

WebEdit v1.2 Online Documentation -- Contents

WebEdit 1.2 Release Notes

Copyright And Distribution Notice

About Ivory Tower Software

About WebEdit

The WebEdit Interface

Using WebEdit As A Text Editor

Using WebEdit As An HTML Editor

Configuring WebEdit

Project Files

Setting Up A Project Area

Setting Up Project Files

The HTML Assistant

About WebEdit

WebEdit is a Visual BASIC application. In fact, it is based on the standard demonstration MDI application that comes with VB 3.0 Professional. I've added a bunch of features and cleaned up some bugs and incomplete code in that application. Additionally, the product has some smarts with respect to HTML.

This Help File was constructed with ForeFront's ForeHelp Windows Help development environment.

Using WebEdit As A Text Editor

WebEdit works just the way you'd expect it to as a straight text editor. If you understand how to use Notepad, nothing about WebEdit will surprise you.

You should be aware, however, that WebEdit has some advantages and some limitations over NotePad and other Windows text editors.

Limitations:

1. Text files above 30K in size can't be edited (this is the result of a limitation in Visual BASIC)
2. Printing is somewhat primitive; you can select the default printer, but that's about it (this is due to laziness on my part)

Advantages:

1. WebEdit is a multiple document interface (MDI) application, which means you can have multiple text files open at the same time.
2. WebEdit understands the notion of a project. If you invoke WebEdit with a filename as its argument, WebEdit will open the file you specify, read filenames from it, and automatically open every file it finds in the file you specified that actually exists. You can even open the project file itself, and save it as part of the project.
3. WebEdit will let you insert files into open files (something I always wanted Notepad to do).

Using WebEdit As An HTML Editor

I wrote WebEdit for friends of mine who are trying to do Web publishing with little or no background in UNIX-style text processing or Standard Generalized Markup Language. I tried to find a freeware editor that provided a relatively easy-to-use interface for HTML creation, but all the ones I found assumed you already knew exactly what you wanted to do, and understood all the crabbed HTML codes and coding conventions.

When you use WebEdit as an HTML editor, it tries to help you create sane HTML.

If for example, you highlight a block of text, and then select Insert -> Typeface -> Bold, WebEdit will wrap the HTML coding around the highlighted text. This allows you to create the document using flat text (no HTML coding) and then go back and add the coding you want later.

WebEdit also helps you create correct URLs and image source declarations. For some people, the forms Webedit steps you through will be annoying. This is a clear sign that you don't need WebEdit's help anymore.

Finally, WebEdit lets you maintain project make files that name a set of related HTML documents, one per line. If you pass the MAKE file name as the argument to WebEdit when it is executed, it will open all the files in the MAKE file automatically.

The File Menu

The file menu options allow you to print, save, open and close files. The file menu also lists the last ten files you opened for quick access.

About Ivory Tower Software

Since I don't do much technical work as part of my job any more, I dink on the side to keep my hand in and try to solve some of the millions of annoyances that come with the Windows environment. Some of these dinkings see the light of day as software -- this is one of those.

All Ivory Tower software products are free. You can distribute them as you like as long as you distribute them *in toto* (for example, sending the help file along with the binary). **Support, however, costs money.** If I know you, or you've paid me the \$10 per year that it costs to get support, bug fixes and enhancements for all Ivory Tower software products (yes, \$10 US for support on 1, 2 or a dozen products), I'll answer your questions and fix what's busted. Otherwise, KMACYOYO -- this is free software.

Send those checks to:

Marc Demarest
3836 NE 63rd Avenue
Portland, Oregon 97213

demarest@hevanet.com

Configuring WebEdit

WebEdit gets a minimal amount of information that it needs to be sensible from a file called WEBEDIT.INI, which is located in your default WINDOWS directory (usually C:\WINDOWS).

The typical WEBEDIT configuration file will look something like this:

```
[User Information]
User Name=Marc Demarest
Address=marc@sequent.com

[Application Information]
Version Number=Release 1.1
Installation Date=01-22-1995
Release Date=January 1995
Product Version=Freeware Personal Edition
Serial Number=00000001
Product Name=WebEdit -- An HTML Editor for Windows
Short Product Name=WebEdit v1.1

[Tools]
Browser=C:\NETSCAPE\NETSCAPE.EXE
Graphics Editor=C:\windows\LVIEW31.EXE
Hotspot Editor=C:\windows\hotspot.EXE
Stripper=C:\vb\projects\WEBEDIT\HTMLSTRP.EXE
Text Editor=C:\WINDOWS\notepad.exe
Mailer=C:\WINDOWS\MSMAIL.EXE
Publisher=\\ELMO\SHARED\EPUBS\PUBTOOLS\INDIVPUB.EXE
Word Processor=C:\WINWORD\WINWORD.EXE

[User Assistance]
Prompts=YES
Warnings=YES

[Recent Files]
RecentFile1=C:\TEMP\HTMLPRIM.HTM
RecentFile2=C:\TEMP\TEST3.HTM
RecentFile3=C:\TEMP\TEST3.HTM
```

To change your configuration, select **Tools --> Edit WebEdit** Configuration from the pulldown menu. Make sure you specify complete path names when you change binaries.

Setting Up Project Files

WebEdit can use project (.WEP) files to automatically load a set of files you work on a a group.

The WEP file can have one of two formats:

Format 1

```
filename.1  
filename.2  
filename.3  
filename.4  
filename.5
```

Format 2

```
Directory=DirName  
File2  
File3  
File4
```

Format 1 is just a flat list of file names (with their complete path specification), one per line. This is the **recommended format**.

Format 2 sets a directory in which all files may be found, and then lists the file names without a path specification (because all files named should be in the directory specified in the Directory= line)

You can create and maintain project files with WebEdit. You can (and probably should) include the name of the project file in the project file, so the project file is opened with the rest of the project, as in:

```
c:\windows\webedit.ini  
c:\windows\msmail.ini  
c:\windows\localsig.txt  
c:\temp\test.prj
```

In this example, the file test.prj is the file containing this list.

You can add comments to the file by using the standard semi-colon comment convention -- a semicolon as the first character will mark the line as a comment.

Titles

Every HTML document should have a title. A title is generally displayed separately from the document and is used primarily for document identification in other contexts (e.g., a WAIS search). Choose about half a dozen words that describe the document's purpose.

The title of your document is used by the browsers that interpret it and by Web search tools that try to discover your pages for use by others. Take the title seriously and make it clear and descriptive.

This is a good title:

```
<TITLE>Joe Batz's Electronic Commerce Web Site  
Listing</TITLE>
```

This is not a good title:

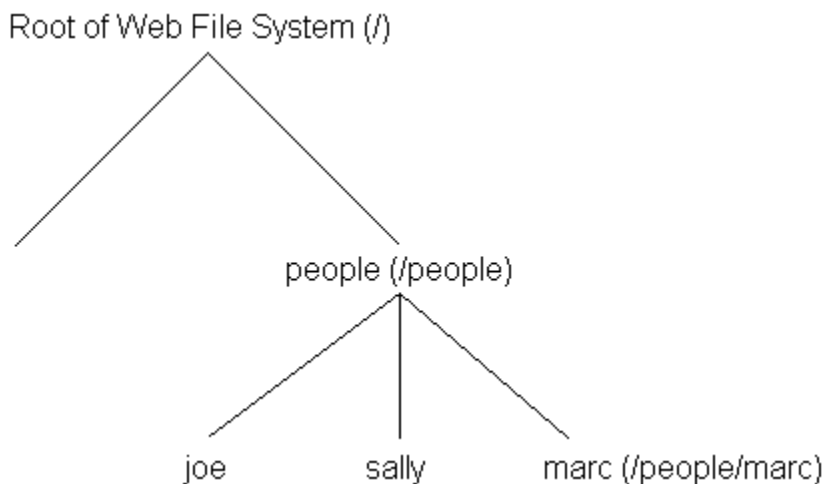
```
<TITLE>My Home Page</TITLE>
```

Setting Up A Project Area

If you are going to write significant amounts of HTML - more than, say, a simple home page for yourself -- and you are not using a source code control system or other publishing system that provides some discipline to your work, there are a few things you can do to make your life as an HTML developer easier.

First of all, remember that you are going to be adding files and directories onto the Web directory structure of a server. This directory structure is used by many people and organizations; your work will be only a part of it. Your Web administrator will give you an attachment point in this directory structure, and your work will, for the most part, live "below" that attachment point in the directory structure.

For example, consider this directory structure:



My Web administrator (often called the WebMaster) has told me that I can do anything I like to the directory structure below /people/marc -- it is my space to create my own part of the Web. She has also given me read and write permission in that area so I can create directories and files.

It is not generally a good idea to compose HTML in this space. This space is being examined regularly by Web browsers (acting through your local Web server), and, if you have moved, removed or opened and modified a file, the people reading your part of the Web may get garbage in their viewers. For that reason, I would recommend that you set up a separate project area elsewhere, and copy over your HTML files to the "production" Web directories only after you have (a) written, (b) tested and (c) debugged them.

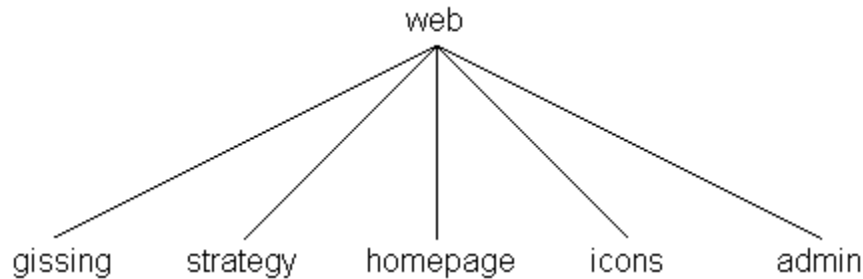
Let's assume, in the example above that I decide to have three kinds of Webs I will work on:

- my home page

- a set of pages with pointers on them referring to Web sites in my specialty (strategy and organizational design)

- a set of Web pages on George Gissing, a Victorian novelist I am interested in.

In my project space, I would create a directory structure like this:



The gissing, strategy and homepage directories are obvious choices, since I have a specific Web in mind for each.

The icons directory is there because I personally like to keep all my GIF and JPG files in one location; I tend to use them over and over again, and if I put them in a common directory, I won't have duplicate copies of the same pictures scattered across my Web directories. This does however make writing inline pointers to those pictures more complicated.

The admin directory I always include because there are some files that, while not perhaps part of the Web, are either pointed to by Web pages or capture information about those Web pages. In some projects I have done, the admin directory has been almost empty; in others, it has been chock-a-block full.

The web directory at the top is equivalent to the marc directory in the production Web directory space given to me by my WebMaster. So, for example, when I put the GissingWeb into production, it will be found at <http://www.sequent.com/people/marc/gissing>. The StrategyWeb will be found at <http://www.sequent.com/people/marc/strategy> and my homepages will be found at <http://www.sequent.com/people/marc/homepage>.

In each of the directories, gissing, strategy and homepage, I immediately create two files:

- index.html, which is the top-level Web page for that Web

- updates.html, which will contain a revision summary of the work done on that Web.

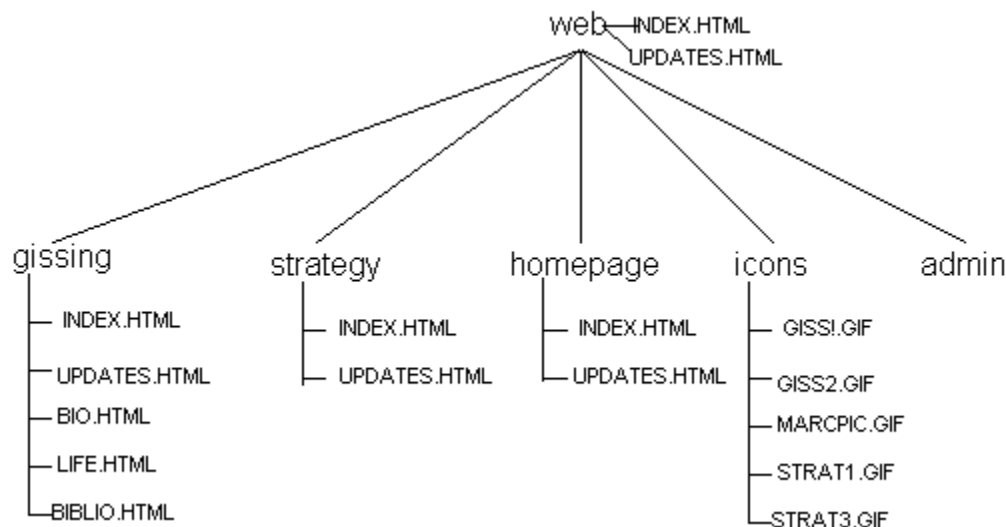
I ALWAYS call the file INDEX.HTML because that is the file a browser will look for automatically if given only a directory structure. I ALWAYS include an UPDATES.HTML file for each Web so that my readers who have been through my Web before can find out quickly and easily what has changed.

As I write the HTML files, I test them using several browsers in the project area. Since almost all browsers can read files from disk as well as from the network, I don't need to move the files into the production Web space and access them via HTTP to test them.

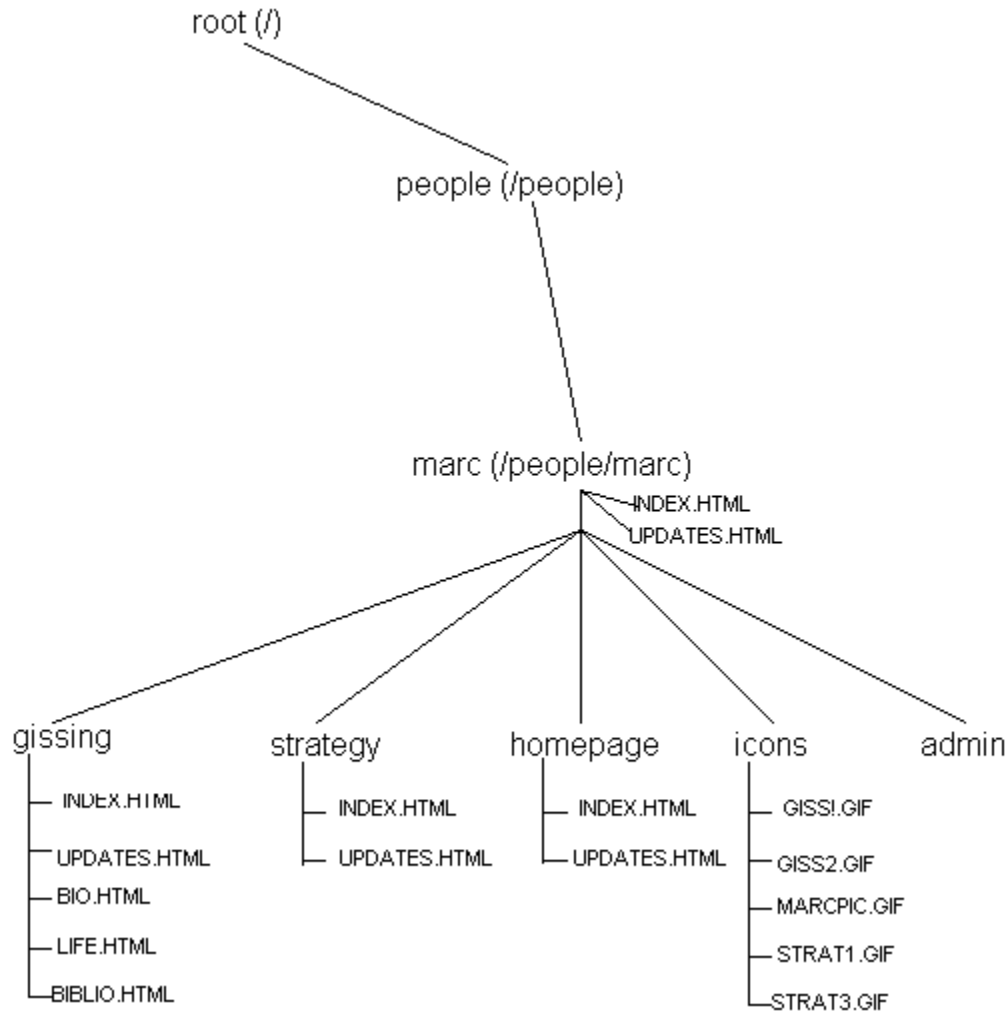
I think you need to test everything -- right now at least -- with two browsers: NetScape, which is the most popular (and robust) browser for the Web, and one of the browsers based on NCSA Mosaic (like AirMosaic from Spry). When I first started publishing Web material, I only tested with NetScape, and I discovered that netScape is VERY FORGIVING of errors, including unclosed tags, overlapping tags and other problems. It "figures out" what the author meant, and does it right most of the time. The other browsers I am familiar with are not forgiving; they have all kinds of strange failure modes, including failing to display anything at all. I learned that pages I had published -- pages I thought very well designed -- appeared as garbage or as blank nothingness to other people...just because they had different browsers.

Once I have written and tested a Web with at least two browsers, I put that Web into production by copying the entire directory out of my project space and into production space. [And, because of the way my WebMaster has allocated me space, I add a file INDEX.HTML under /people/marc, pointing to my three webs.]

What I end up with looks something like this, in my project area



and like this in my WebMaster's production directory area:



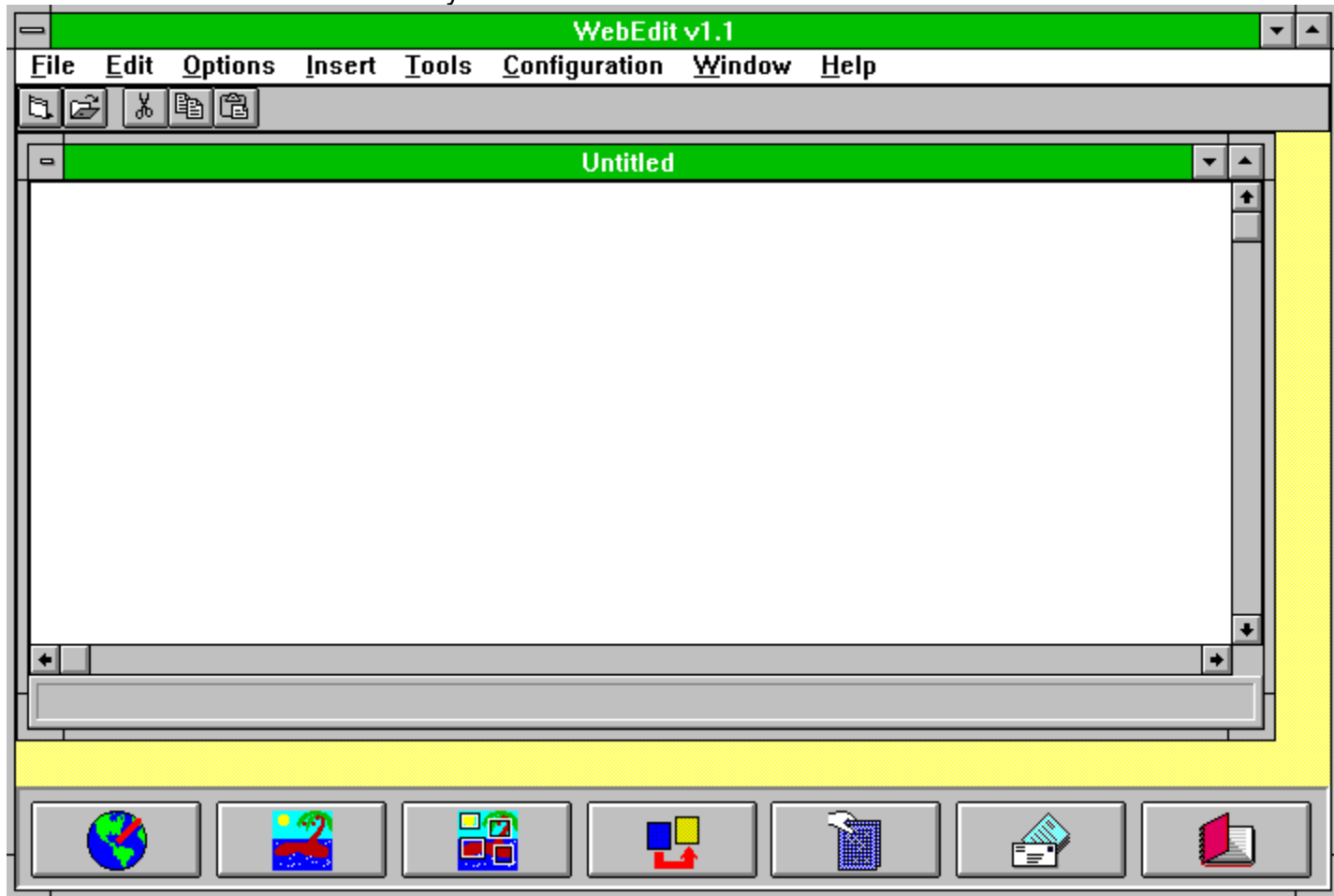
There are some files in my project area that never get copied into the Web production directories, like the .WEP files for WebEdit that keep all the files under gissing, strategy and homepage together (and loadable into WebEdit at one time).

As I make changes to my Webs, I make them in my project area, copying the project area over to the production area on a regular basis (once a week or once a month), whether I have made a lot of changes in the preceding period, or only a few. And, of course, the UPDATES.HTML files in each area of the Web keep a simple summary of the changes made, so my periodic readers can see if they need to revisit my Web areas or not.

The WebEdit Interface

Although the WebEdit interface is fairly simple, you may have questions about particular functions or areas of the interface.

Click on the area of the interface you need information on.



Copyright And Distribution Notice

This product is freeware and is distributed without warranty of any sort. If you want support, see the [Support Notice](#).

This hypertext is freeware

No warranty, expressed or implied, accrues to this product. Use it at your own risk.

Some of the information in this hypertext will eventually be replaced by a more complete commercial product. This BETA TEST preprint may be distributed, in its original binary format with all information intact, without restriction as long as it is distributed without charge.

The author reserves all rights to the portions of the hypertext he authored: the text, the examples and illustrations and the design formats. The remainder of the document is under the control of its authors, who are identified.

WebEdit 1.2 Release Notes

This version of WebEdit fixes two bugs:

1. There was in WebEdit 1.1 a transient error on File-->New invocation. It is fixed.
2. The installation for WebEdit 1.1 didn't work properly when the target system had no MAPI messaging system.

The distribution for WebEdit 1.2 has two possible installs:

- NET, which installs the network (MAPI) version of WebEdit
- STD, which installs the standalone (NO MAPI) version of WebEdit.

The Edit Menu

The edit menu allows you to cut, copy, paste, select, find and insert date/time stamps.

The Options Menu

The Options menu lets you turn the utility bar and the toolbar on and off.

The Insert Menu

The insert menu provides you with:

the ability to insert a file from disk into a document you are working on.

insert and access HTML code insertion capabilities.

If you are using WebEdit as a straight text editor, you won't use the insert menu functions much.

The Tools Menu

The tools menu reads information in WEBEDIT.INI and provides you with menu-based access to your Web browser, graphics editor, map editor, HTML publishing tool, and other important "helper" applications. These same applications are accessible via the Toolbar.

The Configuration Menu

The configuration menu lets you edit the WEBEDIT.INI file and change the default font used in the text edit window.

The Window Menu

The window menu lets you arrange the open windows in various forms.

The Help Menu

The help menu invokes the help files you are using right now.

Text Entry

All text entry is done in this area.

You can edit files up to 32K in size.

File Status

Browser Launch

This icon launches the browser configured in your WEBEDIT.INI file on the file in the currently active window.

Graphics Editor Launch

This button launches the editor specified in the WEBEDIT.INI file that you use to create JPG and GIF files for inclusion in HTML files.

Hotspot Editor Launch

This button launches the MAP editor configured in your WEBEDIT.INI file so that you can create coordinate maps for graphical access within HTML.

Stripper Launch

If you have a copy of HTMLSTRP (or some other HTML stripping application), and that file is called out in your WEBEDIT.INI file, this button will feed a copy of the HTML file in the currently-active window to that stripper, and display the stripped results in a new window.

Publisher Launch

If you use an application to publish HTML files to your Web server, and that publishing application is specified in [WEBEDIT.INI](#), you can invoke that tool via this button.

Mailer Launch

If you have an active MAPI-compliant mail system, this button will submit the text in the currently active window to the standard MAPI-embedded mail form, allowing you to send the file to someone else via e-mail.

Word Processor Launch

If you have specified a word processor in WEBEDIT.INI, this button will allow you to move an HTML file into that word processor.

If you are using Word for Windows 6.X, the application will use OLE2 to place the document text in Word for you. If you are using some other word processor, WeEdit will put the document's contents on the ClipBoard and launch the word processor -- you will have to paste the text in manually.

MDI

This is a multiple document interface (MDI) application. The main window allows multiple windows to be opened within it. This background is inactive.

Utility Bar

The utility bar, above the work area, provides you with button-based access to cut, copy, paste and other functions.

Toolbar

The toolbar provides you with icon-based access to the most important tools listed under the Tools Menu.

Document Title

The document title appears in this window. When you maximize a document window, the name of the document appears in the application title bar.

Enter the full path name of the Web browser you use here. WebEdit has been tested with NetScape, NCSA Mosaic, and Air Mosaic.

Enter your full internet mail address here.

Don't change this information; WebEdit will behave unpredictably if you do.

Enter the full path name of the graphics editor you use to create and edit GIF andf JPG files here.

Enter the hotspot editor you use here. WebEdit includes the mapedit.exe shareware application, which you must pay fo if you use!!

Enter the name of the application you use to convert HTML files to plain text here.
Webedit includes my HTMLSTRP application, which should be in the same directory
where you installed WEBEDIT.

This is currently unused.

If you use a publishing tool to publish your HTML files to your Web server, put the full path name of the application here.

Put the full path name of your word processor here.

This area will contain up to 10 of the last set of files you edited. You should not change this information.

Put your name as you expect it to appear in your HTML files here. WebEdit uses this name to automatically embed your name in HTML files.

