# Site under construction

**Tim Anderson begins a new Hands On series with an introduction to dynamic web site development.**

If you want to build dynamic web sites, this new *Hands On* section is for you. A dynamic web site is one where the content is not fixed, but partly generated under program control. This opens up a huge range of possibilities, such as database access, sites that are easy to navigate, online purchasing, and more.

## Web development scenarios

Before going anywhere with web development, you have to be clear about how your solution will be deployed. The two key factors are first, which technologies you use; and second, where the website is hosted.

Fig 1 shows the most common technologies, divided into those that are executed on the client and those that run on the server. Fig 2 illustrates the matrix of hosting possibilities.

Client-side technology will always be problematic on the internet because you know nothing about the client except that it's some kind of browser. You can flag your site as for Internet Explorer 4.0 and higher only, or have browser detection with alternate sites for different browsers, but it isn't easy to get it right. Server-side technology is easier, in that you at least know what server you're using. If the server is in a box next to your desk, you can have everything your way.

On the other hand, for those with dial-up accounts and web space hosted by an ISP, there are problems. Your web space is on a server shared by other users, so ISPs are picky about letting you run scripts or executables that increase the server load and might cause problems or crashes. Generally, the more commercial (and expensive) accounts with ISPs have fewer restrictions.

## Run your own server

Even if your website is destined for web space hosted by an ISP, running a local web server makes a lot of sense. The easy option is the Personal Web Server or Internet Information Server, bundled with Windows. IIS, in particular, is an excellent product, except that your ISP probably doesn't use it. On the internet, nearly 60 percent of websites use Apache [Fig 3], almost all running on some variety of Unix.

If that includes your ISP, it makes a lot of sense to download Apache and install it on Linux, neither of which need cost you anything more than online time and a spare PC on your network. Learning Unix is handy for web developers, in any case. The web grew up on Unix, and chances are you'll be telnetting into a Unix server and will need some basic commands.

Having said that, many ISPs do offer NT and IIS as an option, so you can stay inside the cosy Windows world if you want. On a Windows intranet, the tables are turned and IIS is the obvious choice. This column will cover both.

## Server Side Includes

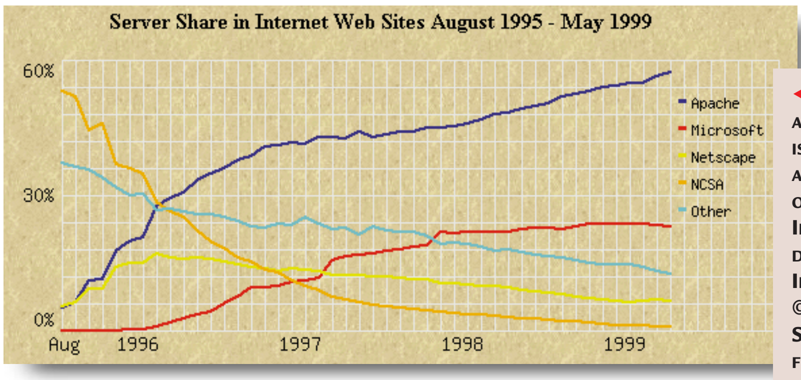
Server Side Includes (SSI) are a great place to get started with dynamic content, since they are simple and widely supported. This is in the realm of server-side technology though, so check with your ISP before assuming that anything will work. It's no use trying to run SSI tags by opening

**Fig 1: Web technology choice**

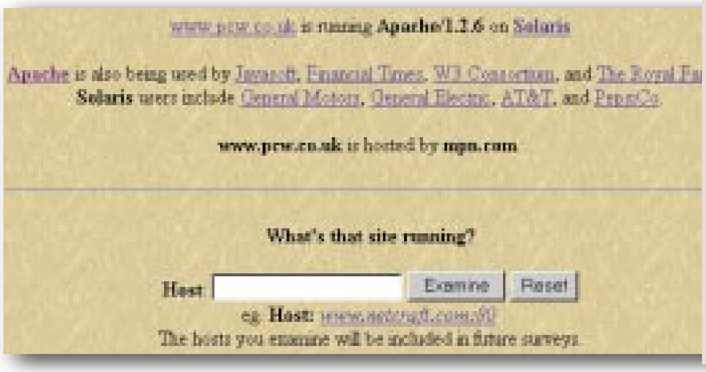
| Client side | Server side |
|---|---|
| Javascript<br>Java applets<br>DHTML<br>XML<br>Plug-ins<br>ActiveX | CGI<br>SSI<br>Perl<br>ISAPI/NSAPI<br>Java servlets<br>Active Server Pages<br>FrontPage Extensions |
| **Advantages**<br>No special server requirements<br>Good performance once downloaded<br>Needed for muiltimedia | **Advantages**<br>Browser-independent<br>Enables data-driven sites<br>Performance depends on server load and capabilities |
| **Disadvantages**<br>Browser-dependent<br>More potential security risk for clients<br>More to download means slower performance | **Disadvantages**<br>Server-dependent<br>ISP may not allow scripts and/or executables<br>Performance depends on server loads |

**Fig 2: The web development matrix**

| | Client side | Server side |
|---|---|---|
| Intranet | Easy | Easy |
| Leased line or own server at ISP | Difficult | Easy |
| ISP-hosted | Difficult | Difficult |



◀ **Fig 3** Web servers in use as at May 1999. The blue line is Apache, which has approaching 60 percent share on the internet and climbing. In second place but some distance away is Internet Information Server. © Netcraft 1999. See www.netcraft.com/survey for the latest figures.



www.pcw.co.uk is running Apache/1.2.6 on Solaris

Apache is also being used by Jaguarsoft, Financial Times, W3 Consortium, and The Royal Fa
Solaris users include General Motors, General Electric, AT&T, and PepsiCo

www.pcw.co.uk is hosted by mpn.com

What's that site running?

Host: [Examine] [Reset]
eg. Host: www.netcraft.com:80
The hosts you examine will be included in future surveys

◄ **Fig 4** What web server does your ISP use? One way to find out is to ask them. The other is to run this handy query from www.netcraft.com

web pages as files — they must be served by a web server.

SSI works by parsing placeholders in your HTML page. Each placeholder is replaced with content generated by the server. SSI placeholders are of the form:

```
<!-- #element
attribute="value" -->
```

and one of the most useful is the simple include command. which inserts a file into the HTML page before it is sent to the browser. For example, you might be developing a site with a navigation panel on the left.

You could use frames, but these have some drawbacks, so another approach is to have the navigation panel repeated on every page. That works well, except that if you add a page, every panel in the sequence needs to be amended.

The include command lets you keep the panel in a separate file, so that you only have one copy to maintain. For example, you might have a file called navpanel.html that looks like this:

```
<a href="/site/firstpage.shtml">First page</a>
<p><a href="/site/secondpage.shtml">Second page</a></p>
```

Then one of your content pages might look like this:

```
<html>
<body>
<table cellspacing="1" cellpadding="1" border="0">
<tr>
   <td width=15%>
   <!Ñ #include virtual="/site/include/navpanel.html" Ñ>
   </td>
  <td bgcolor="#000000"></td>
   <td valign="top"><H1>Welcome to the first page</H1></td>
</tr>
</table>
</body>
</html>
```

You need to save this with an .shtml extension. The different extension tells the web server to parse the file. On IIS, you can also use .stm. These extensions are configurable options. You could tell the web server to parse every .html file to look for includes, but this is a heavy performance hit.

Here are some other useful commands:

```
<!Ñ #echo
var="value" Ñ>
```

returns the value of a web server environment variable. Useful variables include DATE_LOCAL, LAST_MODIFIED, HTTP_USER_AGENT and others. Try the following to get a list of all the variables on an Apache server:

```
<!Ñ #printenv Ñ>
```

The following gets the size of a file, handy for downloading pages:

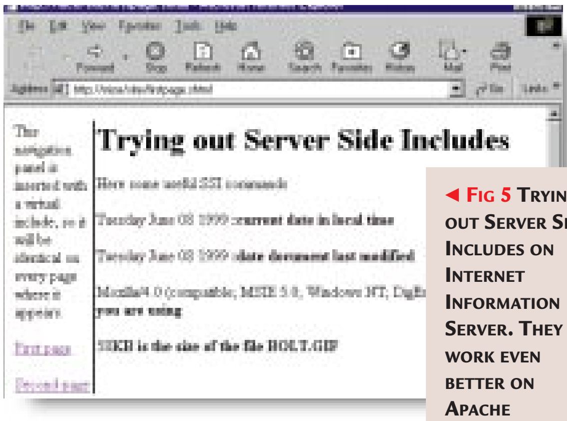
```
<!Ñ #fsize virtual="value" Ñ>
```

In this and other commands which refer to files, you can either use the virtual= attribute, which uses the relative URL, or the file= attribute, which is a relative file path. To configure the format for dates, times and file sizes, use the config element.

Two interesting SSI elements for a programmer are exec and if. The first executes a command or script. If you use the exec element, use the CGI attribute to specify a CGI script and the CMD attribute for a system command. The output from the command ends up in the HTML page.

A useful tip is that you can execute CGI scripts using a virtual include, so exec isn't often necessary.

The flow control elements are part of extended SSI (XSSI). The elements available are if, elif, else, and endif:



This navigation panel is inserted with a virtual include, so it will be identical on every page where it appears

First page

Second page

**Trying out Server Side Includes**

Here some useful SSI commands

Tuesday June 08 1999 **current date in local time**

Tuesday June 08 1999 **date document last modified**

Mozilla/4.0 (compatible; MSIE 5.0; Windows NT; DigE
**you are using**

**55KB is the size of the file HOLT.GIF**

◄ **Fig 5** Trying out Server Side Includes on Internet Information Server. They work even better on Apache

```
<!Ñ #if expr="condition" Ñ>
Show this text <p>
<!Ñ #else Ñ>
Show this other text <p>
<!Ñ #endif Ñ>
```

These work in Apache 1.2 and above, but not on IIS. The condition supports regular expressions as used by the Unix egrep search command, so keep that Unix reference handy for this one.

While IIS does support SSI, you can't use much SSI in an Active Server Page because an ASP has its own parser.

## PCW CONTACTS

*What do you want to know about web development? Email webdev@pcw.co.uk with your questions and comments, as these will determine the future direction of the column.*

- *You can discover what server your ISP is using by visiting www.netcraft.com and running a query there.*
- *Get Apache from www.apache.org, in source or binary form.*
- *There's an online reference to SSI at www.apache.org/docs/mod/mod_include.html*