

Chapter 4: HTML and HTML Authoring Tools

To many Web authors, HTML, the language of the Web, seems to be a relatively new way of presenting content. Actually, HTML derives from the heritage of SGML, the Standard Generalized Markup Language, which has been around for two decades or more. HTML dates from the late 1980s, but practically speaking was known by only a handful of people before the appearance of NCSA Mosaic in 1993. Now the whole world uses HTML.

This chapter provides an overview of HTML, and an overview of tools – so-called “authoring tools” – that make it easy for you to create HTML documents. Most Web content providers find that it’s still necessary to learn HTML “tags” in order to tailor pages for the best possible presentation. This is especially true for the more visible pages on your site, such as “splash screens” and other high-level pages. However, in many cases, your content providers may use an HTML editor (or word processor that can translate to HTML) without learning a single HTML tag. Nonetheless, we begin with a brief overview of HTML.

HTML: the Language of the Web

HTML is an example of a “markup” language. Simply put, this means there are special markers, known as “tags,” that denote special meaning in an HTML document. These markers are visible and appear interspersed among the text that makes up your actual content. For example, in HTML if you want to identify the title of your document, you’d do so in this way:

```
<title>Smallville Community Information</title>
```

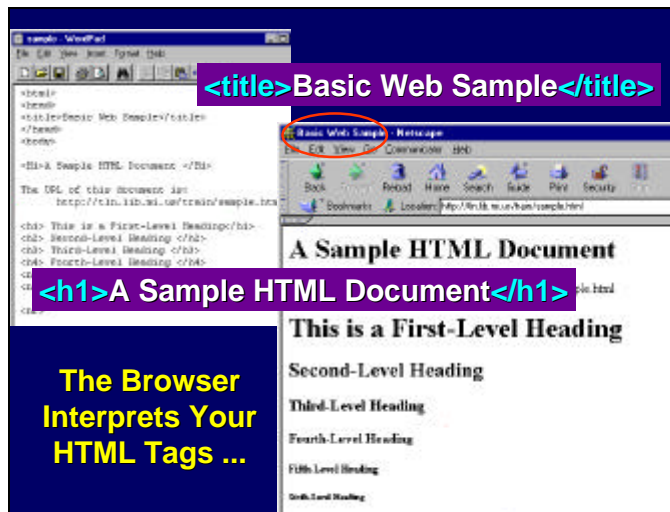
It’s really quite straightforward: the `<title>` tag identifies that the following text is the title of the document; the `</title>` closes the title. Everything in between is the actual title of the document.

Tags in HTML theoretically denote *logical* concepts – the `<title>` is an abstraction; there is no particular action or display format implied for a title. Practically speaking, a title in an HTML document is a signal to the browser as to what to put in the little blue bar atop the document window on screen. But the title might be used in different ways by different browsers. Furthermore, other tools may make other uses of something as special as a title – for instance, a search engine might treat titles specially in its index.

Building a Community Information Network: A Guidebook

```
HTML: Text and "Tags" Reside  
in the Same File  
  
<title>Community Information</title>  
...  
<h1>Welcome to Smallville</h1>  
<p>Smallville is a picturesque town on  
the banks of the Au Sable River  
</p>
```

An individual "page" consists of HTML markup and your textual content. Web browsers such as Netscape Navigator and Microsoft Internet Explorer know how to interpret HTML tags in order to render the content of each page coherently on screen..



The Web consists of a global network of servers, each of which holds a number of documents in the form of HTML files, ready to be served on demand to a user. The situation is a little more complicated when we consider so-called dynamic pages, produced for instance by a database server, but for now we're considering hand-crafted, "static" HTML pages.

Let's put all these terms into context:

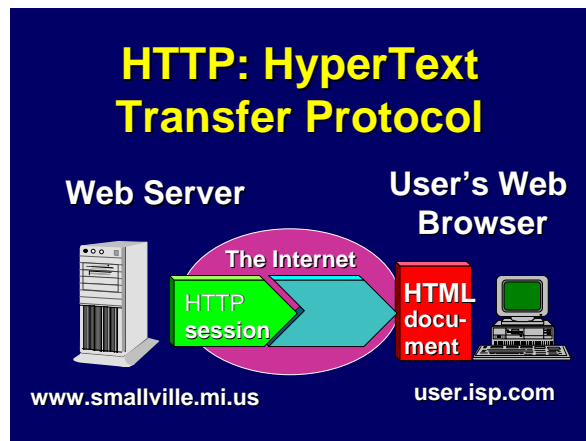
- A Web browser is software used to "surf" the Web – to pull down desired Web pages and to display the contents of each page on screen. Netscape Navigator, Microsoft Internet Explorer, the Web TV browser, and Opera are examples of Web browsers.
- A Web server is the hardware and software that deliver Web pages on demand.
- A Web "page" is a single HTML document as displayed by a Web browser. A Web page may consist of multiple files on the server; in fact, this

Building a Community Information Network: A Guidebook

is typically the case, as graphics and photographs within a single page are typically stored as separate files, one per image.

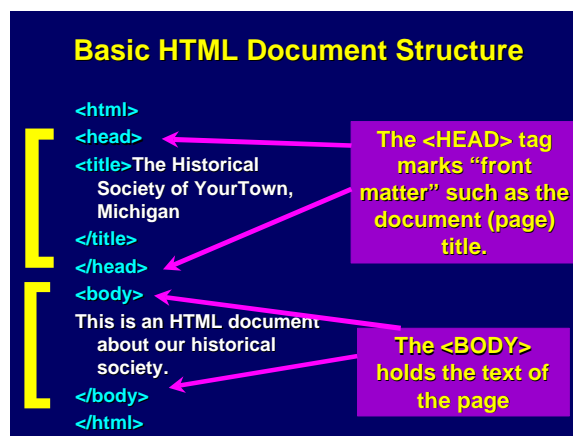
- A URL or "Uniform Resource Locator" is the address of a Web server or a Web page.

The following diagram shows the relationship among these elements:



As you compose HTML files, you place them on the server, which waits until a user requests a file by its URL. The server delivers the file to the user's browser, which displays the file on screen after downloading it.

Within each HTML document there is a basic structural layout:



This structure will be consistent across all HTML documents you write.

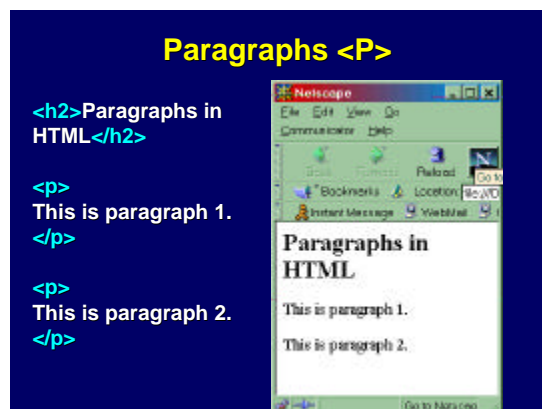
When composing HTML, keep some basic points in mind: the "case" of the tags doesn't matter; `<title>` and `<Title>` and `<TITLE>` and `<title>` are all the same tags. However, many HTML authors use a convention of typing their tags all in lower case. For readability, some HTML authoring tools can be configured to convert whatever you type into all lower case or all upper case.

Building a Community Information Network: A Guidebook

Spacing in your source document doesn't matter; include as many carriage returns and extra spaces in your source document as you wish in order to make your document readable and easily edited. Your Web browser looks only at tags to determine layout on screen; it does not notice extra "white space."

Now let's consider some basic HTML tags. The `<p>` tag separates paragraphs in HTML. Originally, the `<p>` tag was truly a separator; it appeared *between* each paragraph. More recent HTML standards treat the `<p>` as a "wrapper" analogous to the `<title>` tag:

```
<p>
This is paragraph 1.
</p>
<p>
This is paragraph 2.
</p>
```

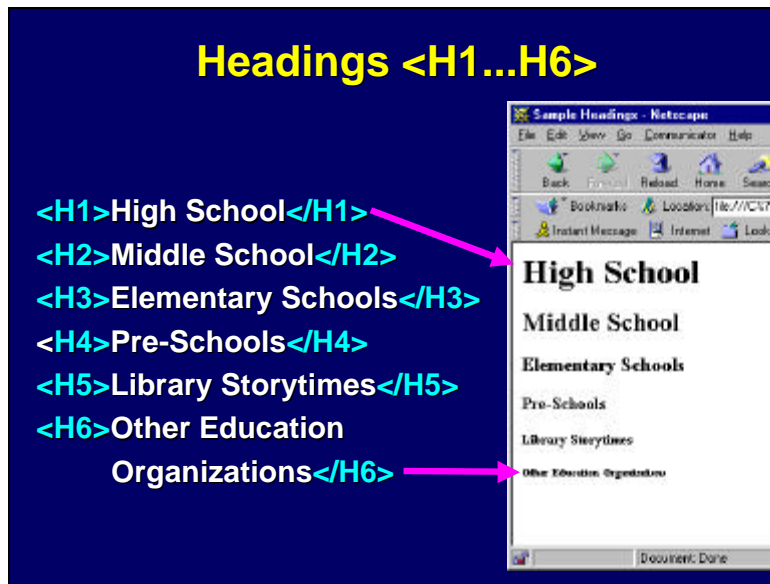


Typically, browser show paragraph boundaries on screen by separating them with a blank line.

HTML defines up to six levels of section headings. You designate these headings with tags of this form:

```
<h1>This is a Level One Heading</h1>
```

Each heading is shown on a line (or more than one line) by itself, in a bold and prominent font.



The `<center>` tag informs the browser that the text that follows is to be centered within the browser window. You can center an entire paragraph, or a section heading, or other elements of text or graphics. Whatever text you include between `<center>` and `</center>` will be centered within the browser window.



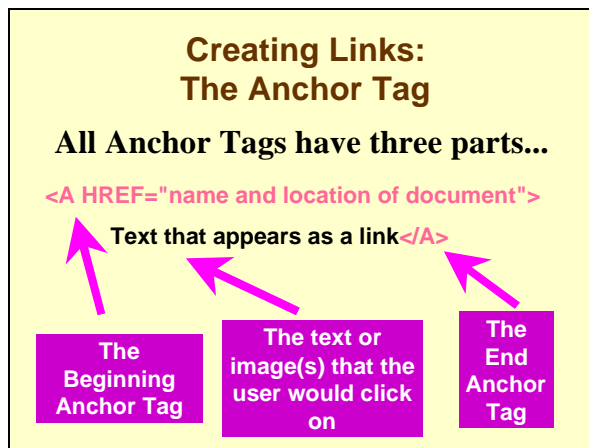
The Web wouldn't be the Web without links. Pages are linked together to make up an interwoven Web site; sites are linked together to make up the global Web. In order to link from one Web page to another, you use the `<a>` tag, called an *anchor*. This tag takes the following form:

```
<a href=http://www.smallville.gov>Smallville  
Government Web Site</a>
```

Building a Community Information Network: A Guidebook

In this example, we'll see on the screen a hyperlink for Smallville Government Web Site – it will appear in blue and underscored. If we click on that link, our browser will connect to the server named `www.smallville.gov`, which will return its default, or “home” page to our browser. The text string `www.smallville.gov`, is a Uniform Resource Locator, or URL.

An anchor can link to another page within your Web site – or to any other page on any other server on the global Web. The URL within the anchor tag determines what you're referring to:



The URL of the page to be fetched is *not* shown in the user's Web browser – only the text outside the `>` and before the `` that closes the anchor tag. For instance:

Creating Links to Other Web Sites

```
<H3>Contact Information</H3>  
<B>The Historical Society</B><BR>  
1212 Main Street<BR>  
YourTown, MI 48195<BR>  
<HR>  
<A HREF="http://milf.tln.lib.mi.us/mcin/comm.htm">Go to  
the Milford Community Network</a>
```

Contact Information

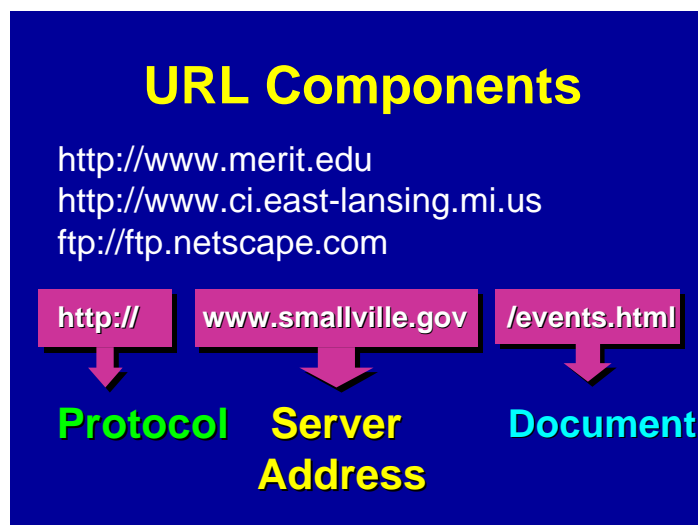
The Historical Society
1212 Main Street
YourTown, MI 48195

[Go to the Milford Community Network](http://milf.tln.lib.mi.us/mcin/comm.htm)

Since we're using URLs in our anchor tags, let's explore how URLs work in a little more detail.

Building a Community Information Network: A Guidebook

In this example, `www.smallville.gov` is once again the address of our server. (In this chapter, we use varying possibilities for our server's address, from `www.smallville.gov` to `www.smallville.mi.us`. See Chapter 10 for an explanation of the Domain Name System, and what “`www.smallville.gov`” really means.) Optionally we have a part of the URL that specifies a particular file within the server – in this case, `events.html`. Thus we're asking the server at `www.smallville.gov` to deliver a file named `events.html`. In the next section we'll explore the layout of files on our server. The “`http://`” part of the URL simply indicates that we expect to use the Web's normal transfer protocol, HTTP, to fetch the file from the server. Before we explore server file layout further, we have a little more HTML to explore.



Just as it's important to be able to link to other documents, you also need to be able to incorporate images into the current document. The `` tag lets us do this, for instance:

```

```

Here, we're referring to a photograph that's stored in GIF (Graphics Interchange Format; see Chapter 6). The image tag is actually a way to request that an image be displayed “inline” – that is, on screen adjacent to other text and graphics.

In an HTML document, you would include a separate image tag for every inline image you want displayed. It's quite common to have five or ten images on a page – an image might not be a photograph, but instead it might be a graphical element such as a logo or an icon you want displayed.

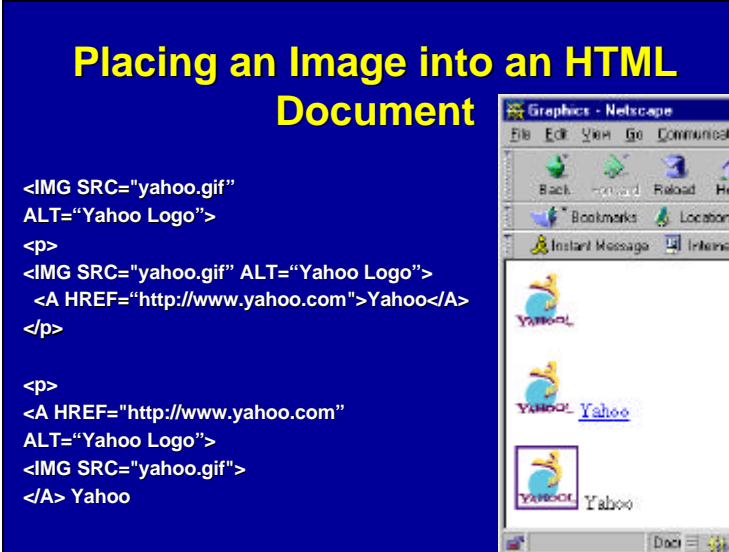
Building a Community Information Network: A Guidebook

A good option to include with your image references is the `alt` parameter. For example:

```

```

The `alt` tag is used by some browsers to give a label for an image during download. Recent browsers display the `alt` tag for an image when you put the cursor atop it. Also, “talker” browsers used by the sight impaired can read the content of `alt` tags to the user, giving an idea of what in image on screen represents.



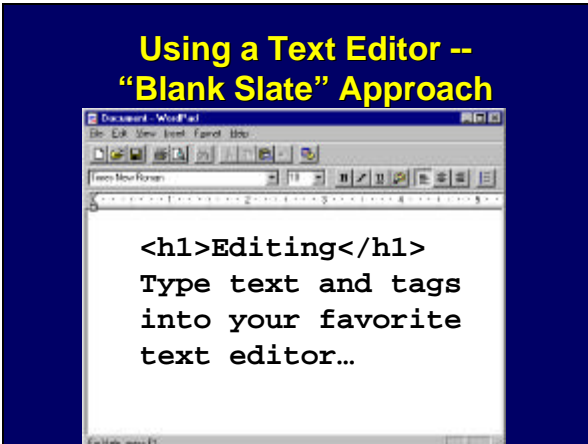
Placing an Image into an HTML Document

```
<IMG SRC="yahoo.gif"
ALT="Yahoo Logo">
<p>
<IMG SRC="yahoo.gif" ALT="Yahoo Logo">
<A HREF="http://www.yahoo.com">Yahoo</A>
</p>
<p>
<A HREF="http://www.yahoo.com"
ALT="Yahoo Logo">
<IMG SRC="yahoo.gif">
</A> Yahoo
```

The slide also shows a screenshot of a Netscape browser window displaying the rendered HTML code, which includes the Yahoo logo and the text "Yahoo".

Other HTML Tags

This brief introduction should give you a flavor for the basics of how HTML works. You will probably want to learn more about HTML before you begin building Web pages. You'll want to learn about lists, and tables, and forms, frames, and background colors, and many other options.



Using a Text Editor -- "Blank Slate" Approach

```
<h1>Editing</h1>
Type text and tags
into your favorite
text editor...
```

The slide shows a screenshot of a text editor window with the above HTML code and text.

Building a Community Information Network: A Guidebook

It is possible to write Web pages without learning HTML at all. Later in this chapter we will discuss authoring tools that allow you to do exactly that. Many HTML purists will argue that you need to learn to edit HTML “by hand” and learn most of the basic tags before moving on to an authoring tool; others see no reason why anyone needs to learn any HTML at all.

A common approach to editing HTML “by hand” is to use a simple text editor such as Windows Notepad or Wordpad. You edit your HTML document, typing in all tags manually, and you then Save your work in progress and inspect it in your browser periodically. (Simply tell the browser to open a file, and click Browse to find the file on your local hard disk.)

If you use this trick, you can even keep your text editor open in one window, and your browser open in another, and hop back in forth, hitting Save in the editor, and Reload in the browser.

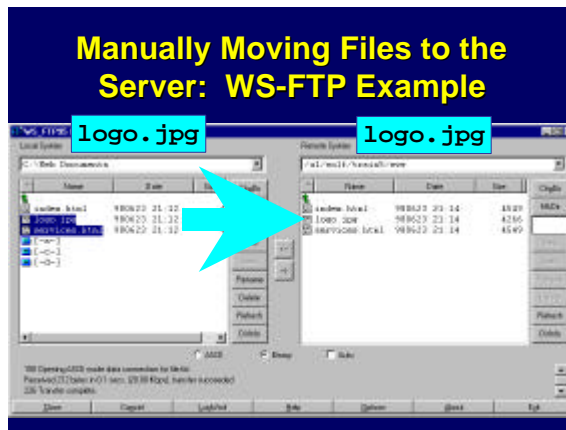


One of the best ways of learning HTML is to inspect others’ pages. If you wonder how a page accomplishes a particular trick, use the View Source feature in your browser. Warning: some tricks are pretty fancy.

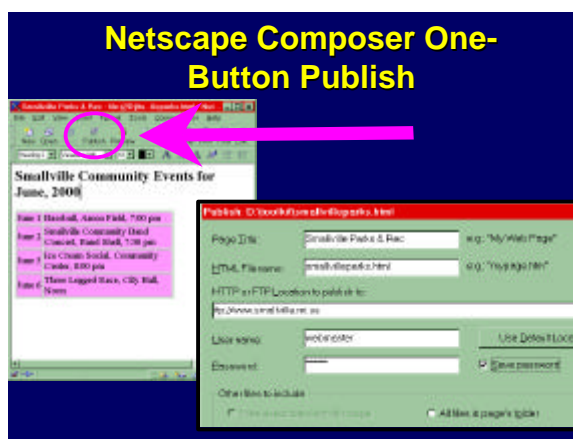
File Organization on the Server

Typically, you edit your HTML documents on a local PC. After you have created a basic set of HTML documents, you need to move them to your server, where they will await delivery as demanded by your users. There are several ways to move files to the server:

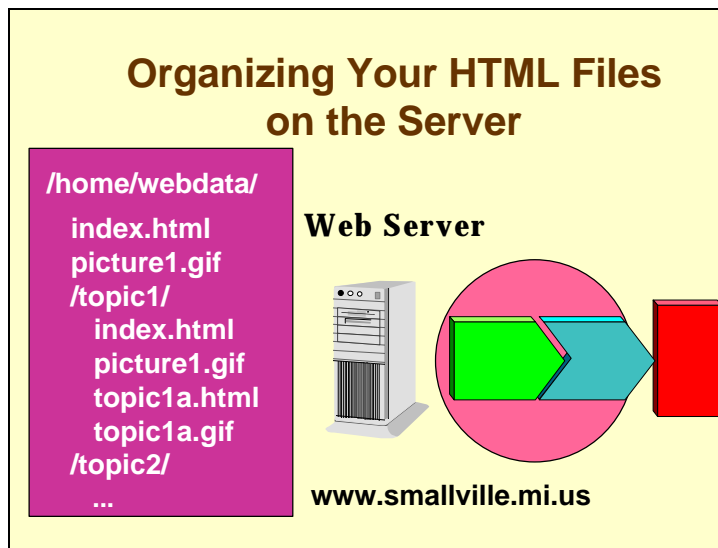
Building a Community Information Network: A Guidebook



- You may have been given a user ID and a password and permission to use the File Transfer Protocol (FTP) to move files from your hard disk to the server.
- You may be using an authoring tool such as Netscape Composer that offers one-button publishing of a single page to the server.
- You may be using an authoring tool such as FrontPage that has a special mechanism for posting an entire collection of files to a server. In such an environment, you might be able to edit a number of files, and define where the entire collection belongs on the server, and post all of the files en masse by clicking on a simple “Publish” button.
- You may be part of a network, such as Windows Network Neighborhood, and permissions may have been defined so that you can “drag and drop” files onto the server. (This typically applies only if you, the HTML author, work physically in the same building or campus as the Web server that hosts your files.)



The server, like all computers, will have a file system that is organized hierarchically. When the Web server software was installed, its administrator be placing files into a particular spot – perhaps at the start of the tree, perhaps at one of the branches.



Here we see a simplified example of how a server might have its file system laid out. Our server is named `www.smallville.mi.us`. The server administrator has designated an arbitrary starting point for our Web document tree, in this case `/home/webdata`. We are free to organize our files anywhere beneath that starting point in any way we want.

But we do have some conventions to consider. First, we want to have a file in every folder that identifies all the other files in that folder – by convention, on Unix servers that file is named `index.html`. (On Windows NT servers, that file would be named `default.htm`.) This file is itself an HTML document. It will refer to all of the other files it needs in order to tell its story – any other HTML files (which would be linked via `<a>` anchor tags) as well as inline image files (such as `picture.gif`, which would be referenced using an `` tag).

Under `/topic1`, we have a new folder with its own set of files, beginning with its own `index.html` file. Here we also have a `picture1.gif`, which might be referred to in our `index.html` file. And we also have an additional HTML file, `topic1a.html`, and a corresponding image file.

The URL for these documents is going to consist of the server name followed by the path name of the file in question, *omitting the part of the server file system that is “above” the document root*. Thus we might have a URL of the form:

```
http://www.smallville.mi.us/topic1/topic1a.html
```

In organizing files on the server, you’ll want to come up with a design that matches the topical layout of your Web site in general. For instance, if your

Building a Community Information Network: A Guidebook

site is divided into 5 main topic areas, you'll probably want to group each of

Because the file system layout you choose your users will see, you'll want to avoid the extremes:

all
directory. This would be a mistake, because you would have a very hard time

- At the other extreme, you could sub-divide excessively, which would directories. This would be unwieldy for your users.

forward slash, not the back-slash. Windows continues to use the back slash servers, but the correct separator for Web Windows-based Web server will translate forward slashes in URLs to back-embedded back-slashes is usually in error.

To continue with our example above, let's look at how our file might refer to an associated image file,

```
< src="http://www.smallville.mi.us/topic1/topic1a.g
```

This would tell the user's browser to go and fetch the file from the server and display it as a part of the

That form of URL is called "absolute" – it gives the one, only, absolute global Web.

```
<
```

This is called a "relative" URL. By relative, we mean that the address is a encountering such a URL, the browser will, in effect, say "Hmm, give me the `topic1a.gif`

Building a Community Information Network: A Guidebook

www.smallville.mi.us as the one that holds the file I'm now viewing."

The advantage of relative URLs is simple: they allow you to pick up and move *all* of the files within a folder without having to adjust the references – inline images, or hyperlinks – within the HTML documents. Because the references are relative, they adjust automatically no matter where you move the folder – into another spot in the hierarchy of your current server, or to another server altogether.

Relative and Absolute References to Same File...

http://www.smallville.mi.us/topic1a.html
...includes this *absolute* reference to topic1a.gif image:

/home/webdata/
/topic1/
...
topic1a.html
topic1a.gif

http://www.smallville.mi.us/topic1a.html
...includes this *relative* reference to topic1a.gif image:

The diagram illustrates the relationship between absolute and relative references. A central box shows the file structure: /home/webdata/, /topic1/, ..., topic1a.html, and topic1a.gif. Three pink arrows point from this box to the absolute reference code above and the relative reference code below, showing how both refer to the same file.

The advantage of absolute URLs is the flip side of the coin: if you move an HTML document but you don't move the files associated with it, then the absolute URL will always work. Absolute URLs also allow you to refer to files on many different servers – perhaps more than one server on your own site, or various servers around the planet.

You might use absolute URLs if you have a particular folder where you keep files you know you're going to refer to from many places. For instance, you might have a folder called `http://www.smallville.org/logos` to hold any image files for logos you'll use.

If all this seems confusing, the best thing to do is to create some sample files and experiment with relative versus absolute URLs. Watch how the browser handles both forms. (In fact, that's the best advice for understanding any Web-related technical question – experiment.) But take note: *the choice of when to use relative URLs versus absolute URLs is a key design decision.* If, for instance, you always use absolute URLs in all your HTML files, you will have created a site whose component folders won't be portable when you choose to reorganize your site.

HTML Standards

Building a Community Information Network: A Guidebook

HTML is supposed to be a language defined by an open standard; however, due to battles for industry dominance, in particular between Microsoft and Netscape, there have been some unfortunate detours from the open standards process.

As of this writing, the highest level HTML standard endorsed by the official standards body, the W3 Consortium, is HTML 4.0. HTML 4.0 attempts to incorporate in a single standard many of the features introduced by Microsoft and Netscape in 1997 and 1998. See www.w3.org/MarkUp for more information.

If you adhere to the specifications of HTML 4.0, any recent browser, from Microsoft, Netscape, or others should be able to display your content. Unfortunately, many users lag several months behind in installing new releases of browsers – perhaps 50% or more of your users will use a browser that is six months or more behind. Others may use browsers such as the WebTV browser that do not aspire to handle all the latest bells and whistles of newer mainstream browsers.

Web Authoring Tools

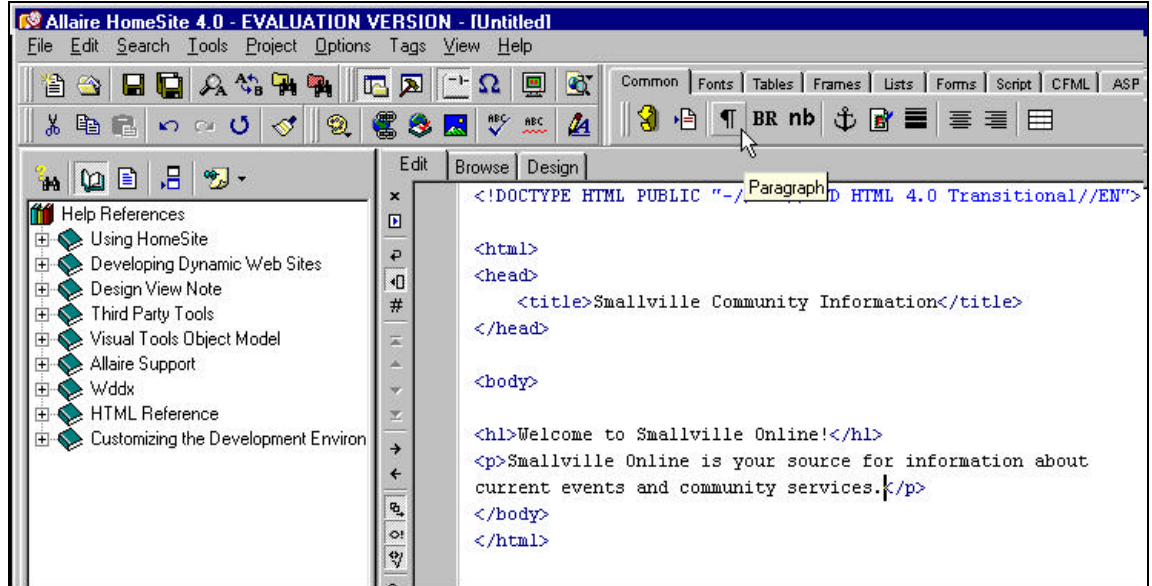
While some experienced HTML authors prefer to code their pages by hand, most admit that authoring tools can enhance productivity and accuracy. Authoring tools fall into two broad categories:

- Some tools *translate* from another format to HTML. Translators exist to move from MS Word, WordPerfect, Pagemaker, etc. to HTML. In fact, just about every common text editor or page layout program now has an option to translate from the native format to HTML. In most cases, it's as simple as going to the File menu, and selecting Save As, and selecting HTML as your desired format.
- HTML Editors are used to compose original HTML documents, or to edit existing ones. These include Adobe PageMill, Netscape Composer, HTML Assistant Pro, and MS FrontPage.

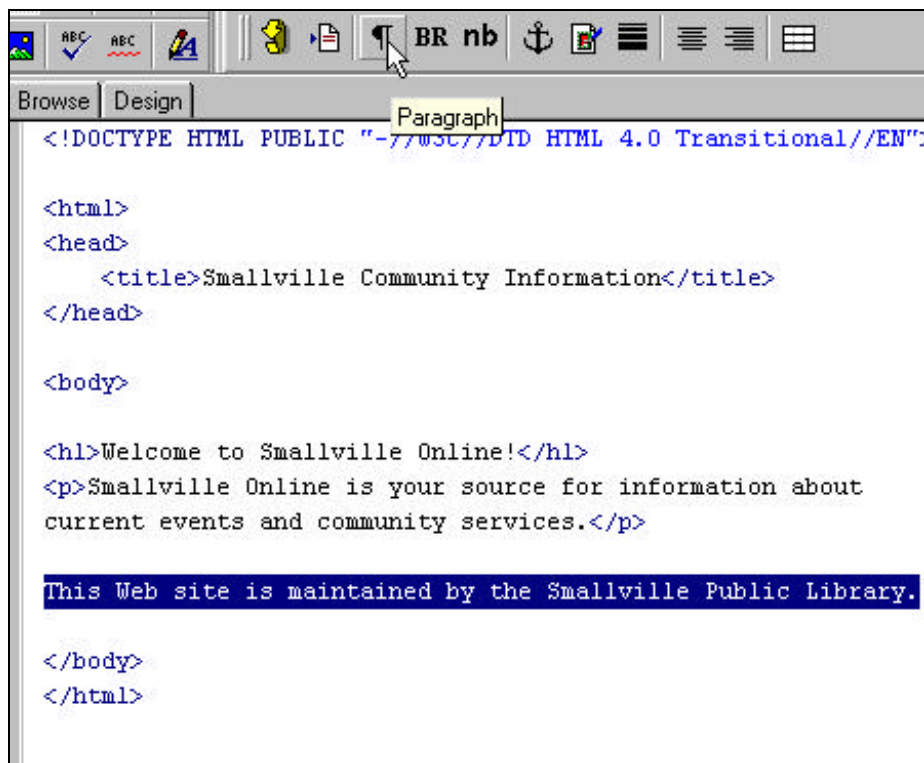
As a general rule, translators do the best job they can converting from the native format to HTML, but the resulting Web pages may not be as attractive as they would be if they were created using a native HTML editor – or carefully crafted by hand. You may want to use HTML editors or hand-crafting for your most visible pages. (With Microsoft Office 2000, the vendor claims that the Office suite has come much closer to using HTML as a native format.)

The simplest of HTML editors are in a sense “macro” tools – they save you the labor of typing HTML tags, but the tags remain visible to you. Such tools include HTML Assistant Pro and Allaire Homesite. Here is an example of a page being edited in Homesite:

Building a Community Information Network: A Guidebook



Note that the cursor is on an icon in the Toolbar – the ¶ symbol, representing a new “paragraph.” Every time you click on this icon, Homesite will insert the `<p>` and `</p>` tags to open and close the paragraph. If you highlight text you’ve typed, the `<p>` and `</p>` will wrap the text, creating a new paragraph for you.



Building a Community Information Network: A Guidebook

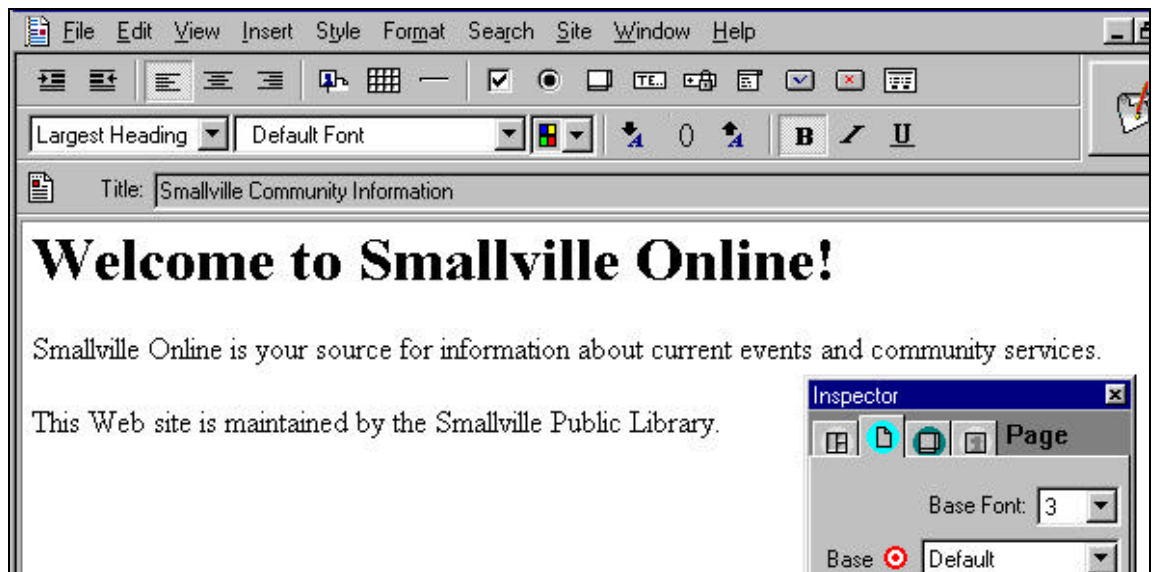
Here is the resulting HTML after we click on the ¶ icon:

```
<p>Smallville Online is your source for information about  
current events and community services.</p>  
  
<p>This Web site is maintained by the Smallville Public Library.</p>
```

Note that the `<p>` and `</p>` tags have been supplied automatically.

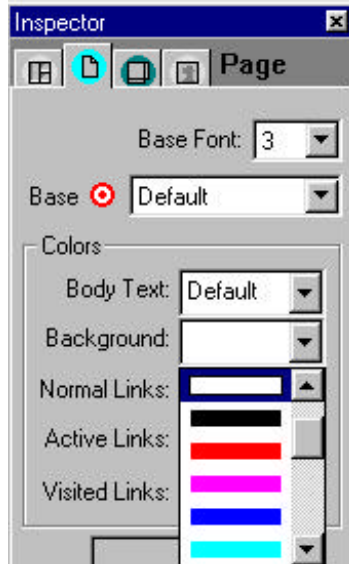
Or, suppose we want a word to be rendered on screen in a boldface font. The HTML tag for this is ``. We simply highlight the word we want to appear in bold using the mouse, and click on the `` tab on the toolbar. The editor supplies the `` and `` tags before and after the text in question.

This is an example of an editor that doesn't try to hide HTML from you. Other editors fall into the "What You See Is What You Get" (or WYSIWYG) category. If we used Adobe's Pagemill to edit the same file, by default we would not see HTML tags at all: we'd see the text rendered on screen pretty much in the same way a Web browser would render it:



Note that a separate window labeled "Inspector" appears on screen. The Inspector allows you to manipulate the look of your Web page without memorizing how to accomplish the task with tags. For instance, we've used the Inspector to change the default background color of our page. To do this, we simply click on the "pick list" next to Background, and click on the color we want for our page's background.

Building a Community Information Network: A Guidebook

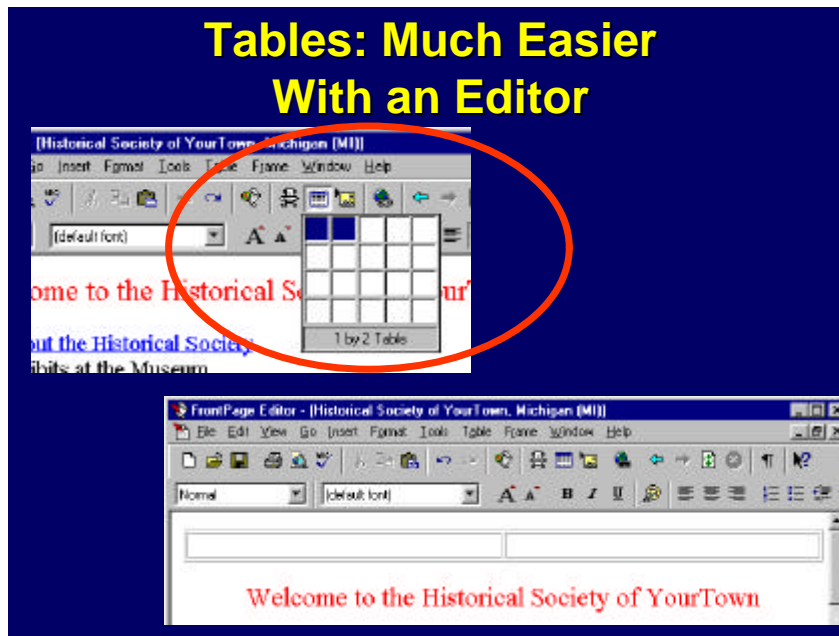


If we were editing by hand, we'd have to know the HTML codes for specifying a background color. This happens to be an addition to the `<body>` tag, namely `bgcolor="#ffffff"`. The "ffffff" is a *hexadecimal* code to specify the color's red, green, and blue components; we've specified full intensity for each component color, which yields white. Many people find it far easier to let the authoring tool handle all these details.

Here we've switched Pagemill from "preview" mode to "source" mode so we can inspect the HTML it is producing for us:

```
<HTML>
<HEAD>
  <META NAME="GENERATOR" CONTENT="Adobe PageMill 3.0 Win">
  <TITLE>Smallville Community Information</TITLE>
</HEAD>
<BODY BGCOLOR="#ffffff">
```

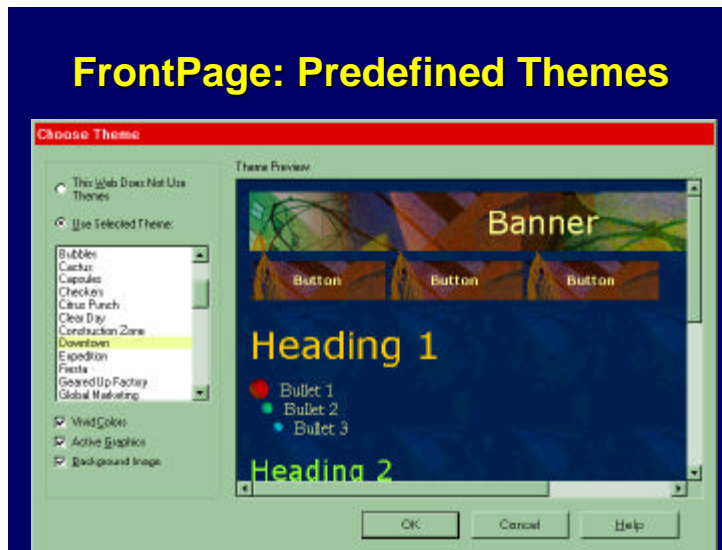
Some important HTML features, such as tables, are very difficult for humans to code by hand. An authoring tool can make it much easier to handle tables, allowing you to create a table just as you might in a word processor, using a table editor, as we see in this example with FrontPage:



The FrontPage table editor allows you to create the table with a graphical interface. You use the mouse to drag the table to contain the number of rows and columns you desire. FrontPage in turn will create the necessary HTML tags to define the table, which can be somewhat difficult to code by hand. Many authoring tools contain a graphical table editor.

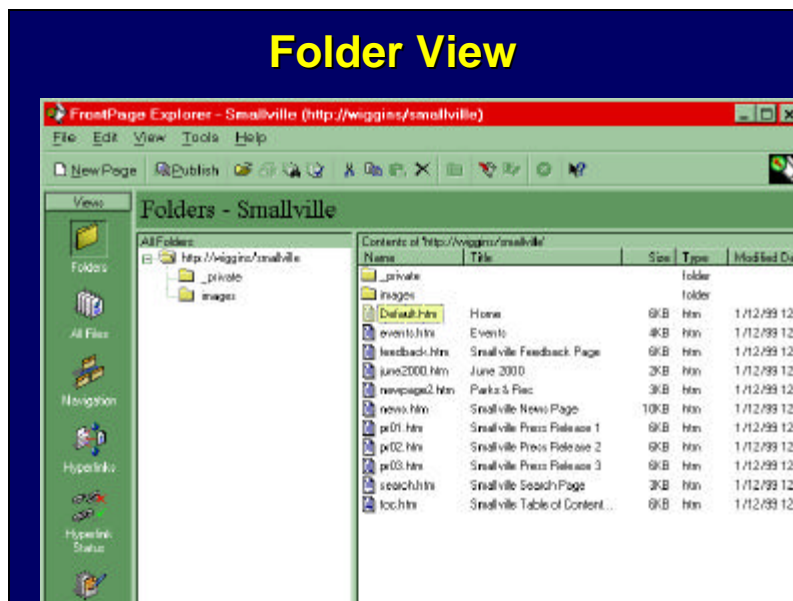
Similarly, HTML forms can be difficult to create. Here we see how Pagemill can produce an HTML form for us. On the left is the HTML code produced; on the right is the editing panel we interact with:

Authoring tools may offer predefined themes or templates that allow you to impart a look and feel to a page or a site without having to carefully design every element of your layout. While these approaches may have the sameness one sees when everyone uses the same clip art, they can save time and yield reasonably attractive results. Here is a page from a standard FrontPage theme:



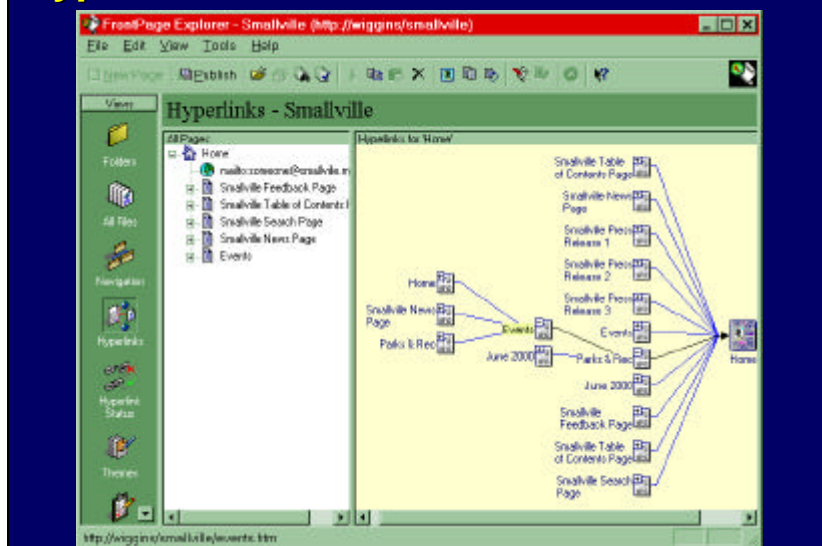
Increasingly, authoring tools are becoming site managers, not just page managers. For instance, FrontPage allows you to edit an entire “web” locally on your own PC, then publish that entire Web to the server. FrontPage even provides multiple views, or perspectives, so that you can examine the relationship among your files locally.

For instance, here is the Folder view in FrontPage:



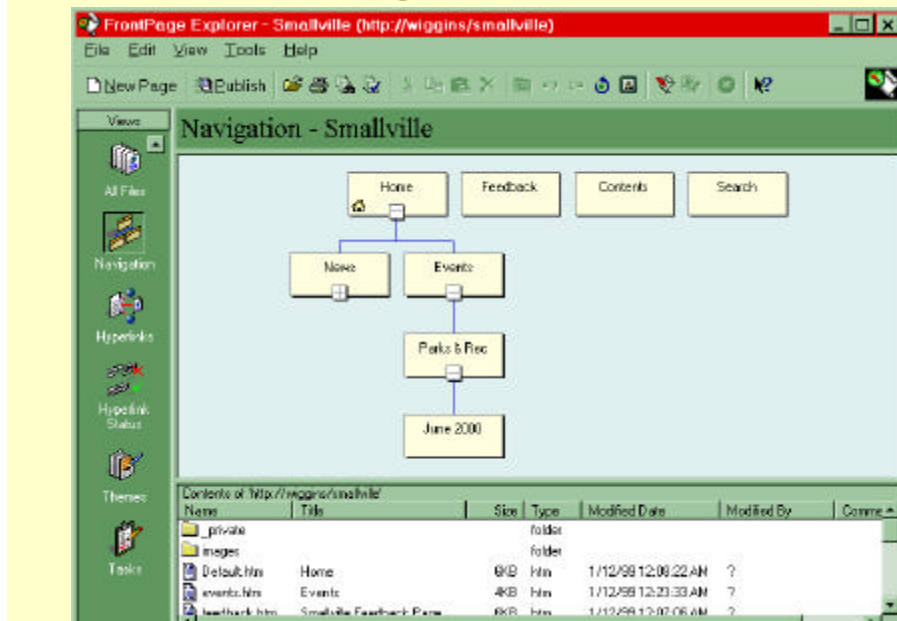
By contrast, the Hyperlink view lets you see in a graphical form the hyperlinks you’ve created and the linkages they imply among all the various pages that make up your site:

Hyperlink View – Visualize Internal Links



The Navigation view allows you to create menu bars in FrontPage and change relationships among pages using a graphical interface: update the linkages among pages, and FrontPage automatically updates the associated menu bar items.

Navigation View



Advanced Authoring and Site-Management Tools

Building a Community Information Network: A Guidebook

Over time, Web authoring tools have become more sophisticated, and new, highly specialized tools have evolved. In fact, it's difficult to keep up with the huge array of commercial products now available.

Authoring tools offer basic advantages to the content provider:

- They can save keystrokes and improve productivity.
- They can help you generate "correct" HTML. Because HTML is a structured language of sorts, it's possible to have errors, such as an `` tag that's opened but never closed. Some authoring tools include HTML verifiers to test your existing code; others check your HTML as you edit it.
- Specialized tools can even generate code for you. For instance, some tools will generate the HTML and JavaScript code needed to implement a set of "rollover" menu buttons.

Authoring tools may have shortcomings as well:

- Some tools are hard to learn. They may be very powerful, though; eventually they multiply productivity, justifying the up-front learning effort.
- Tools tend to insert excessive tags as they perform automated layout control. It can be especially frustrating to open a hand-crafted page, edit it in an authoring tool, and find that dozens of new tags have been introduced into your document – tags you don't want or need. *Make a copy of your hand-crafted pages before letting an authoring tool edit them.*
- Unfortunately, some tools sometimes generate "incorrect" HTML – HTML with syntax errors, or HTML that's tuned to only one browser.

You may already have a satisfactory beginning tool on your computer. Netscape Communicator includes Netscape Composer, a fairly powerful WYSIWYG editor. Microsoft ships FrontPage Express, a stripped-down version of FrontPage, with Windows 98 and with many other products.

Authoring tools fall into a variety of categories. Here are a few examples of categories and some tools within each:

HTML Editors

| Tool | Vendor | Vendor Web Site | Notes |
|---------|------------------|---|--|
| TextPad | Helios Software | http://www.textpad.com | A simple text editor alternative to Windows Notepad, with a link to an HTML validator. |
| HotDog | Sausage Software | http://www.sausage.com | A very popular HTML editor. |

Building a Community Information Network: A Guidebook

| | | | |
|--------------------|----------------|---|---|
| HoTMetaL | SoftQuad | http://www.softquad.com | An HTML editor from a company with a long history in markup tools. HoTMetaL is very careful to generate "correct" HTML. |
| PageMill | Adobe | http://www.adobe.com | A popular WYSIWYG editor for Windows or Macintosh. |
| Dreamweaver | Macromedia | http://www.macromedia.com | A visual page layout tool with ability to generate HTML that uses Cascading Style Sheets for precise layout control |
| HTML Assistant Pro | Brooklyn North | http://www.brooknorth.com | One of the first HTML editors; simple and easy to use |

Graphics Editors and Illustration Tools

A number of tools exist to simplify the job of creating and manipulating high-quality graphics for your Web site. Chapter 6 discusses graphics in general and some specific graphics editors. Here we list some advanced tools that carry the task of creating Web graphics to a high level.

Many of these tools can assist you in creating "image maps" – images with clickable objects that map to specific pages. See Chapter 3 for an explanation of image maps.

Most image editing tools are "pixel-oriented." Some of the newer tools are "object-oriented." We discuss the distinction in Chapter 6.

| Tool | Vendor | Vendor Web Site | Notes |
|-------------|-------------|---|---|
| WebRazor | Ulead | http://www.ulead.com | A tool to edit and optimize images and create animated GIFs. |
| DeBabelizer | Equilibrium | http://www.debabelizer.com | A popular image editor and optimizer for the Macintosh and for Windows. |
| Fireworks | Macromedia | http://www.macromedia.com | A tool to edit and optimize images and create animated |

Building a Community Information Network: A Guidebook

| | | | |
|-------------|-------|---|---|
| | | | GIFs. Object-oriented (editable unit is an object) |
| PhotoShop | Adobe | http://www.adobe.com | The industry-leading image editing program; used for both Web and non-Web publishing applications. Sophisticated, can take time to learn. |
| PhotoDeluxe | Adobe | http://www.adobe.com | An image editing tool from Adobe with many of the capabilities of Photoshop in an easy-to-use package. |
| ImageReady | Adobe | http://www.adobe.com | A tool to edit and optimize images and create animated GIFs; available for Windows or Macintosh. Pixel-oriented. |
| ImageStyler | Adobe | http://www.adobe.com | An object-oriented alternative to ImageReady. |
| Illustrator | Adobe | http://www.adobe.com | A sophisticated tool for drawing and creating original graphics |
| PhotoPaint | Corel | http://www.corel.com | A competitor to PhotoShop; bundled with image drawing tool. |

Database / Middleware Tools

This family of tools goes beyond preparing HTML or graphical content into a different dimension: these tools provide ways to link your Web site to an external database, and/or they generate programs for you using a graphical development environment. These tools tend to be quite powerful. They also tend to require much more investment in learning the tool than a simple HTML editor would require.

Database integration is a logical “next step” to take after an initial venture into Web publishing. With database integration, your Web users can use their familiar browser environment as a direct window into “live” data. The Toolkit calendar application, for instance, connects a Windows NT server running Microsoft’s Internet Information Server (IIS) software to a back-end database running under Microsoft-Access or SQL. One user can update a calendar

Building a Community Information Network: A Guidebook

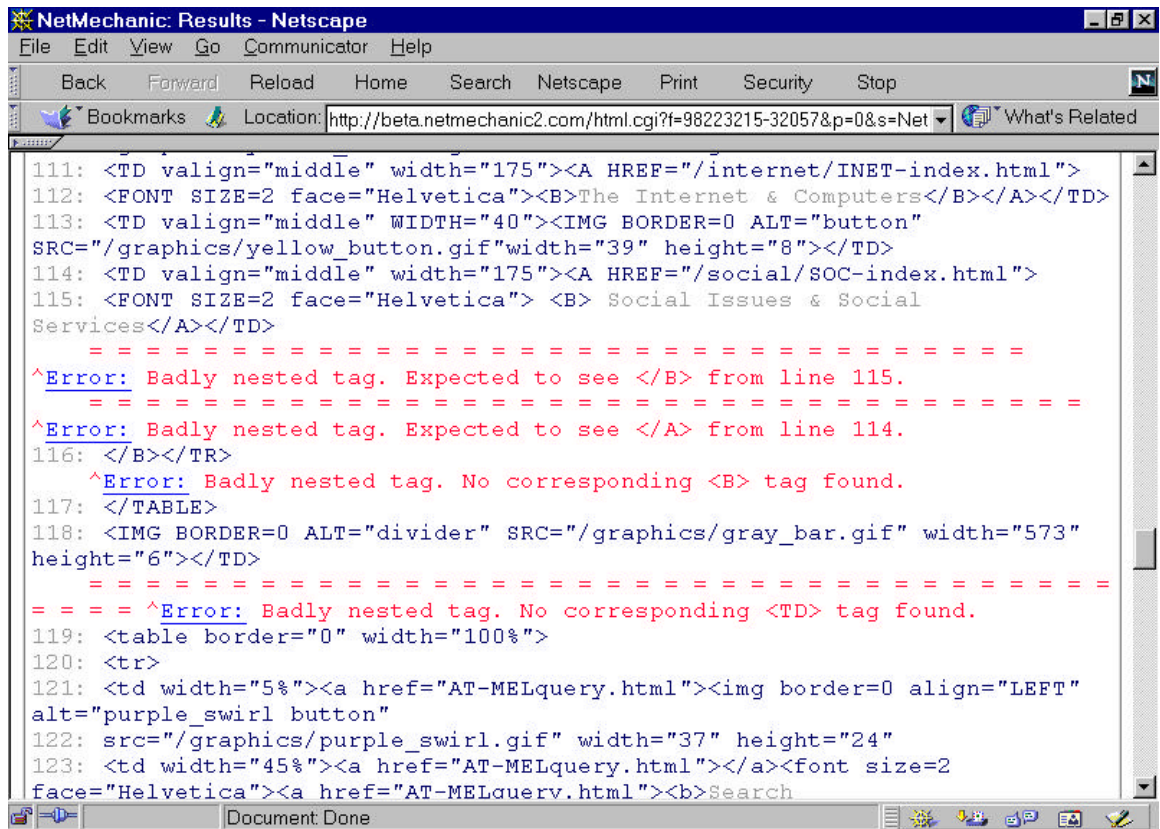
entry using a Web browser, and another can see the update using his or her Web browser, with neither having the slightest concern as to how the data is handled in the server. (The Toolkit software components were developed using Microsoft's development tool, Interdev.)

| Tool | Vendor | Vendor Web Site | Notes |
|-------------------|--------------------|---|--|
| Drumbeat 2000 | Elemental Software | http://www.drumbeat.com | A graphical Web-to-database development environment for connecting your server to Active Server Pages (ASP) |
| Interdev | Microsoft | http://www.microsoft.com | Microsoft's graphical development environment for ASP databases. |
| NetObjects Fusion | NetObjects | http://www.sausage.com | A WYSIWYG (graphical) site-building environment that can generate HTML pages and linkages to live databases. |
| Cold Fusion | Allaire Corp. | http://www.allaire.com | A database connectivity tool; author inserts Cold Fusion Markup Language statements in the middle of HTML documents. Popular for simple database integration applications. |

Free Web-Based Authoring Assistants

There are many services available on the Web that will help you verify the correctness of your HTML pages, or even optimize the size of your GIF images. (To "optimize" an image is to increase its compression factor, minimizing size and download time, without unacceptably degrading image quality. See Chapter 6 for more details on image optimization.) For example, NetMechanic (www.netmechanic.com) will examine a page or portion of an entire site, generating a report for you, showing line-by-line which HTML statements may have syntax errors.

Building a Community Information Network: A Guidebook

A screenshot of a Netscape browser window titled "NetMechanic: Results - Netscape". The address bar shows the URL "http://beta.netmechanic2.com/html.cgi?f=98223215-32057&p=0&s=Net". The main content area displays HTML source code with several red error messages. The errors are: 1. "Badly nested tag. Expected to see from line 115." 2. "Badly nested tag. Expected to see from line 114." 3. "Badly nested tag. No corresponding tag found." 4. "Badly nested tag. No corresponding <TD> tag found." The code includes various HTML tags like <TD>, , , <A>, , <TABLE>, and <TR>. The browser's status bar at the bottom shows "Document Done".

```
111: <TD valign="middle" width="175"><A HREF="/internet/INET-index.html">
112: <FONT SIZE=2 face="Helvetica"><B>The Internet & Computers</B></A></TD>
113: <TD valign="middle" WIDTH="40"><IMG BORDER=0 ALT="button"
SRC="/graphics/yellow_button.gif"width="39" height="8"></TD>
114: <TD valign="middle" width="175"><A HREF="/social/SOC-index.html">
115: <FONT SIZE=2 face="Helvetica"> <B> Social Issues & Social
Services</A></TD>
=====
^Error: Badly nested tag. Expected to see </B> from line 115.
=====
^Error: Badly nested tag. Expected to see </A> from line 114.
116: </B></TR>
      ^Error: Badly nested tag. No corresponding <B> tag found.
117: </TABLE>
118: <IMG BORDER=0 ALT="divider" SRC="/graphics/gray_bar.gif" width="573"
height="6"></TD>
=====
===== ^Error: Badly nested tag. No corresponding <TD> tag found.
119: <table border="0" width="100%">
120: <tr>
121: <td width="5%"><a href="AT-MELquery.html"><a href="AT-MELquery.html"></a><font size=2
face="Helvetica"><a href="AT-MELquerv.html"><b>Search
```

The W3 Consortium offers an official validator to check your pages against the HTML 4.0 specification at validator.w3.org.

One Authoring Tool, or Many?

Some HTML authors find that one tool is better than another for a particular task. They may like the table editor in FrontPage and the forms editor in PageMill.

While authoring tools can be a huge boon, it's prudent not to mix and match too many tools on a single page. Actually, some experienced Web authors will use multiple tools to create *parts* of a single page – one tool to create a table, another to build a fill-in form – and then stitch all the pieces together into a single physical file. Beware the potential pitfalls of this strategy:

- If you use more than one authoring tool to create a page, it becomes difficult to keep up with where the “real” original page resides. Is it the HTML file you put on the server? Or is it each of the original fragments you've built in the various tools' environment?
- If one of the tools manages an entire site (e.g. FrontPage) it may be tricky importing the HTML from the other tool in order to publish everything as a coherent part of the site.

Building a Community Information Network: A Guidebook

- Because authoring tools tend to add their own flavor of tagging, editing entire files successively in different tools can be especially prone to yielding mish-mash.

As a team, you'll also want to consider whether you want to standardize on a single tool. You may find it hard to manage your site if every content provider uses his or her favorite editor.

In practice this is easier said than done, however. The choice of a favorite editor is an extremely personal, subjective decision, and you may find it hard to get everyone to agree on a single tool.