If you sell "web-service" software there may be something important missing from your sales page!

By Bill Hely

NOTE: For the purpose of this article the term "web-service software" describes any type of software or script that is intended to be installed on a website server by the buyer of the application.

There is a very common mistake that many sellers of web-service software make repeatedly. That mistake is that they fail to specify exactly what the System Requirements are for the operation of the program or script they are selling. And if you don't think that's important, please grant me a few minutes to convince you, because it's probably costing you money.

Too many people, mostly marketers but including a lot of tech types who should know better, think only in terms of the web server Operating System (O/S) with which they are most familiar. More often than not that is the open source Apache web server software running on the Linux O/S, which is the cheapest platform on which to Host a website.

But these marketers are forgetting that there are also MILLIONS of websites hosted on Windows Servers running Microsoft's Internet Information Server (IIS). If you haven't considered this then you may be severely limiting your potential market.

In this article I want to look at the reasons for the oversight, why it's important, how to determine if your software package qualifies for a wider market share, and what to do about it.

You see, very often the person who developed a web-service application only wrote it for, and tested it on, a Linux server. And yet (and this is a point rarely properly understood by non-programmers) the exact same application may run on a Windows Server as well, thus considerably extending your potential customer base. However, you must be able to specify that feature with accuracy and confidence.

In the following paragraphs I hope to convince developers and marketers of web-service applications that placing an accurate summary of the System Requirements for their product on the sales page can mean extra money in their pockets and fewer support hassles. I also want to show marketers why their application may be more versatile than they think, and to knock a few fallacies on the head.

Now this may seem like a bit of a leap, but quite often the reason that marketers and shoppers alike think that an application is limited to certain web servers is due to a misunderstanding of an oft-repeated but much misunderstood term: CGI.

CGI stands for Common Gateway Interface, but don't worry about that, it's not important. Just call it "See Gee Eye".

Many people think CGI is a type of scripting language. It's not!

The important thing to understand is that **CGI is a STANDARD**, not a programming language or an Operating System or anything else. A "standard" is a set of rules. The CGI Standard defines and stipulates how external programs may be run on web servers. Stay with me … we're not going to get heavy! All will become much cleared as you read.

While the term “CGI” is mostly heard in relation to Linux web servers, there are a number of proprietary standards that extend the basic CGI Standard. One such is the “Internet Server Application Programming Interface” – better known as simply ISAPI. You can think of ISAPI as the CGI for Windows.

Again, don’t worry about all that heavyweight terminology! We're not going to delve into the technicalities. But hopefully you can see that using the term "CGI" as being something that is exclusive to Linux web servers is incorrect.

Now, to the part that causes so much confusion…

By definition (see above) there has to be a PROGAM involved. The program can be written in any programming code that can be legitimately executed on a server, including PHP, ASP, ASP.NET, Perl, TCL, Python, C, C++, Pascal, MIVA, Basic, VB, Java, FORTRAN, Unix Shell … and a whole bunch of others you’ve probably never heard of either.

For no good reason other than historical usage, CGI is often associated with the Perl programming language, and many people are under the impression that a CGI program is one written in Perl. Well, it might be. Or it might not. [Perl is in fact an acronym for *Practical Extraction and Report Language*].

It didn’t help that many Perl programmers began using .cgi as the filename extension for their Perl scripts. This was another factor that contributed to many people believing that CGI was a programming or script language in its own right. In fact the mandated extensions for a Perl file are .pl (a Perl script language source code file) or .pm (Perl Script Module). So instead of calling a Perl file something like filename.pl, many programmers were naming such a file filename.cgi. This was made possible because developers of web server software enabled their servers to automatically assume that the extension of .cgi denoted a Perl program, and the server would automatically open the .cgi program using its inbuilt or associated Perl interpreter.

Remember, there is no binding connection between Perl and CGI – the apparently close relationship just described is only the result of historical usage. So if you encounter a file with a .cgi extension, it is almost certainly a program written in the Perl programming language.

Anyway, the program, regardless of what type of code it is written in, is a "CGI program", but it is **NOT** "CGI". Remember, "CGI" is a **standard**; a "CGI program" is … well … a program. Two different things.

OK? Let’s move on to the next piece in the puzzle. It’ll all fall into place shortly.

Now a program - any program - can be completely self-contained or it can make "calls" to external resources, such as to subroutines buried in the Linux, UNIX or Windows Operating System itself.

So, if your web-service program is fully self-contained (no calls to external resources) there is an excellent chance that the said program will run on web servers that are hosted on **either** Linux/Unix **or** Windows.

The commonest misconception I encounter is that PHP and Perl CGI Programs will only run on Linux servers. That is not correct. Windows Servers can run PHP code if the Host administrator provides PHP support, which many do. Likewise, a Windows server administrator can, and often does, provide support for Perl.

Bottom line: a CGI Program written in any programming language **MAY** run on a Windows Web Server. But if that program makes external calls (such as to subroutines that are part of the Linux Operating System) then it will **NOT** run on a Windows Server.

The important point is this:

The reason the program won't run on a Windows Server has nothing to do with the programming language in which it was written, or whether it's called "CGI". It won't run because it ALSO uses code that is exclusive to the Linux/Unix Operating System and which is not available under the Windows Operating System.

So if no such restriction applies to your application – that is, if it doesn't rely on external code in a particular operating system - you may well have a bigger potential market than you thought, so TELL THEM ABOUT IT! Get the developer to write an accurate System Requirements summary and publish it on your sales page.

Note that an **accurate** System Requirements summary can only be written by someone who properly understands the terminology and the resources that have been used to create the application --- and ideally that is the developer/programmer himself. It is necessary to point this out because so many applications these days are being written by sub-contract programmers at the request of a marketer who has perceived a need for a product. Too often the marketer does not exactly and accurately blueprint her requirements beforehand, and rarely will the sub-contractor go out of his way to try and guess what unspecified extensibility might also be useful. Then the marketer writes a System Requirements summary based on limited or incorrect knowledge.

Sadly I have lost count of the times I have asked a marketer if their application will run on a Windows Server platform, only to receive either straight misinformation or meaningless gibberish in return.

The visitor reading the sales page for your product may be just as confused about the whole "CGI thing" as you are. He may very well see reference to CGI or PHP or MySQL or Perl, and immediately assume that your product is no good for his Windows-hosted website. Click. Gone! You just lost a sale.

If you are selling any type of web-service program, please, in your own best interests, consider a System Requirements summary a must-include on your sales page. It will never cost you a sale, but it might gain you one or several.

Some common misconceptions about "CGI"

"This program is written in CGI". A common claim, but rubbish never-the-less, and it doesn't tell your site visitor anything useful about System Requirements for your program. As we have already discussed, there is no such programming language as "CGI". To determine which web server platforms an application will run on, you first need to know the nature of the CGI program, as explained in the above article.

"This program will only run in the Linux cgi-bin". "cgi-bin" is the name of a special folder almost always found in the web space of Linux servers. Contrary to popular belief, there is nothing magical about it. It is a folder like any other, but with certain permissions set to allow programs and scripts stored in that folder to be executed – as opposed to being downloaded or displayed. Thus, on a Linux web server, the statement is often true, though it should be noted that some scripts do not have to be placed in the cgi-bin folder in order to be executed. It is pretty much up to the web

server administrator how and where he allows applications to run. Exactly the same functionality can be achieved with Windows folders as is provided by Linux's cgi-bin.

"You cannot set permissions on Windows web servers". Well, that depends. While YOU might not be able to, your Host administrator certainly can, and any reasonable Host will accede to reasonable requests. And even YOU may be able to set certain folder permissions yourself if the Host provides you with a Control Panel that allows such.

"You can't run a PHP script from the cgi-bin". Maybe, maybe not. Depends on how the administrator has set up the server. There is no embedded impediment to running PHP from the cgi-bin. However, there may be security reasons why a webmaster may prevent this.

"CGI can't use the MySQL data base". Well, if you read the main article above you already know that's a meaningless statement, because … again … CGI is a **standard**, not a programming language. So let's rephrase it…

"A CGI program can't use the MySQL data base". Still wrong! Some people think that MySQL is strictly "a PHP thing". For one thing, as has already been demonstrated, CGI programs can be written in PHP. Other programming languages can also interface very nicely with MySQL. For example, Perl using the Perl DBI (Database Interface) is a much used combination, but in fact most popular programming languages will work fine with MySQL, including C/C++, ASP, Python and most of those mentioned earlier.

"You can't run ASP web pages on Linux web servers". It's not something I'd be keen to do, but having dealt with the PHP-on-Windows question above, it's only fair to look at the converse situation. Yes, you can run ASP on Linux web servers. The solution requires 3rd party software, there will be certain restrictions and I suspect it could get a bit messy, but you certainly can do it.

In conclusion, allow me to remind you again: When advertising any application (be it executable program or script) that is designed to be installed on the buyer's web server, display an accurate and plainly worded System Requirements summary. It really is worth it.

Note 1: To keep things simple no real distinction is made between "program" and "script".

Note 2: This article is aimed at non-programmers, so it is necessarily a bit "fuzzy" in some respects. Trying to be too exact will only lead to confusion, thus defeating the purpose.

Note 3: Where the Operating System Linux is mentioned, the same generally applies to UNIX. Linux web servers are much more common than UNIX web servers these days.

-oOOo-

Bill Hely is a technologist, consultant and author living in Brisbane, Australia. For most of the last two decades his professional focus has been on advising and supporting small business operators in Information Technology and Office Productivity issues — and rescuing them when they didn't heed his advice the first time around. He is the author of several books on technology for the business operator, including the Bible of Internet and computer security *The Hacker's Nightmare*™. For more information on this must-read tutorial and reference visit: http://www.HackersNightmare.com