HTML code a browser does not understand, it usually just ignores.

The unwarned can spend endless hours, even days or weeks, perfecting and tweaking the visual impression and technical content of their company's Web page, only to find it look like a mosaic on an unfriendly browser. To quote the NCSA General Internet HTML-PrimerAll.html document:

> Hence these words of advice: code your files using correct HTML. Leave the interpreting to the browsers and hope for the best.

What Is a URL?

When a Web surfer, or end user, or client (the system being used by the Web surfer) makes a request to a server, it uses the HTTP protocol, mentioned earlier, across the network to request the information in the form of a URL (Universal Resource Locator—URLs specify the location and name of a World Wide Web resource such as a Web site or an HTML document) from the server.

The server in turn processes the request and, again, using HTTP protocol, transfers the requested information back to the client. It is the server's responsibility to tell the client the type of document being transmitted. This is usually defined as a MIME (Multipurpose Internet Mail Extension), an enhancement to Internet e-mail that allows for the inclusion of binary data such as word processing programs, graphics, and sound. The client must then process the information before it splashes on the surfer's screen.

Some aspects of the displayed document are fixed, such as titles, paragraphs, lists, and so on, but there are components of the document which are considered live or dynamic. These dynamic elements display more current information, for example, the ever-present "hit counter" telling you how many other Web surfers have surfed this page before you.

What Is Scripting?

At this point, now that you have a general overview of HTML and browser capabilities, you might just be asking yourself if there's any more to this whole thing than just displaying formatted text and multimedia images. Well, the answer is yes. Welcome to the world of scripting.

Scripting

> The process of writing a script. Scripts are executed by Web servers, performing functions such as HTML document searches and form-filling functions.

CGI

One of the earliest scripting languages was CGI, or Common Gateway Interface. Its most common application is in forms processing. CGI allows you to create pages for users as individual requests come in, and you can customize pages to match that information. The user usually fills out a form and clicks on a submit button. Then the user's browser sends to the server a request that includes the information the user entered into the form. The server then sends this information on to another program for actual processing and responds with the appropriate output at the client or user end. Actually, depending on the kind of server your site resides on, you can write CGI programs in C++, Perl, even AppleScript. If you don't want to go to the trouble of writing complex programs to create pages with server-side interaction, you can purchase products like Microsoft's FrontPage Web site editing program. It incorporates bots, which are small programs that attach themselves to the host server. Bots can pro-

cess forms, provide timestamps, and streamline site-specific search engines.

SCRIPT Tags

HTML page programming takes place via SCRIPT tags which tell script-aware browsers that a script will follow. Within these SCRIPT tags are HTML tags. This approach can be extended to do far more complex things, such as creating cookies. Cookies store small amounts of information about a user's preferences on a user's machine and use that information to create customized pages on subsequent visits. You can customize most aspects of Web pages and even the browsers.

Perl

Perl is undoubtedly the most common language used for scripting CGI in UNIX environments with its combination of C syntax and the power of UNIX regular expressions. It is possible to write simple programs in Perl with a minimum of effort.

JavaScript and JScript

Netscape version 2.0 is credited with the introduction of JavaScript. Immediately, Microsoft Internet Explorer 3.0 countered with its own flavors called JScript and VBScript based on the easy-to-learn Visual Basic. The good news is that JavaScript and JScript are evolving toward one another; however various browsers still respond in nonuniform fashion.

These languages provide HTML developers with additional programming horsepower that enables them to make browsers do new and different things. Not everything has to take place on the server end; now the client can take on more of the responsibility of processing.

VBScript

Offering Visual Basic programmers the programming enhancements of a scripting language, VBScript came bundled with Microsoft's Internet Explorer 3.0. While JavaScript and JScript have a very C- or Java-based flavor, VBScript offers Visual Basic programmers the familiarity of their popular language. VBScript also easily integrates Microsoft's ActiveX controls in a Web environment.

Plug-Ins and ActiveX

Netscape is credited with developing the first plug-ins, Microsoft with developing ActiveX controls. The idea behind both is that the controlling software is loaded onto the user's computer, and then the Web page contains another file that contains the specific instructions or content.

While there are significant structural differences between plug-ins and ActiveX controls, their purpose on the page is basically the same. Like a Java applet, they add additional features and functionality to a Web page without directly affecting the host page. They also create a bidirectional communication between the end user and the plug-in.

The downside to both technologies is they add to the download time of a Web page. In addition to the time it takes to download and install the actual plug-in or control, there's also the extra time to download the content files. In addition, neither technology really provides interaction with other elements on the page. There are some ActiveX controls that provide features like tooltips or pop-up menus, but, like plug-ins, these items are operated directly by the control with no ability to go beyond the feature itself.

What's New in HTML 4.0

Code-named Cougar, the W3C has once again taken on the task of setting the standard that will divide the markup of content from the appearance of a Web page. The Cougar protocol separates the physical style from content markup by more reliance on style sheets (style sheets specify style information whose parameters will govern the formatting of an entire document).

HTML 4.0 is being authored by David Raggett and Arnaud Le Hors, and it continues to recognize frequently used tags from older versions of HTML. If it didn't, we would be looking at all kinds of problems when different browsers accessed pages that exploited its new features. HTML 4.0 isn't really that different from the previous versions of HTML. The key changes can be summed up by the word dynamic. HTML 4.0 will now allow the user to manipulate and access the text and image elements directly. The Web page has become dynamically updatable and the properties are easily accessible from code. This might seem like no big deal, since most operating systems have boasted this for years, but in Web pages it's quite a major undertaking.

The updated standard, however, does not simply incorporate a few new tags and attributes. While most of the tags from previous versions remain, it's this new dynamic ability to manipulate and access elements with a scripting language that forms the unique cornerstone of HTML 4.0. For this reason, both Netscape and Microsoft have christened this new HTML version dynamic HTML.

Both Netscape Communicator and Internet Explorer 4.0 have changed the structure of just how a Web page is displayed by exposing their browsers to these new dynamic HTML capabilities. For example, with advancement of dynamic or movable images, shoppers can now slide pictures of their purchase items into an image of a shopping cart. While these are new features of dynamic HTML language itself, they do, of course, depend upon the browser to display the results. You can't expect the HTML 4.0–specific parts of a page to do much in Navigator 3 or Internet Explorer 3. As with all other new developments in HTML, the browser must provide support before the page can perform as

intended. The following seven features highlight the exciting potential of this latest HTML version:

- Dynamic Fonts—allow you to create your own fonts that others can download, similar to image files. This adds pizzazz to a Web page by making it more eye appealing and unique.
- Absolute Positioning of Elements—allows you to manipulate the x-, y-, and z-coordinates of objects on a page, placing each item where you want it, rather than letting the browser decide.
- Cascading Style Sheets—add even more page format control, are accessible to scripting languages, and allow for a consistent look by applying fonts, colors, and element positioning to multiple pages.
- Canvas Mode—allows the developer to view the HTML-driven page full screen, instead of in the frame of the browser window.
- Dynamic Redraw—allows real-time updating of any element on a page, instead of the time-delayed approach of redrawing the page whenever the user has made a selection.
- New Event-Handling Techniques—event capturing allows events to be passed along from one object to another. It also allows the capturing of events not supported by one object from another.
- Document Object Model—allows the JavaScript language to program the formatting and positioning properties for elements on a page.

HTML—What You Need to Get Started

All you need to create an HTML document is a text editor that is capable of saving files in ASCII format, and a browser such as Netscape's Navigator or Microsoft's Internet Explorer. With a quick click of your mouse and an already established Internet con-

nection, you can download the latest versions of both these browsers to your system via the following addresses:

For Netscape Navigator

http://www.netscape.com

For Internet Explorer 4.0

http://www.microsoft.com

All of the examples and figures for this book were taken using an IBM-compatible PC, running Windows 95, using Windows Notepad text editor for generating the HTML documents. As mentioned earlier however, HTML is a hardware- and platform-independent language. For this reason you can easily create, view, and test, the same HTML documents created throughout the text on a UNIX or Apple Macintosh machine and their associated operating systems.

Should you choose to purchase one of the more popular HTML editors coming onto the market, such as Microsoft's FrontPage, you'll discover a more point-and-click approach to visually creating your Web pages. These programs work much like paint programs only the graphic primitives you place are actually HTML elements and attributes.

While you are graphically designing the page, these products are simultaneously generating the required HTML tags. All of this is fine, as long as the product is designed to do everything you would like from your Web page. However, it is suggested that you stay with a plain text editor for the purposes of this book, since there are some tricks and techniques that even the most powerful HTML editors cannot generate.

Interesting Sites to Visit

www.cern.ch/—The consortium created in 1994 in conjunction with CERN, the European Laboratory for Particle Physics in Switzerland. CERN was the group that gave birth to the Web in 1989.

www.corel.com/products/graphicsandpublishing/—CorelDraw!, a high-end vector program that comes in a bundle by the same name along with Photo-Paint, Dream 3D (a 3D rendering program based on Ray Dream Designer), and Presents (a multimedia presentation program), allows you to export an entire page as HTML.

www.fractal.com/— This site provides excellent painting capabilities such as oil, watercolor, and chalk.

www.cyberdog.apple.com/—This site provides a versatile tool that includes a Web browser, using Apple's OpenDoc technology.

www.hwg.org/—The HTML Writers Guild is an international organization of World Wide Web page authors and Internet publishing professionals. Provides several levels of paid memberships offering benefits such as mentoring programs, educational classes, access to a job board, and software discounts!

www.htmlhelp.com/—This site offers a collection of HTML tips, tricks, and hacks covering both document authoring and Web server management, presented in a question and answer format.

www.browserwatch.iworld.com/— A browser that offers breaking news in the browser and plug-ins industry, plus browser usage statistics and a rich library of plug-ins and ActiveX components.

ActiveX

An integration technology developed by Microsoft to add new features to the Internet

Explorer Web browser. ActiveX allows programmers to extend Web browser functionality with the programmer's own code. If you consider a Web browser as a miniature operating system, you can see why programmers would love to extend a browser's capabilities.

www.nyu.edu/pages/wsn/susir/lynx/platforms.html/— Lynx is a fully featured World Wide Web browser designed for both the UNIX and VMS platforms, using cursor-addressable, character-based terminals or emulators. This is a text-based browser frequently used in universities, libraries, and other environments where affordable access is needed to bring the information of the World Wide Web to as many individuals as possible.

www.developer.netscape.com/index.html/—This Netscape site's DevEdge provides communications, tools, support, and marketing assistance to speed the planning, development, and deployment of Internet and Intranet solutions.

Intranet

An internal network that uses Internet technology. Many companies now describe the network they use for intracompany communications as an Intranet.

www.microsoft.com/sitebuilder/— This site provides access to Microsoft's in-depth resources for developing Web sites using Microsoft technology. Their Site Builder network accesses a multilevel library of technical information, products, technologies, services, and ideas and support for using the latest Internet technology, such as the new dynamic HTML 4.0, ActiveX controls, and Java applets.

VBScript

Developed by Microsoft and similar to the JavaScript Web page scripting language only it is based on what some consider to be the easi-

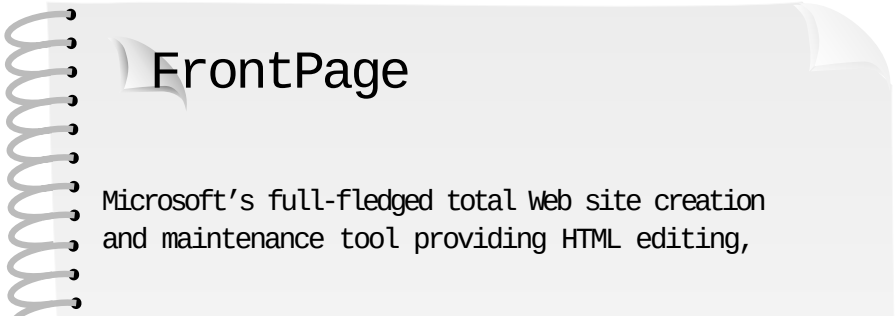
VRML

Virtual Reality Modeling Language is a language that supports the display of 3D objects in HTML

Applet

A program, written in Java, with single or limited function. Many Windows and Win95 special function programs are called applets, and, from that usage, simple Java programs are also called applets. In contrast to an application, which is a program that performs a specific type of work, such as a word processor. Also, applications are standalone products capable of running under an operating system disconnected from the Internet/intranet, while applets are designed to be downloaded by a browser which then executes it.

www.w3.org/pub/www/— Led by Tim Berners-Lee, director of the W3C and creator of the World Wide Web. This platform-inde-



pendent, vendor-neutral consortium works with the global Internet/intranet community to produce freely available specifications and reference software.

www.w3.org/pub/WWW/TR/REC-CSS/—Accesses the document specifying level 1 of the cascading style sheet mechanism, or CSS1.

www.w3.org/pub/WWW/MarkUp/DOM—A site providing an over-view of materials related to the Document Object Model, or DOM.

DOM

A platform-independent, language-neutral interface, allowing programs and scripts to dynamically access and update Web page structure, content, and display style.

WWW.W3.ORG/HYPERTEX/WWW/ADDRESSING/ADDRESSING.HTML— A site dedicated to defining the various types of URLs available, as well as explaining URIs and URNs.