Weave a spell on the Web

The World Wide Web (WWW) is a system which links together the vast amounts of disparate information held on the Internet. The information is stored in the form of pages, and each page may contain text, graphics, audio and even video clips. What's more, the pages are in hypertext format, which allows related information to be linked together. These links can be accessed by the click of a mouse button. To gain access to the WWW, users need a piece of dedicated software known as a Web browser. One of the most popular, and one of the easiest to use and install, is Netscape, but others include Mosaic and Cello.

E-mail is still the most popular of the many Internet resources, but coming up fast into second place has to be the World Wide Web. The development of Web browsers such as Netscape has meant that the Internet is at last becoming easily accessible to all users. And as well as broadening the information superhighway for home users, WWW provides business users with an excellent opportunity for marketing, direct sales and customer feedback. Many corporates are already using it as their electronic shopfront, and setting up your own Web page is a simple and effective way of attracting attention.

HOW DOES THE WEB WORK?

There are three basic ingredients that go into the making of the WWW: HyperText Markup Language (HTML), HyperText Transport Protocol (HTTP) and the Web browser. HTML is the programming language which is used to write the pages that you see on the WWW. Don't worry if you're not a computer programmer, as HTML really only describes the format of the page together with providing the 'hyperlink' information to connect bits of information together. It is similar to making notes in the margin of a page proof to tell a printer that you want certain words in bold, some paragraphs to start with bullet points and where you want the headlines to go.

HTTP is the transport mechanism used to move the WWW information around the Internet as requested. This is not a factor that you, as a Web Weaver (a Web-based information provider), need to worry too much about, and you can leave the company that provides you with WWW storage space to deal with the technicalities of HTTP.

The browser becomes the user's window on to the WWW. Without it, the information could not be accessed in such an attractive and useful way. The browser uses the HTTP mechanism to gather HTML information, upon request, from anywhere on the Internet. This information is then displayed as a Web page, the format of which depends upon the type of browser that's used.

Of course, you need to add other ingredients to get the most out of this particular recipe, not least of which is the ability of the person who is writing the Web pages to communicate information in a clear, precise and attractive way. Which brings us neatly on to our step-by-step guide to becoming a Web weaver.

WEB WEAVING BASICS

The first thing you will need to do with regard to establishing your presence on the WWW is to decide where to site your pages. Although it is perfectly possible to set up a computer as an HTTP server, this isn't a feasible option for most people because of the costs involved. Not only would you need to dedicate a machine, or part of one, to the job, but you would also need it to be connected to the Internet directly and permanently. A much better option is to find an Internet service provider that offers Web storage as part of its service. If you are part of an educational establishment then this may already have such facilities; otherwise you'll need to contact one of the companies listed (see *WWW storage providers*, page 443).

What you get for your money is a set amount of space on the service provider's Web (HTTP) server, and a URL (Universal resource locator – essentially your WWW address) which you can advertise to your clients. It is worth checking with your

The World Wide Web gives access to all the information held by the Internet in a practical and attractive graphical interface. Readers are offered easy access to data through live text links, while businesses or hobbyists can set up their own WWW pages to share information or advertise their wares. Davey Winder shows you how to weave your very own part of the Web
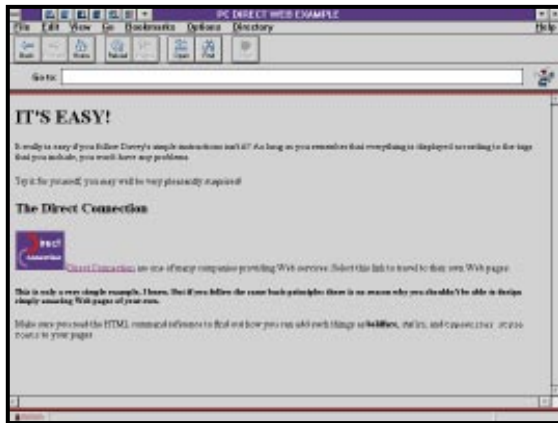
If you follow the commands given in the box below and display the document made via a Web browser, this should be what you will see

current service provider if it offers free Web pages to its members as some do, although the amount of space offered is usually only a page or two.

Once you have found somewhere to put your Web pages, the next step is deciding how to present your information. There are numerous factors that you should bear in mind when designing your pages which will help ensure that you end up with something that is not only pretty to look at but also easy to use, efficient and, above all else, informative.

Much of it comes down to good old-fashioned commonsense, such as the fact that dates are written in many different formats (1/12/95 could be 1 December or 12 January) so use longhand instead of shorthand: 1 Jan 1995 can't be mistaken for anything else.

Think about the structure of your pages. A layout that is composed of logical headings and subheadings is much easier to follow than a disorganised page with words and buttons all over the place. On the WWW 'all over the place' could be exactly that, and nobody wants to have to travel half of cyberspace just to get to the end of your document. A sensible structure will also be consistent so that pages follow the same rules for headings, subheadings, menus and submenus. This makes it a lot easier for the user to follow.

Don't get carried away with it all. In the early days of the WWW, many weavers succumbed to the temptation of filling pages with incredibly large and detailed pictures. This is fine, apart from the fact that the users (your potential customers) have to download all this data on to the screen before they can read it. The bigger the image, the bigger the page, and the more it is going to cost them to download it all. When you need to provide a great deal of information, you can still do so by dividing large documents into smaller hyperlinked ones.

When designing your pages, allow for those who may use different browsers. Although access to the Web is made much easier and more powerful by using a graphical browser such as Netscape or one of the other Mosaic-like programs, not everyone will be using these. Many users access the Web via character-based browsers such as Lynx, which cannot display graphics. These utilities are mainly used through terminal-based Internet providers like Cix, ▶

The *PC Direct* WWW example page

Writing a simple WWW page using HTML commands involves placing tags within Ascii text to format text and add pictures.

Here is a simple HTML document which will create an example *PC Direct* WWW page.

```
<html>
<head>
<title> PC DIRECT WEB EXAMPLE </title>
</head>
<body>
<h1> IT'S EASY! </h1>
```
It really is easy if you follow Davey's simple instructions. As long as you remember that everything is displayed according to the tags that you include, you won't have any problems. `<p>` Try it for yourself, you may be pleasantly surprised.

```
<h2> The Direct Connection </h2>
<img alt="x" align="bottom" src= "./dircon.pcx">
<a href="http://www.dircon.co.uk">Direct
Connection</a>
```
is one of many companies providing Web services. Select this link to travel to its own Web pages.
```
<h5>
```
This is only a very simple example, I know. But if you follow the same basic principles there is no reason why you shouldn't be able to design simply amazing Web pages of your own. `</H5>`
Make sure you read the HTML command reference to find out how you can add such things as `<b>boldface</b>`, `<i>italics</i>`, and `<tt>typewriter style fonts</tt>` to your pages
```
</body>
</html>
```

CompuServe and Delphi. People who use such browsers may not be able to access your pages if you don't allow for them at the design stage. This doesn't mean that you can't have graphics – you can, but there are simple rules which need to be followed in order to allow character-based browsers to access them. Of course, if you want to prevent tens of thousands of potential customers from seeing your Web pages, that's up to you.

HOW DO I HTML?

There are three options open to you when it comes to the writing of your Web pages. You can use the Save function of your browser to save a Web page that looks like how you would want your page to end up. Once it is saved to hard disk you can load the file into a text editor and you will be able to see how all the links and tags have been made.

Many people use the 'steal and copy' technique to produce their own pages, when all they do is replace the saved text, graphics and hyperlinks with their own information. This is not recommended because the best pages are usually the ones that have been designed to accommodate your own needs and not someone else's. However, studying other Web weavers' efforts is a good way of learning the basics of page make-up and provides a useful training technique.

Next, you should take the time to learn how to program in HTML, which isn't as daunting as it might sound and shouldn't take up more than a couple of hours of your time. We would recommend that you consider learning the language anyway, even if you use one of the automatic page creators. A good understanding of how HTML works results in the production of much better Web pages.

The final option is to let a sophisticated software package convert your plain Ascii text documents into a Web page for you. HTML documents are written in plain Ascii text, so you can create them using your
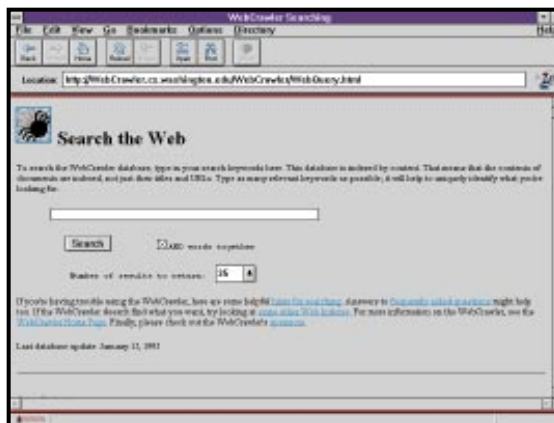
Use a Web searching engine, such as this example, based at:
http://WebCrawler.cs.washington.edu/WebCrawler, in order to search for some interesting WWW sites

favourite word processor. Save the document with an HTM file extension, and then to view it you will need a Web browser, such as Netscape, into which you can load it as a local file.

HTML BASICS

This is only a simple guide to writing in HTML, and it will use a very simple Web page as an example. Writing a more complex page is obviously more tricky, but not greatly so. Follow the basic rules set out here and you should find that everything is as easy as one, two, web.

HTML uses a system of tags to tell the browser how to display text and graphics. Tags are also used to inform the browser of the presence of 'hrefs', the hyperlink references that point to linked information stored elsewhere. HTML isn't case-sensitive, and if a particular browser doesn't support a particular tag (they don't all support all tags), it will simply ignore it.

When writing an HTML document, the whole of the document should be enclosed between the tags: <html> and </html>

This tells any software reading the text that it is dealing with an HTML document. Now you need to give your document a header, not only to tell both the user and the browser what document they are reading, but also as a bookmark entry, should a user choose to save the location for easier future access. Make the header brief and informative, then enclose it between the following tags: <title> and </title>

This should then be enclosed between the header tags, like so:
<head>
<title> Insert header </title>
</head>

The main body of a Web page is enclosed within the tags <body> and </body>, and this includes all the information that you want to display on your page, including text, graphics, audio and links to other connected information. So assuming that we are calling our test-run Web page 'PC DIRECT WEB EXAMPLE', we simply insert this heading between the two title tags.

Next you have to add your headings, and these can be chosen from six types. <h1> is the most prominent heading, used

for main page titles, through to <h6> which is the least prominent and should be kept for the smallest subheadings that you wish to include.

Try to keep the use of headings consistent throughout your document; it makes it much easier to read if you do. A main heading can now be inserted, still within the confines of the <body> and </body> tags, which you enclose between tags of <h1> and </h1>.

Now you need to add some text. To do this, you use the paragraph tags to prevent all your text being in one large paragraph, which would be difficult to read. Simply insert a <p> paragraph tag anywhere in your text where you want a new paragraph to start.

If you want to display a picture, you need to store this in the same directory as your HTML document, or ensure that you point to the right storage place, and use the following tag (remember to enclose the filename of the graphic between the quotemarks, or it won't be displayed):
<img src= "./filename.pcx">

You can specify the position of any accompanying text by using an argument of 'align=' followed by top, middle or bottom depending on where you want the text positioned. If you wanted text placed by the top of the graphic you would use:
<img align="top" src= "./filename.pcx">

To help ensure that people using character-based browsers can still access your pages, you can use the 'alt' argument to ask it to display something else. Using alt="x"

would result in an X being displayed instead of the graphic. Adding this to what we already have would give you:
<img alt="x" align="top" src= "./filename.pcx">

Finally, you may want to add a hyperlink to another piece of information or resource. This is accomplished by using an href hypertext reference to specify exactly where the information is held, which could

be anywhere on the Internet. A typical command would look like this:
TEXT

The "url" part refers to the Uniform Resource Locator, that is the exact location of the link, while the >TEXT< part defines what you wish to become the hotlink that will activate the hyperlink. This will be the part of the text that is highlighted in the browser. In our example we're using the example of The Direct Connection, WWW pages, and will make the words 'Direct Connection' the hyperlink.

We've also added a bit more text, and included tags to allow some words to be displayed as bold or italics, or a typewriter-style font.

Check the HTML command reference for details of these and other tags that you can use to make your Web pages exciting. If you enter the text and HTML commands for our example page (see *The PC Direct WWW example page*, page 441), the simple, but attractive result will resemble the screenshot on page 441.

ONLINE FORMS

Finally, you can include forms within your document that can be completed and submitted online. These have obvious potential with regard to customer enquiries, sales and market research purposes. It would, however, take another sizeable article to give an sufficient explanation of how to manufacture one of these. All is not lost, however, because you can obtain an excellent online guide to online forms by pointing your Web browser towards http://kuhttp.cc.ukans.edu/lynx_help/HTML_quick.html. Another good site for Web weavers can be found at http://www.charm.net/~web/Vlib.html.

HTML command reference guide

You can create many display effects that will make your WWW pages look that little bit better than those of your competitors. All you need to do is add the correct HTML tags into your document.

Here is a quick look at some of the commands you can use:

Address   **<address> </address>**
This specifies the author of a document, and usually includes an E-mail contact address.

Bold   ** **
Definition lists   **<dl> </dl>**
The text of the list itself is composed of an item which is tagged with <dt> and a definition which is tagged by <dd>.

Document title   **<title></title>**
Escape sequences   **<   (for <)**
                          >   (for >)
                          &   (for &)
                          "   (for ")
Certain characters can't be displayed in an HTML document without the use of escape sequences because they are used by the language itself. The characters concerned are shown to the right of the above escape codes.

Forced line break   **
**
Most browsers insert a blank line between paragraphs, but use the <br> tag when you don't

want this, such as in an address.

Headings   **<h1> </h1>**
Headings can be anything from <h1> through to <h6> in order of prominence.

Horizontal separator   **<hr> </hr>**
This will display a horizontal line, the width of the browser, between sections of text.

Italic   **<i> </i>**
Links   **<a href"filename or URL">TEXT**
Lists   **(not numbered)   **
            **(ordered)   **
The text for the actual lists should be enclosed by tags of <li> and which are placed within either the unnumbered or numbered tags.

Menu   **<menu> </menu>**
An interactive menu requires that the items to be included are enclosed by tags of <li> and which are placed within the <menu> </menu> tags themselves.

Paragraphs   **<p>**
Paragraphs don't need an ending tag.

Preformatted text   **<pre> </pre>**
Preformatted text is of specific use when displaying such things as binary code, programme listings and so on. It allows for multiple spaces to be displayed as such, for example.

Typewriter style text   **<tt> </tt>**