

---

---

# Understanding Visual Interdev

---

---

*FrontPage may be useful for creating static Web pages, but Microsoft's Visual Interdev can help you create complete, Web-hosted applications - for instance, an online shopping site. We take you on a guided tour.*

**By Mike Lewis**  
**Technical Journalist**

**V**isual Interdev is Microsoft's flagship Web development tool. As such, it goes well beyond the more familiar HTML authoring products, such as Microsoft FrontPage. With Visual Interdev, the focus is not on creating static Web sites, but on developing complete Web-hosted applications. If you wanted to build, say, an online shopping site, complete with interactive order forms, a searchable catalogue and secure payment options, Visual Interdev would be a good product to use. Visual Interdev is not an alternative to Java, JavaScript, Active Server Pages, CGI, Perl or any other Web-based language or technology. On the contrary, it lets users continue working with the language they know best. What Visual Interdev provides is a platform: a set of tools for editing, prototyping, debugging and deploying your Web-based applications.

In this article, I will introduce you to Visual Interdev and explain how the product can be put to work. Parts of the article will be tutorial in nature, so if you have a copy of Visual Interdev installed you might like to work along.

Visual Interdev is currently at version 6.0. In fact, this is only the second version of the product. Its predecessor was Visual Interdev 1.0, which it replaced in 1998. The jump in version number was accompanied by a huge increase in functionality, with version 6.0 being generally regarded as vastly superior to the first release. You can buy Visual Interdev either on its own or as part of Microsoft's Visual Studio suite (which includes Visual Basic, Visual C++, Visual FoxPro and several other development tools). In general, if you plan to buy any two of the components of Visual Studio, it will work out cheaper to buy the entire suite.

## ***Server Requirements***

To develop an application with Visual Interdev you need a connection to a Web server. Visual Interdev itself is server-independent, but one of its key features is its support for Active Server Pages (ASP). This in turn depends on specific servers. If you want to take full advantage of Visual Interdev, your application really needs to run on Microsoft's Internet Information Server (version 3.0 or later), preferably with FrontPage Extensions installed. It can also run on certain other servers, including Lotus Domino and O'Reilly, provided that these have a third-party ASP engine. If you don't have access to a server while you are developing the application, a good alternative is to install Microsoft's Personal Web Server on your workstation. Personal Web Server is a cut-down version of Internet Information Server (IIS). It would not be suitable for hosting a live application, but it is perfectly adequate for testing and prototyping. It can be installed on a computer without a network connection.

When you install Visual Interdev, the setup program invites you to install certain server components as well. These include the NT Option Pack for Windows 9x (which includes Personal Web Server for Windows 95/98), FrontPage 98 Server Extensions, Visual Interdev Server Components and Data Access Components. If you don't have network access to a computer running IIS, you will need to install at least the first three of these components in order for Visual Interdev to work properly.

## ***Projects***

When you develop an application in Visual Interdev, your main focus is on a project. Put simply, a project is a container for the Web pages, images, program code and other components that make up the application. It is strictly a developer concept:

the end-user is not aware of projects. Where more than one team member is working on the same application, they will typically share a single project. The project is stored in a file with the extension VIP. This does not contain any actual components; it is merely a list of their filenames and paths. At the next level up, Visual Interdev supports something called a solution, which is essentially a container for holding multiple projects. Solutions are held in files with the extension SLN.

### **Master And Local Copies**

When you tell Visual Interdev to start a new project, it creates an entire directory structure to hold the files that will make up the application. In fact, Visual Interdev creates two copies of this structure: a master copy on the server and a local copy on the workstation. How these copies are used depends on whether you choose to work in master mode or local mode. When you open a file for editing, Visual Interdev copies it down from the server, overwriting the local version. You edit this copy, with your changes being saved at the workstation. In master mode, the file is also copied back to the server when you save it. This ensures that the master copy is as up-to-date as possible. In local mode, the edited file is only copied back to the server when you explicitly release it. As well as being faster, this reduces the risk of

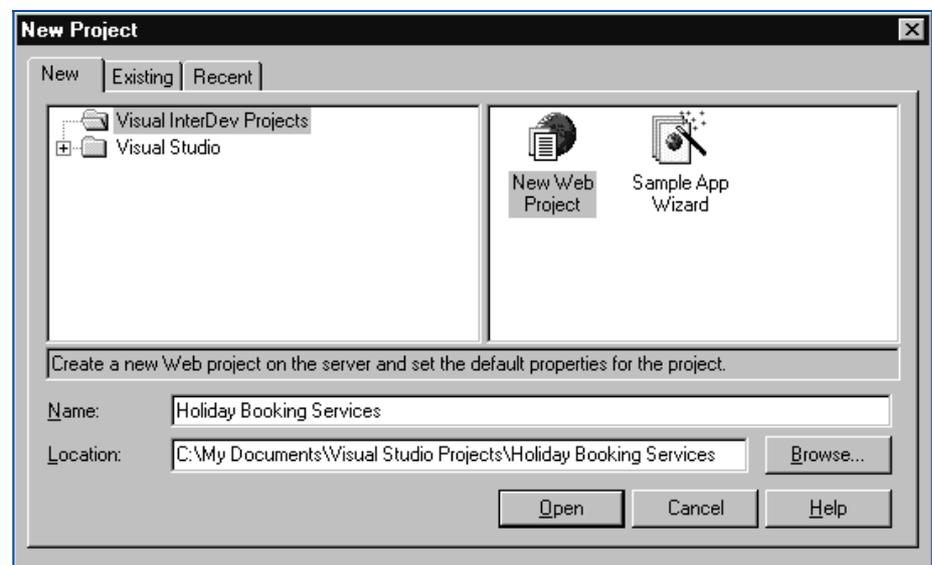


Figure 1 - The New Project dialog.

---

*“You make the initial choice between master and local mode when you create the project. You can subsequently change the mode, or go to offline mode, by selecting Web Project/Working Mode from the Project menu.”*

---

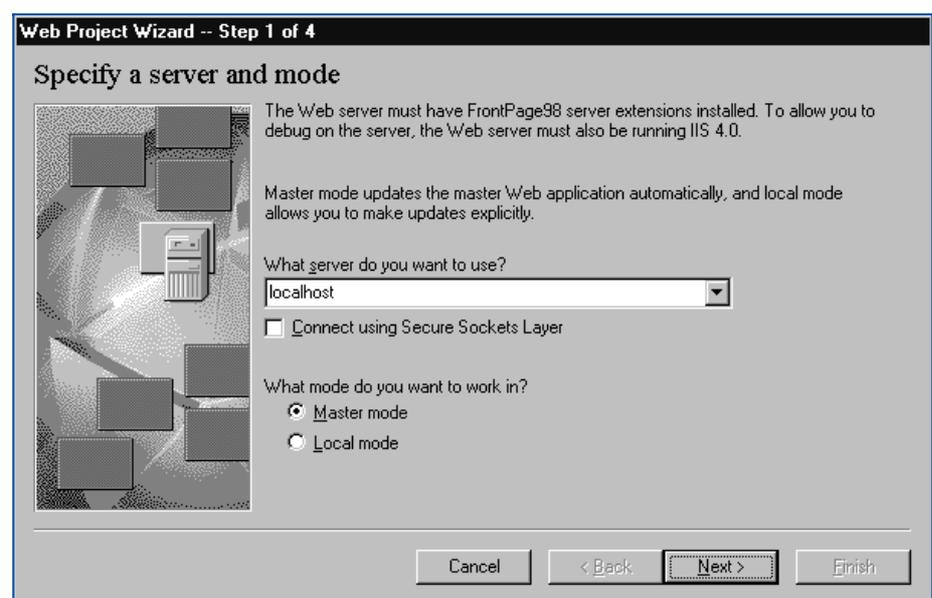


Figure 2 - The Web Project Wizard.

developers overwriting each others' edits. Local mode is also useful if you are modifying a live application, as it allows you to test your changes before making them generally available.

Visual Interdev also lets you work in offline mode. This is useful if you are temporarily without access to the server. In this mode, all editing is applied to local files. You can run the application as normal, except that you won't be able to use any functions which require resources on the server. When you go back online, all edited files are copied back to the server. You make the initial choice between master and local mode when you create the project. You can subsequently change the mode, or go to offline mode, by selecting Web Project/Working Mode from the Project menu.

### Creating The Project

Let's go ahead and create a project. The first step is to specify the project's name and its location on your workstation. You do this from the New Project dialog (see Figure 1), which you reach by selecting File/New Project. By default, this dialog also appears when you start a Visual Interdev session. When you finish the New Project dialog, Visual Interdev launches the Web Project Wizard. This starts by prompting you for the name of the Web server which is to host the master copy (see Figure 2). If the server is installed on your workstation, you should enter either localhost or the computer's network name at this point. For the remaining steps of the wizard, you can usually keep the default settings. However, in steps 3 and 4, you might want to choose a theme and/or a layout for the application. As we will see later, these determine the site's overall appearance and the positions of various controls within the pages.

### The Project Explorer

Once the Web Project Wizard has finished, you will see the project in a window like the one in Figure 3. This is the Project Explorer (if it is not visible, open it from the View menu or by hitting Ctrl-Alt-J). As you can see, the Project Explorer window contains a solution (called Holiday Booking Services in the illustration), which contains the project you have just created (also called Holiday Booking Services in this example). This in turn contains several folders and files. You use the Project Explorer to navigate the project and to perform operations on the files and folders within it. For example, you can open a file for editing, preview it in a browser, remove files from the project, apply a new theme or layout and release a file to the server. The commands to perform these tasks are available either from the Project menu or by right-clicking on the appropriate object.

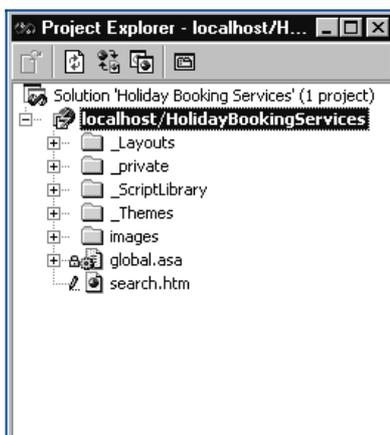


Figure 3 - The Project Explorer.

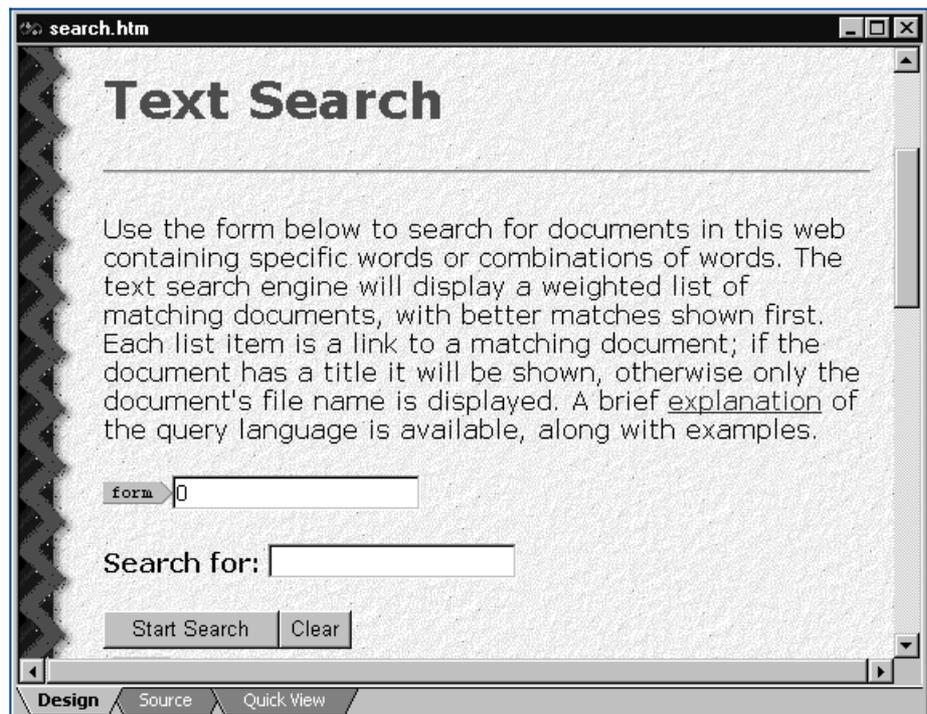


Figure 4 - The Visual Interdev editing window.

## Page Editing

When developing Web-based applications you will be spending a lot of time editing individual Web pages. These might be straightforward HTML documents, or they might be pages with embedded script. Either way, you open a page for editing by double-clicking on it in the Project Explorer window. Visual Interdev will then ask if you want to retrieve a working copy of the file from the server or open a read-only local copy for viewing - you would normally choose the former option. This choice is the same regardless of whether you are working in master or local mode. Figure 4 shows the SEARCH.HTM document open for editing in the Visual Interdev document window. This document, which provides a full-text search feature for your site, is generated automatically by the Web Project Wizard. You might not want to retain it for your application, but it does provide a useful sample for trying out Visual Interdev's editing features.

The editing window offers three ways of viewing the document. In Design View (shown in Figure 4), you directly edit the text, apply styles and formatting, and insert images. Source View lets you view and edit the underlying HTML or script code. Quick View is a non-editable preview, showing how the page might look in a browser. (To preview the page in an actual browser, select View In Browser from either the editing window's context menu or the View menu.)

You might like to try making a few changes to SEARCH.HTM at this point. Try editing the text in Design view, and also altering some of the fonts and colours. If you want to insert an image, you can do so by dragging a graphics file from the Windows shell and dropping it into the editing window. (In practice, you will store all your graphics files in the project's Images folder; this will enable you to drag images directly from the Project Explorer.) When you have finished editing the page, use either the File/Save command or the Save button on the main toolbar to save your work. This will save a local copy at your workstation. If you are working in master mode, the edited document will also be copied back to the server at this point. If not, you will need to explicitly copy it back in due course. To do so, right-click on the file in the Project Explorer and select Release Working Copy.

## Navigation

. However, Visual Interdev provides a much more sophisticated method - one which not only looks more professional but which is also easier to maintain as the structure of the site changes. To use this feature, you must first create a site diagram. This is a graphical device which helps you manage the structure of the site and the relationships between the pages. If your site has more than just a few pages, the site

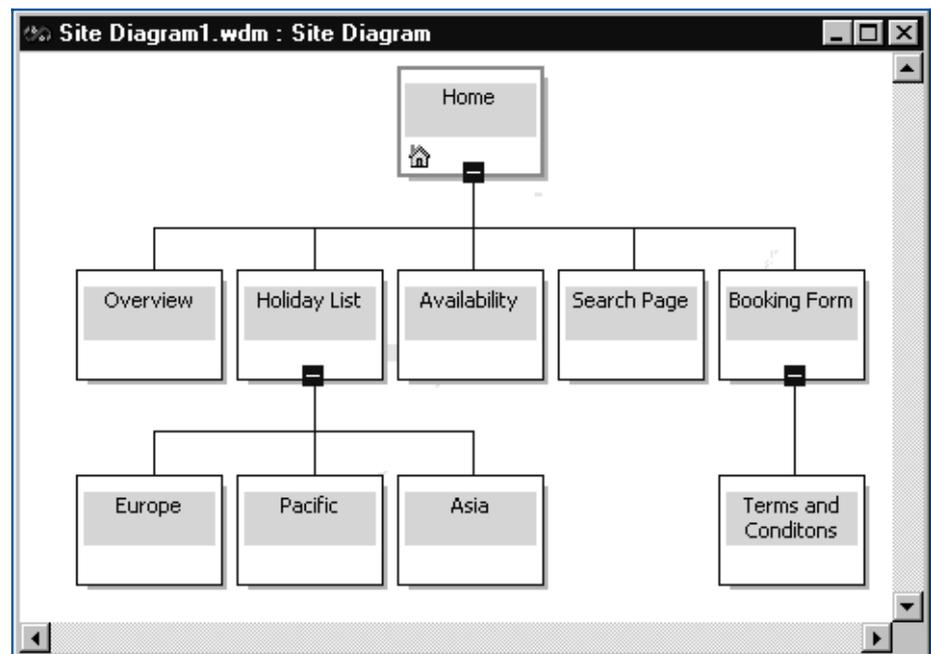


Figure 5 - A site diagram.

---

*“One of the most important features of a Web-based application is the ability to easily navigate between pages. In most Web sites, this is done by means of HTML hyperlinks.”*

---

diagram will save you a great deal of time and effort when you need to alter its structure in any way.

To create a site diagram, select the project in the Project Explorer, then select Add Item from the Project menu. In the resulting dialog, select Site Diagram and enter a name for the diagram. When you click on Open, a site diagram window will appear. You can now add your HTML pages to the diagram. You can either drag existing pages from the Project Explorer (try this with the SEARCH.HTM page), or you can create new pages directly in the diagram. For the latter, select New HTML Page from the Diagram menu. The page will be given a default name which you can then edit. Having populated the diagram with your pages, you now define their relationships. This is extremely easy. You just arrange the pages so that they form a tree structure. Figure 5 shows an example.

Next, you apply a theme and a layout to the pages (if you have not already done so). As I mentioned earlier, the theme determines the graphical style of a page, while the layout specifies the positioning of certain controls within the page. You can establish the default theme and layout from within the Web Project Wizard, or you can right-click on the project in the Project Explorer and select Apply Theme And Layout. If you want to override the choice for an individual page, right-click on the page and again select Apply Theme And Layout. Once you have applied a layout, you will notice something different about the editing window. This will now include several large grey rectangles. Some of these bear the caption "Add your content above" or "Add your content below". These are place-holders which show where you can insert your text and graphics.

Others are labelled PageNavBar - and these are more interesting. PageNavBars are a Visual Interdev device for providing page banners and navigation controls for the site. The nice thing about them is that they are almost completely automatic. To make them work, all you have to do is to ensure that a theme and a layout have been applied and that the pages have been positioned correctly within the site diagram. To see this in practice, just go ahead and preview one of your pages in your Web browser (right-click on it in the Project Explorer and select View In Web Browser). The result should look something like Figure 6. This page has the Redside theme and the Top 1 layout. The "Search Page" banner and the navigation bar below it are both derived from PageNavBars which Visual Interdev inserted and configured automatically. The navigation buttons are live - when you click on one of them, you will jump to the appropriate page.

Now let's see what happens when you change the structure of the site. Go back to your site diagram (you can open it from the Project Explorer) and add a new page. Move this into the tree structure so that it is one level down from the home page. Now save the diagram, switch back to the browser and re-load the current page.

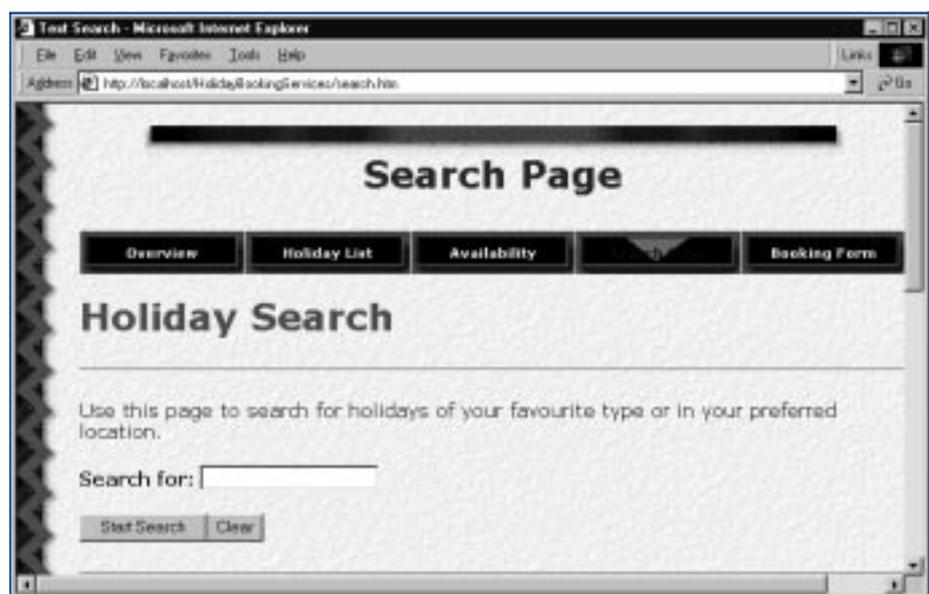


Figure 6 - This page includes a banner and navigation controls.

---

*“Visual Interdev is not an alternative to Java, JavaScript, Active Server Pages, CGI, Perl or any other Web-based language or technology.”*

---

You will see that all the pages in the site have been updated to reflect the new structure.

If the navigation bar does not provide exactly the links you want, it is easy to adjust it. To do so, right-click on the PageNavBar control in the document editing window, select Properties and, in the resulting dialog, make your choices from the Type and Additional Pages fields. These should be self-explanatory.

### ***Scripting***

So far, we have created a nice-looking site with some useful functionality. But it still falls short of being a complete Web-based application. To make it do anything useful, we need to add some intelligence to it. One way to do that is through scripting. There are two broad approaches you can take to scripting. You can write client-side script, which is embedded in HTML pages and interpreted entirely by the browser. Or you can write server-side script, which executes on the server and sends its results to the browser.

Client-side script is usually the better choice for processes that do not require server resources. In a customer order form, for example, you might want to validate the order quantity - to ensure that it is above a minimum permitted amount, for instance. By doing this with client-side script, you can perform the validation and display any resulting error message as soon as the user enters the value, without involving the server. But once the quantity has been accepted, you will probably need to make further checks against a central database. For example, you might need to ensure that there is sufficient stock to meet the order. Since the browser does not have access to the database, this part of the processing must be done in server-side script.

Client-side script is usually written in either Visual Basic Scripting Edition (also known as VBScript) or in JavaScript. VBScript is generally regarded as being easier to learn than JavaScript, but the only browser which supports it is Internet Explorer. If you want your application to be accessible to the widest possible audience, you should opt for JavaScript for your client-side script. In server-side script, on the other hand, it makes no difference which browser the user has. That's because the server executes the script and only sends standard HTML to the browser. Server-side script can be written in VBScript, JavaScript, Perl or several other languages.

Visual Interdev does not itself interpret the script, and it does not force you to use any particular language. However, it does provide editing and debugging support for VBScript and JavaScript. By default, Visual Interdev assumes that you will use VBScript for server-side scripting and JavaScript for client-side scripting. You can override these defaults from the project's property sheet, which you get at by right-clicking on the project in the Project Explorer. For this article, I will concentrate on server-side scripting. If you would like to see some examples of client-side script, see "Understanding JavaScript", PCSA Updates 139 and 140.

### ***Active Server Pages***

On Microsoft servers, server-side script is implemented through the medium of Active Server Pages (ASP). At its simplest, an ASP file is a standard HTML page whose extension has been changed to ASP. It can contain normal text and HTML tags, and is rendered by the browser just like any other HTML page. More typically, the ASP page will also contain embedded script. When the browser requests the page, a run-time engine on the server executes the script. Part of the job of the script will be to generate some sort of a result, such as a page of information for the user or perhaps a simple error message. The run-time engine embeds this information in the original page, and at the same time removes the script. It then sends the page to the browser, which renders it in the usual way.

*This article will be concluded in a future issue of PCSA.*

---

***"A project is a container for the Web pages, images, program code and other components that make up the application. It is strictly a developer concept."***

---

**PCSA**

*Copyright ITP, 2001*

## New Reviews from [Tech Support Alert](#)

### [Anti-Trojan Software Reviews](#)

A detailed review of six of the best anti trojan software programs. Two products were impressive with a clear gap between these and other contenders in their ability to detect and remove dangerous modern trojans.

### [Inkjet Printer Cartridge Suppliers](#)

Everyone gets inundated by hundreds of ads for inkjet printer cartridges, all claiming to be the cheapest or best. But which vendor do you believe? Our editors decided to put them to the test by anonymously buying printer cartridges and testing them in our office inkjet printers. Many suppliers disappointed but we came up with several web sites that offer good quality cheap inkjet cartridges with impressive customer service.

### [Windows Backup Software](#)

In this review we looked at 18 different backup software products for home or SOHO use. In the end we could only recommend six though only two were good enough to get our "Editor's Choice" award

### [The 46 Best Freeware Programs](#)

There are many free utilities that perform as well or better than expensive commercial products. Our Editor Ian Richards picks out his selection of the very best freeware programs and he comes up with some real gems.