Designing Web Pages

I t's been about a year since I wrote anything about HTML and the Web. I've been engaged in a complete redesign of my Web site and because the thoughts on that work are uppermost in my brain, I've decided that I should dump some of these ideas into the editor and send them to you via the medium of print.

In the Web world, many things have altered in the last year. Perhaps the two most significant events were the emergence of the HTML 3.2 standard and the free availability of Microsoft's Internet Explorer. The HTML 3.2 standard gave a resounding "Yes" to much of the new HTML markup that had been described as "Netscape Extensions." This has made me much happier about using many of these features.

Explorer, finally, was brought up to a level where it could use all these later extensions sensibly. If you like, it arrived at a point where it could compete with Netscape.

These days, my Web browsing machine of choice is a Windows NT system. It has a screen and graphics card capable of displaying 24-bit color images. I am not limited to Sun's 8-bit color-mapped display, on which embedded Web images can become grainy with all those little dots of incorrect colors. The computer supporting NT is more than an entry-level low-cost machine, yet cost me a little less than buying a 24-bit color framebuffer and monitor for my Sun. In addition, the new machine includes a 1-GB disk and a 200-MHz Pentium processor. I wonder how many other people have made this comparison and have chosen this route? I hope Sun is listening.

Using the NT system means that I can use several of those nasty plug-ins that Netscape thinks are such a good idea. Many of the nicer ones don't run on UNIX. Actually, I am deeply suspicious of the whole notion of plug-ins. I don't want to spend my life loading odd bits of code to visit your Web pages. Usually, I get to a page of interest and then find it uses that really extra special all-singing all-dancing Hippopotamus plug-in, which I haven't got. At this point, I usually decide to go away.

This is not to say that I have abandoned the Sun. No way. The UNIX system provides me with a layer of sanity on my network. It hugs the NT system, giving it mail access, name and time service. Also, Microsoft doesn't seem to understand the need to access tape drives on the network, so the NT system has no adequate backup. As a consequence, I run Samba on my Sun, allowing NT to use the Sun's filestore for any important files. If someone out there has a freely available NT 4.0 tape driver that will talk the Berkeley tape protocol into my UNIX machines, then please mail me, I will be grateful.

Of course, I could protect the NT disks by extending the Solstice backup system that already happily supports Solaris. But I am too stingy (or too poor, believe what you wish) to pay the money for the licenses that I need. The step from single-server to network licenses is a costly one.

So, I have a bunch of different machines running different operating systems with different browsers. If you are in the Web site building business, then this is a must. I could do with a Mac, but again I am too stingy, or too poor.

I started hacking on the server back in the summer, when

MARGERY MINTZ

the sun was shining and I could take new photos for my "Tour of Canterbury" Web site. The tour grew from around 100 to 300 photos. I was very pleased when, out of the blue, the tour was given a Three Star Award by the *Magellan Internet Guide*. As the summer faded, I realized that I was generally dissatisfied with the look of the pages on the server. The tour pages were (and are) fine, but the other pages left a lot to be desired.

What was actually wrong did not really crystallize until I read David Siegel's Web pages (see below for the URL). I don't agree with everything he says, but he makes many good points that spring from graphic design rather than computing.

I must emphasize at this point that I have no training in graphic design, and I have acquired the knowledge of all that follows recently. Actually, when you start thinking about these issues, the simple layout rules are plain for you to see when you examine magazines like this one. However, I cannot pretend to be an artist because I am not.

Horizontal Space

First, consider horizontal space. Most of this magazine is printed in three columns because the designer knows that humans read short lines more easily. Short lines simplify the process of line traversal, hitting the end of one line and finding the start of the next. My articles usually appear in two columns so that I can insert fixed-width text code sections in line without you having to break off from reading

and find that annoying box with the example placed in it. However, I suspect that the designer for this magazine finds my two-column layout is a little wide for his taste.

Now, what happens with page display on the Web? You take some text and put it into a Web page and it's usually displayed as a bunch of long lines stretching right across the screen. The line width is under the control of the reader, who will probably set their window size to accommodate images rather than for ease of reading text. The result is that your text will appear unappetizing, and the reader will surf off without looking at it.

Ideally, the page author needs some control over the width of lines of text. In fact, this is not difficult: You place all the text inside an HTML table, and the `WIDTH` argument can be used to specify the number of pixels that you wish a line to occupy. While you are doing that, it's a good idea to supply a margin down the left-hand side of the page. Again, a margin helps the reader identify the start of the next line.

Here's a template for creating documents:

```
<!-- Header -->
<HTML><HEAD>
<TITLE>Title</TITLE>
</HEAD>
<!-- Body -->
<BODY>
<!-- Page start -->
<TABLE WIDTH=450
```

```
        BORDER=0
        CELLSPACING=0
        CELLPADDING=0>
<!-- Row start -->
<TR>
<!-- Column 1 -->
<!-- Margin -->
<TD WIDTH=50><BR></TD>
<!-- Column 2 -->
<!-- Information -->
<TD WIDTH=400>
content
</TD>
</TR>
</TABLE>
</BODY></HTML>
```

Now I am hoping that you are familiar with HTML. HTML markup consists of words enclosed in `<...>` called *tags*. Most tags are actually pairs, with the opening tag being the word, and the closing tag being the same word preceded by a slash. For example, the whole body part of the document is enclosed between `<BODY>` and `</BODY>`. Some tags stand on their own, marking some point in the text. An example is the break tag, `<BR>`, which forces the current line to be terminated and a new line to be started.

A table is introduced by the `TABLE` keyword, and in the example, this is followed by some attributes. We first set the width of the whole table to 450 pixels. Then we ensure that there is no border on the table (`BORDER`); there will be no spacing between the cells of the table (`CELLSPACING`); and finally, that the cells inside the table will have no internal margin around them (`CELLPADDING`).

We now move onto specifying the table. We first define the one and only row using the `<TR>` tag. There's a matching closing `</TR>` tag at the bottom of the template. The row consists of two columns, each sitting inside the `<TD>...</TD>` tags. The start of each column defines its width in pixels using the `WIDTH` attribute. The first column, the margin, is 50 pixels wide, and the second is 400 pixels wide. Notice how this sums to the 450 that was used to size the whole table.

Incidentally, you might think that the `WIDTH=400` on the right-hand column is not needed because the browser should be able to deduce that. The overall table size is 450, and the left margin is 50, so the remaining space *must* be 400. I feel that this is a reasonable expectation, and so does Microsoft. Its Explorer doesn't need this explicit specification. However, all the Netscape Navigator versions that I have on site will simply lose the left margin column if the explicit width for the body of the text is not supplied.

If there is any text or images in the margin, then I will add `VALIGN=top` to both `<TD>` specifications. This ensures that the contents are nailed to the top of the table and don't get vertically centered, which is the default.

However, in the template, the content of the margin is empty and uses the line break tag to say this. The content

of the main part of the page is what I wish the reader to see. The template ends with a bunch of closing tags to balance what has gone before.

I've used the template fairly extensively on my server. I have sometimes widened the page to 500 pixels, with between 400 and 450 pixels for the page content. I have even sometimes widened the margin to 75 pixels with a body width of 375.

There is, of course, a snag with the whole idea. It imposes a fixed width in the browser window but doesn't control the text that is placed there because the user can change the size of the characters displayed. Ideally, if the user wishes to increase the size of the fonts to make things more visible to them, then we would like to be able to present the same text, but bigger. However, we have explicitly set the number of pixels, so an increase in character point size will mean that less text is being displayed. What is really needed is a way to set the width depending on the size of the font that is being used.

Explorer only allows you a fixed number of settings to specify the font size, and the 400-pixel width seems to work fine with the largest setting that is available. Navigator allows you to choose an explicit point size for the displayed characters, and you can choose values that are so large that meaningful text is not displayed. However, at these point sizes, most Web pages would be displayed incorrectly, so this may not be a real problem.

Color Use

Having sorted out the general page layout, we come to the vexing question of what colors to use. There is now complete choice for the page designer to specify the colors of the background; the text; the links, both visited and unvisited; and the color to which a link changes when it is selected.

It's the colors of the links that dictate what colors you are able to use on your pages. Both Netscape and Microsoft use blue to mean that the text or image is a link that has not been visited and a dark purple to indicate a visited link.

The other day, I found myself clicking fruitlessly with the mouse button at the Web page of an organization that had seen fit to place some blue underlined text in the middle of the screen. It looked like a link. Eventually, I realized nothing was happening and understood that it was not actually a link. However, it was interesting that my brain had decided that this visual clue was enough to indicate the presence of a visitable link.

It may be that the colors of links are part of your reader's implicit knowledge of how to use your Web pages. I have become very queasy about making gratuitous changes to these colors. I'm beginning to think that it's really a bad idea to change link colors on Web pages.

Given that I want to retain the standard default colors, then what about background colors? Well, people go mad with backgrounds, and there are many unreadable pages on the Net. I sometimes wonder if these people are making some statement about content: "I have nothing of interest on my pages. Look at the pretty background and leave. Now."

I am trying to make my pages accessible, and this means

that they need to be readable. I've opted for a white background with black text for most pages because the combination works well with the link colors. I've used light yellow or brown-yellow (bisque) backgrounds for some pages. These colors contrast well with blue and purple. Using a black background with white or yellow text works too, although the blue and purple links can be faint.

The biggest problem with colors is that color is platform-dependent. Most of the light backgrounds that look reasonable on a screen capable of displaying 24-bit color will appear white on a typical 256-color screen. If you are using color as part of the design to impart some flavor to the page, then you want that color to be seen by all your readers.

The other alternative is to place a stripe down the left-hand side of the page in the margin. The trick here is to create a GIF image that is one pixel high and very wide. The browser will then tile the screen with the image, creating a colored stripe on the left with the body of the page being set to white, for example. Incidentally, some people who do this don't use images that are wide enough. My backgrounds are 1,152 bits wide, which will fill the width of my Sun screen should they need to. The GIFs that implement this tiling are about 70 bytes. I've used a stripe in the margin extensively in my server. I am happy with some of the colors that I am using and less happy with others. No doubt this will evolve.

If you create a wide margin, then it's reasonable to use the space to place icons and buttons to allow readers to select different places to go on your server. If you look at the *SunExpert* pages on my server, you'll see that I've added a year selection box in the margin. You can get the same effect with frames, but then you get into the bind of viewers with no frames.

The main problem with placing images or text into the margin is horizontal spacing. Browsers automatically add internal borders to the windows, so that the text displayed by default starts a little way into the window area. This is fine with me because anything in my left-hand margin is not crammed against the edge of the screen window.

Sadly, each browser adds a different size border. Navigator on my Sun uses a 20-pixel left margin, and Navigator Gold and Explorer on my NT machine both add 10 pixels. This is a disaster when you wish to accurately place text or images over some preset background.

Explorer offers some syntax that allows you to set the border size. To set the offsets to zero, add `LEFTMARGIN=0` and `TOPMARGIN=0` to the `BODY` tag on the page. However, this is not in the HTML standard and is not supported by Navigator, and as such, is not all that useful at the moment.

Vertical Space

Siegel points out that the default settings for the use of vertical space in the output from browsers are also a disaster. Again, look at a subhead on this page. Notice that the space above the subhead is greater than the space below it. This spacing is important to make the subhead look correct.

On a Web page, the heading directives insert equal space above and below the text line that they enclose. In general, all the directives that insert vertical space on the Web page

add a fixed amount of white space. Siegel contends that this is why we see so much use of horizontal rules and separators on Web pages. The separators are needed to provide the reader with an indication of changes of content. What is really needed is proper control of spacing, allowing vertical white space to be added for emphasis where it is needed.

To get adequate control of white space, Siegel throws away all the markup directives on his pages and only uses `BR` to ensure that he gets a line break inserted with no additional vertical space. He then controls spacing in both directions by using a single pixel transparent image and tweaking the spacing control supported by the `IMG` command.

For example, a typical Siegel subheading looks like:

```
<!-- start of table with two columns -->
<!-- see above -->
<!-- this is the 2nd column -->

<!-- text of heading in bold --
<B>Heading</B><BR>
<IMG SRC="dot_clear.gif" VSPACE=1><BR>
Text.<BR>
```

He uses the white space inserted by table start directives to provide a wide gap above the subheading. His subheading is then in bold and is followed by a break. He adds three pixels under the heading by specifying the image (one pixel) and two pixels of additional vertical space (`VSPACE=1`), one pixel above and one below the bitmap. This trick seems to work on all the browsers that I use. I am slowly using this effect on my servers to make headings look better.

Siegel also goes on to use the `HSPACE` directive in an image to insert horizontal white space to indent the start of paragraphs. This only seems to work correctly in Navigator on the Windows platforms. Both Navigator on the Sun and Explorer on Windows add extra horizontal pixels around graphic images, and there seems to be no way of eliminating them.

At best, the technique of using in-line images to provide spacing is a hack. In the longer term, I guess we are looking toward the emerging style sheets to provide proper control of the character spacing from a typographical viewpoint. However, until every browser supports style sheets, they are not really usable.

In the interim, I am moving toward the idea of writing all HTML pages in a markup language and processing that language with the `m4` macro-processor to generate optimized HTML. "Tour of Canterbury" has used this technique from the start, and it allows me to make dramatic style changes to the 300 pages with no grief. I've also used `m4` markup extensively for a new Canterbury server that I have been creating.

Finally, we all need to plead with the HTML standardizers to closely specify how both horizontal and vertical space is automatically inserted by browsers. The current situation where each browser inserts space at different points around graphic images and HTML tags is a mess. I want my pages to be browser independent within the limits of the platform on which that browser is running.

Where to Find Things

David Siegel's Web pages are found at `http://www.dsiegel.com/tips`. He has also written a book on Web page design (I have not yet seen it), refer to `http://www.killersites.com` for further details. You can also find his `dot_clear.gif` at these sites. In addition, there is a link to the FTP directory on `ftp://ftp.verso.com/pub/dsiegel/` from `http://www.dsiegel.com/tips/ wonk5/single.html`.

If you are thinking about what colors to use on your pages, then a dose of color theory is no bad thing. See `http://www.contrib.andrew.cmu.edu:8001/usr/dw4e/color`.

My site is `http://www.hillside.co.uk`. Incidentally, you'll find the links in this section attached to the article listing on my *SunExpert* page. So don't type–click and ye shall find.

And, oh yes. To my knowledge, there is no Hippopotamus plug-in for Netscape. If one pops up before this goes to print, then apologies. You should have chosen another pachyderm. ✍

Peter Collinson runs his own UNIX consultancy, dedicated to earning enough money to allow him to pursue his own interests: doing whatever, whenever, wherever… He writes, teaches, consults and programs using Solaris running on a SPARCstation 2. Email: `pc@cpg.com`.