Apache CGI And Perl On Windows XP (Part I)

2008-1-21	Generate this document	HL

Background

This document is the narrative from the reference [1], it is strongly encouraged for you to read the original reference to get even more detailed information.

Apache on Windows XP

First, download Apache from http://httpd.apache.org/download.cgi . Then, follow the instruction for the installation. Choose default directory for the installation, e.g., C:\Program Files\Apache Group\ as shown in Figure 1.

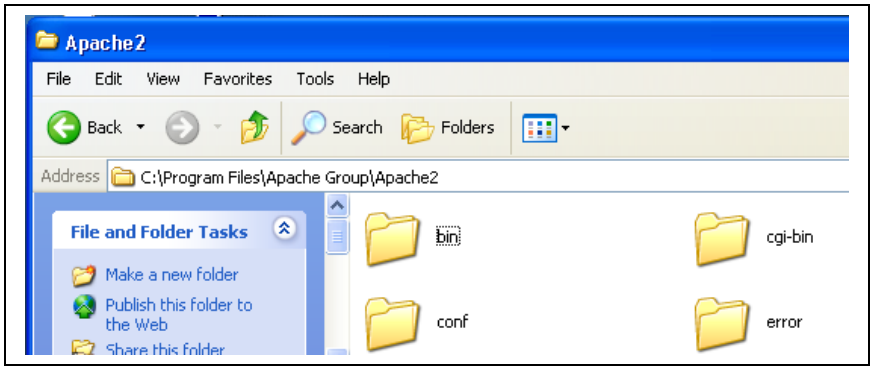

Figure 1. Apache folder locationte.

Now you can point your browser to http://localhost/ to view the web server start page. Once it worked, you can now test through LAN environment, the result is shown in Figure 2.

Figure 2. The Web sever start page.

Note in order to be able to visit your XP web server, you will need to make sure the fire wall sitting is adjusted to allow http services.

CGI Programming

We will take a look at the difference for Windows CGI programming.

3.1 Installing HTTP Server on Windows

See separate report on this topic.

3.2. Installing Perl on XP

Now download Perl from http://www.activestate.com/Products/ActivePerl/ site. Follow the instructions to install it, leave the setup as the default. This will install Perl, PPM (the Perl Package Manager) and programming examples to your hard drive in the location C:\Perl. Leave the "Enabled PPM3 to send profile info to ASPN" unchecked. Under Choose Setup Options, both "Add Perl to the PATH environment variable" and "Create Perl file extension association" should be checked.

3.3 Configure Apache and Enable CGI

First go to "My Documents" and make a new folder "My Website". This is where you're going to store your web pages and CGI programs.

Next modify the Apache configuration file to point to your pages and enable CGI as Start > All Programs > Apache HTTP Server > Configure Apache Server > Edit the Apache httpd.conf Configuration file. There are 3 modifications to be made as in Table 1.

1. Uncomment LoadModule userdir_module modules/mod_userdir.so	LoadModule userdir_module modules/mod_userdir.so	
2. Add a section of UserDir	The whole section is given in the following	
3. Modify the “Options” line in the above section	Options MultiViews Indexes SymLinksIfOwnerMatch Includes ExecCGI	
4. Modify DirectoryIndex line	add `index.cgi` to the end of that line	
5. Modify AddHandler	`AddHandler cgi-script .cgi .pl` `AddHandler server-parsed .html`	

Table 1. Modification for Apache Configurations.

If not using Apache 2.2, then find UserDir section with the first line as

UserDir "My Documents/My Website"

Uncomment this entire section below（removing # sign):

```
#<Directory "C:/Documents and Settings/*/My Documents/My Website">
#  AllowOverride FileInfo AuthConfig Limit
#  Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
#  <Limit GET POST OPTIONS PROPFIND>
#    Order allow,deny
#    Allow from all
#  </Limit>
#  <LimitExcept GET POST OPTIONS PROPFIND>
#    Order deny,allow
#    Deny from all
#  </LimitExcept>
```

#</Directory>

Now change the `Options` line to this:

Options MultiViews Indexes SymLinksIfOwnerMatch Includes ExecCGI

Where Indexes enables server-side includes, and ExecCGI, enables CGI programs in this directory. Note if you are using Apache 2.2 like me, then you will have to generate the entire above section. Once that's done, then scroll down to DirectoryIndex line, add `index.cgi` to the end of the line:

DirectoryIndex index.html index.html.var index.cgi

Now scroll down several pages to AddHandler section. Uncomment the CGI line:

AddHandler cgi-script .cgi

This allows any .cgi file to be processed. If you want to have files with a .pl as CGI programs, add the .pl extension on that same line:

AddHandler cgi-script .cgi .pl

After that add this line:

AddHandler server-parsed .html

Which enables .html files to be searched for server-side include tags. Now save the configuration file, and restart Apache. To restart the Apache, you will have to go start > All Programs > Apache HTTP Server 2.2 > Control Apache Server > Restart. Now your server is up.

3.4 Writing Perl Programs

Write your first perl program as in Figure 3 and its execution in Figure 4. Note you will have to place your first perl program, hello.pl in the cgi-bin directory, in my case, it is C:\Program Files\Apache Software Foundation\Apache2.2\cgi-bin. Then point your browser to: http://192.168.1.108/cgi-bin/hello.pl .

```
#!/perl/bin/perl -wT
    print "Content-type: text/html\n\n";
    print"<h2>This is my first perl</h2>\n";
    print "<h2>Harry: Hello, World!</h2>\n";
```

Figure 3. The first perl program on Windows XP.

Note: `#!`, indicates that this is a script. The next part, `/usr/bin/perl`, is the location (or *path*) of the Perl interpreter.

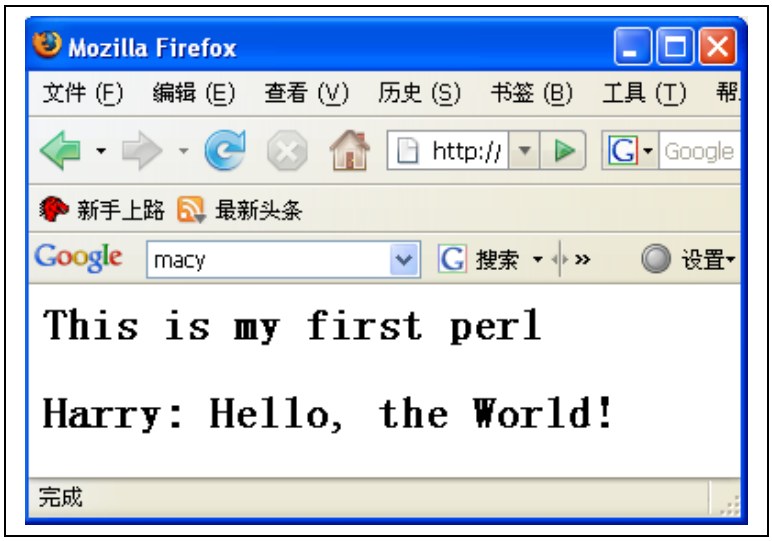

Figure 4. Perl is running.

Note: The difference of perl for Windows is listed in Table 2 below.

	Linux	Windows
The first line in Perl e.g., “shebang” line	`#!/usr/bin/perl`	`#!/perl/bin/perl`

Table 2. Difference of the first Perl line for XP and Linux.

Note: 1. The actual location of Perl may be different from system to system (e.g. `/bin/perl`, `/usr/local/bin/perl`, etc.) For ActivePerl in Windows, this line should be changed to: `#!/perl/bin/perl` If you're programming locally and uploading to a remote ISP, you'll have to change this line each time.... unless your ISP was thoughtful enough to add a symlink to Perl in `/perl/bin/perl`.

Debugging via Log Files

Debugging by examining webserver log files. The server logs are located at /usr/local/etc/httpd/logs/error_log, or /var/log/httpd/error_log. In the Unix shell, you can use the tail command to view the end of the log file: tail /var/log/apache/error_log

3.5 Environment Variables

The parameters in the browser (html files) and http server are passed to CGI by environment variables. The Environment variables are a series of hidden values that the web server sends to CGI program. Your can write a program to parse them. Environment variables are stored in a hash named `%ENV`, they are listed in Table 3 below. You can print the environment variables as follows:

print "Caller = $ENV{HTTP_REFERER}\n";

	Key	Value
1	DOCUMENT_ROOT	The root directory of your server
2	HTTP_COOKIE	The visitor's cookie, if one is set
3	HTTP_HOST	The hostname of the page being attempted
4	HTTP_REFERER	The URL of the page that called your program
5	HTTP_USER_AGENT	The browser type of the visitor
6	HTTPS	"on" if the program is being called through a secure serve
7	PATH	The system path your server is running under
8	QUERY_STRING	The query string (see GET, below)
9	REMOTE_ADDR	REMOTE_ADDR The IP address of the visitor
10	REMOTE_HOST	The hostname of the visitor (if your server has reverse-name-lookups on; otherwise this is the IP address again)
11	REMOTE_PORT	The port the visitor is connected to on the web server
12	REMOTE_USER	The visitor's username (for .htaccess-protected pages)
13	REQUEST_METHOD	GET or POST
14	REQUEST_URI	The interpreted pathname of the requested document or CGI (relative to the document root)
15	SCRIPT_FILENAME	The full pathname of the current CGI
16	SCRIPT_NAME	The interpreted pathname of the current CGI (relative to the document root)
17	SERVER_ADMIN	The email address for your server's webmaster
18	SERVER_NAME	Your server's fully qualified domain name (e.g. www.cgi101.com)
19	SERVER_PORT	The port number your server is listening on
20	SERVER_SOFTWARE	The server software you’re using (e.g. Apache 1.3)

Table 3. Environment variables.
The following prl program prints environment variables, see Figure 5 and 6.

```
#!c:/Perl/bin/perl.exe
##
##  printenv -- demo CGI program which just prints its environment
##

print "Content-type: text/plain; charset=iso-8859-1\n\n";
foreach $var (sort(keys(%ENV))) {
    $val = $ENV{$var};
    $val =~ s|\n|\\n|g;
    $val =~ s|"|\\"|g;
    print "${var}=\"${val}\"\n";
}
```

Figure 5. Perl to print environment variables.

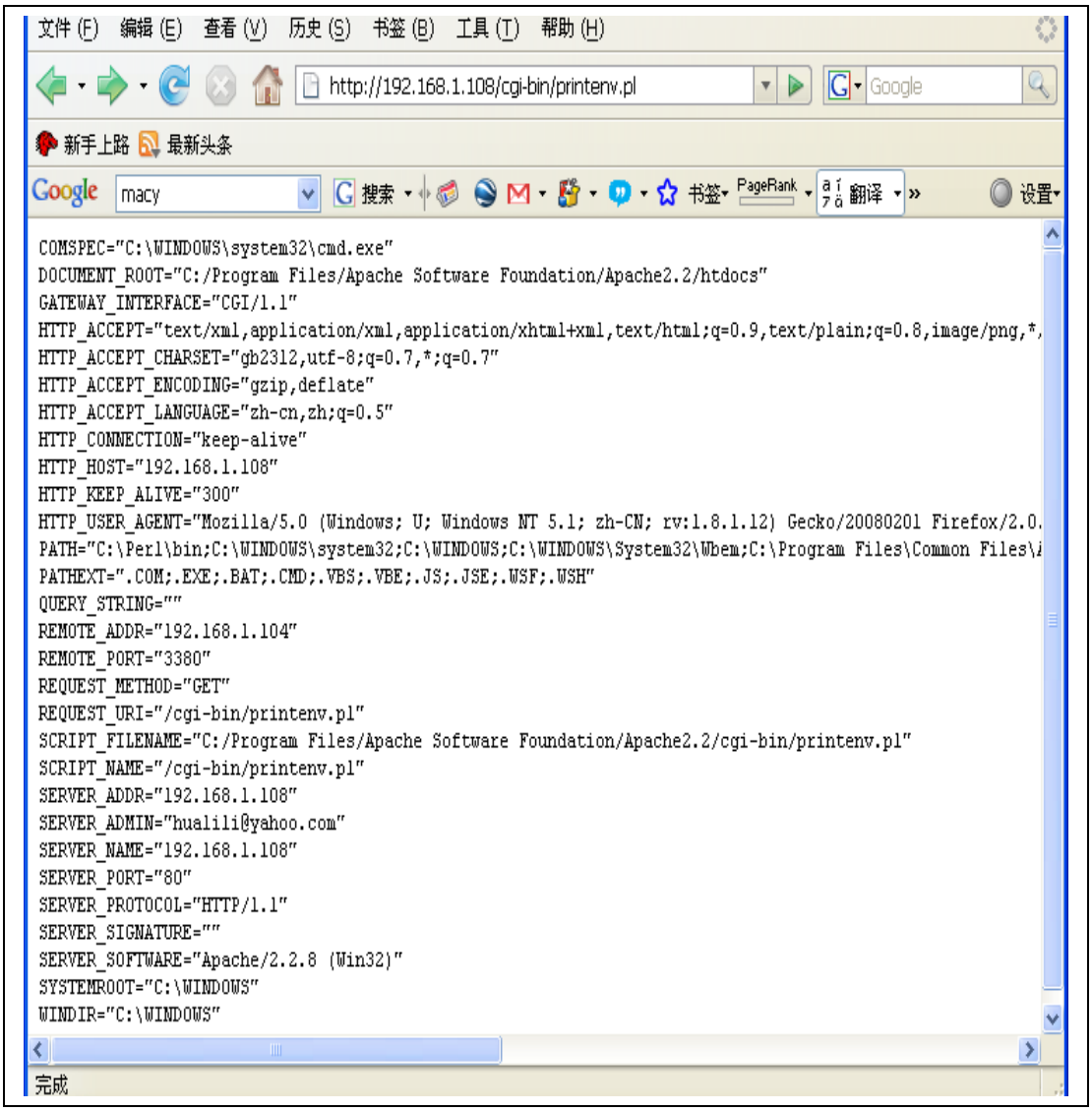

Figure 6. Print environment variables.

3.6 Sending Data from Web Form to CGI

There are two ways to send data from a web form to a CGI program: GET and POST as in Table 3.

Parameter Passing	Description	
GET	(1) The input values from the form are sent as part of the URL ; (2) saved in the QUERY_STRING environment variable	
POST	With the POST method, data is sent as an input stream to the program.;	

Table 3. Parameter passing.

Now a simple program to get variables from CGI to the web server or the browser is listed in Figure 7. Note you will have to make sure that you have "printenv.pl" in the /cgi-bin directory. The execution result is given in Figure 8.

```
<html><head><title>Test Form</title></head>
<body>

<form action="printenv.pl" method="GET">
Enter some text here:
<input type="text" name="sample_text" size=30>
<input type="submit"><p>
</form>

</body></html>
```

Figure 7. HTML code for parameter passing from CGI.

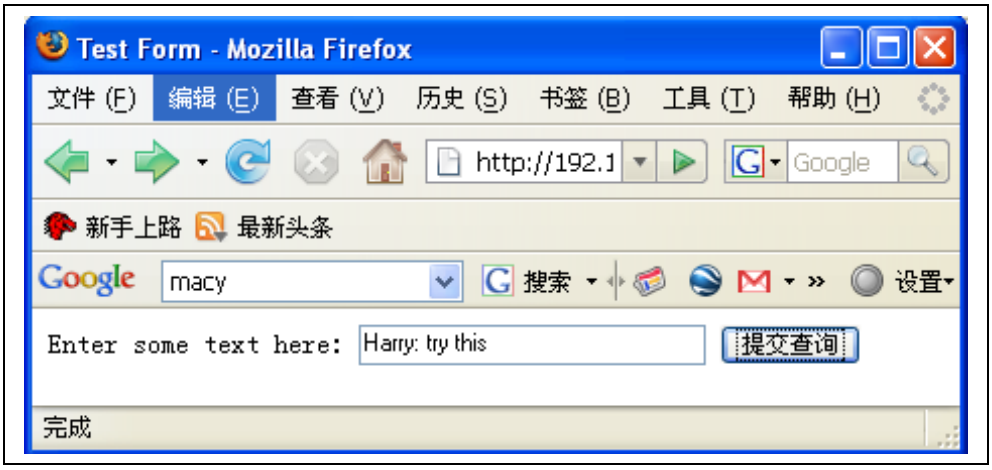

Figure 8. The HTML page to get parameters from Perl via CGI.

Reference

1. http://www.cgi101.com/learn/connect/winxp.html
2. http://www.cgi101.com/book/ch1/text.html

(End)