

第一章：基础的基础

1.1 为什么使用CGI？

我没有把什么是CGI放在基础篇的第一段，是因为实在很难说明白到底什么是CGI。而如果你先知道CGI有什么作用，将会很好的理解CGI是什么这个概念。CGI可以为我们提供许多HTML无法做到的功能。比如 a. 一个记计数器 b. 顾客信息表格的提交以及统计 c. 搜索 d. WEB数据库 用Html是没有办法记住客户的任何信息的，就算用户愿意让你知道。用Html也是无法把信息记录到某一个特定文件里的。要把客户段的信息记录在服务器的硬盘上，就要用到CGI。这是CGI最重要的作用，它补充了Html的不足。是的，仅仅是补充，不是替代。

1.2 CGI是什么？

好了，现在我们来说到底什么是CGI。Common Gate Intergace听起来让人有些专业，我们就管它叫CGI好了。在物理上，CGI是一段程序，它运行在Server上，提供同客户段Html页面的接口。这样说大概还不好理解。那么我们看一个实际例子：现在的个人主页上大部分都有一个留言本。留言本的工作是这样的：先由用户在客户段输入一些信息，如名字之类的东西。接着用户按一下“留言”（到目前为止工作都在客户端），浏览器把这些信息传送到服务器的CGI目录下特定的cgi程序中，于是cgi程序在服务器上按照预定的方法进行处理。在本例中就是把用户提交的信息存入指定的文件中。然后cgi程序给客户端发送一个信息，表示请求的任务已经结束。此时用户在浏览器里将看到“留言结束”的字样。整个过程结束。

1.3 选择你熟悉的编程语言

既然CGI是一种程序，自然需要用编程语言来写。你可以用任何一种你熟悉的高级语言，C, C++, C shell和VB。值得特别指出的，有一种叫Perl的语言。其前身是属于Unix专用的高级语言，其具有强大的字符串处理能力而成为现在写CGI，特别是表单类程序的首选。最近它已经有了Window95，和winnt版本。你可以在搜索程序里找到在那里下载它。VB是Ms的杀手锏，从目前的情况看，微软公司正试图使VB无所不能。自然也包括在Internet 请各位注意，VB开发的程序只能在windows平台上被执行，所以它有一定局限。C Shell，经典的语言。可惜能做的事情不多，而且必须在Unix平台下。C, C++，真正的无所不能。可是在写CGI的时候显得非常难以掌握。特别是缺乏可以灵活使用的字符串处理函数。对程序员的要求也比较高，维护复杂。最后要提醒各位，因为CGI是Server和Clinet的接口，所以对于不同的Server, CGI程序的移植是一个很复杂的问题。一般对于不同的Server, 决没有两个可以互相通用的CGI。实际上 这就是CGI程序最复杂

的地方。

1.4 安全

我想各位敏感的朋友又要问我关于安全性能的问题了。实际上CGI是比较安全的，至少比那些没有数字签名的ActiveX控件要安全的多。除非你有意在程序里加入了破坏Server的命令，否则一般不会有有什么严重的后果。而个人网站不向大众开放CGI目录，则因为怕各位学习不精，无端增加服务器的负担，所以一般不提供。

小结：

本章讲述了CGI基本概念，也说明了各种编程语言的优缺点，同时解释了为什么个人

第二章 Html的表单基础知识

组成Cgi程序的是两部分，一部分是html页面，就是用户看到的东西。另一部分则是运行在务器上的程序。一般来说，我们先些html页面，再些程序。所以我们的学习页是这样。

先看看表单的写法，再去学习CGI程序。于html的表单，有比较复杂和详细的规定。但由于它们是属于html标准里的东西，所以我不打算详细讲解。最好是你自己找本书看看。

Html表单

`<Form></Form>`

标记把输入域组合起来,并且说明了表单提交的方式和地点. Form标签有许多属性,比如NAME等等. 具体的属性,将在使用的时候说明. 在FORM里还有许多不同的标签,正是它们组成了表单的各种成分.

`<INPUT [TYPE=TEXT] NAME="TEXT-ID" [SIZE=NN] [MAXLENGTH=NN] [VALUE=' DEFAULT TEXT']>`该语句在HTML里产生一个文本标签, []里表示该属性是可选的.

`<INPUT TYPE=SUBMIT [NAME=' BUTTON-ID'][VALUE="BUTTON LABEL TEXT"]>`该语句产生一个提交按键,用于表单的提交. 它提供一个发送表单的按钮,信息发向<FORM>标记里ACTION属性里给出的URL. 根据表单方法(METHOD)指定的HTTP请求类型提交信息. 在以后我将详细讲解.

`<INPUT TYPE=RESET[VALUE=' BUTTON LABEL TEXT']>`具有TYPE=RESET属性的INPUT将产生一个按钮,它重置表单各个域的内容到初始状态.

`<INPUT TYPE=RADIO NAME="RADIO-SET -ID" vALUE="CHOICE-ID" [CHECKED]>`表单可以用此标记提示用户从几个选择项里作出选择. 么个选择项目都是一个单选按钮它们具有相同

的NAME值. 在任何时候只要有一个被选中. 此标记有一个VALUE属性, 如果它被选中, 则发送表单时将以"RADIO-SET-ID=CHOICE-ID"的形式发送.

<INPUT TYPE=CHECKBOX NAME='BOX-SET-ID' VALUE='CHOICE-ID' [CHECKED]> 此标签将产生一个' OFF' 或' ON' 的开关. 它同RADIO相似, 但是可以多选. 当提交表单的时候 如果一个复选框是选中的, 那么它的VALUE属性作为BOX-SET-ID表单数据的值被提交. 几个 拥有同样NAME属性的复选框为一组. 它们用', ' 分开的列表.

CGI简明教程 第三章

传送方法:

所谓方法是指调用CGI程序的途径. 事实上, 要执行程序时, 你用一种方法向服务器提出请求, 此请求定义了程序如何接受数据. 下面介绍常用的两种方法: get和post.

1. GET

当使用这种方法时, CGI程序从环境变量QUERY_STRING获取数据. QUERY_STRING被称为环境变量, 就是这种环境变量把客户端的数据传给服务器. 为了解释和执行程序, CGI必须要分析(处理)此字符串. 当你想从服务器获得数据并且不改变服务器上的数据时, 你应该选用GET. 但如果你的字符串长度超过了一定长度, 那么还是选用POST方法.

2. POST

使用POST方法时, WEB服务器通过stdin(标准输入), 向CGI程序传送数据. 服务器在数据的最后没有使用EOF字符标记, 因此程序为了正确的读取stdin, 必须使用CONTENT_LENGTH. 当你发送的数据将改变Web服务器端的数据或者你想给CGI程序传送的数据超过了1024字节, 这是url的极限长度, 你应该使用post方法.

3. 实现方法:

上章讲过html中的<Form>标签, 对于传送数据方法的定义也在这个标签里定义. 如下:

```
<form name="guyi" form" action="http://www.yourname.com/cgi/your.cgi"
method=GET>
```

定义了一个以get方法传送数据的form, 它的服务端程序是
http://www.yourname.com/cgi/your.cgi. 同样的如下的定义将使用post方法.

CGI简明教程 第四章

基本调试环境的建立:

我想不是每个阅读教程的朋友有机会接触到服务器之类的东西的。所以在接下去的阅读过程中没有调试环境的朋友将会非常吃力。为了解决这个问题，我特别介绍一种在win95或NT上建立WEB SITE的方法，以便各位能很好的阅读程序。不过本方法内使用的软件都比较大，如果你不是一个真正的网络发烧友，或许你会放弃这个教程。这个我也没有办法，要知道喜欢一样东西总是要付出代价的。

1. win95&winnt的Server软件：

首先我要各位弄清楚一个概念，所谓Sever只是指在一台电脑上安装了一个特别的软件。该软件可以提供internet web site服务。这台电脑不一定是有什么特别好的机器，只要它能够很好的运行Server软件就可以了。比如我们的个人电脑就可以承担这个任务。在Wins95下运行的server软件不是很多，而且是属于“业余”的。就是说离真正的商业使用有很大的距离。

这主要是windows95平台的局限，比如很差的安全性，不稳定的内存分配系统，混乱的文件系统以及最差的进程管理。不过这些不影响我们使用CGI程序，特别是一个人调试自己的程序，而不是放在internet给数万个人访问。其代表是MS在win OSR提供的pws (person web server)。

你可以通过控制面版-->网络-->服务-->Ms-->个人WEB服务器安装它。不过我希望你们最好还是不要使用这个东西，因为它对我们下面的课程没有什么帮助。我曾试验过用它来和我的perl程序相连，但很显然不支持perl。虽然有消息说修改注册表可以解决这个问题，但我到现在还是没有找到方法。相对而言，Windows NT因为其定位的不同而可以提供较理想的操作平台。而且NT自带有IIS(Internet Information Server)。不过是2.0版的，如果你有IIS3.0就更好了。到本文写成的时候，IIS已经有了4.0版本。当然我相信不是所有人会去下载这东西，因为有104M。

上面讲了这么多，实际上我要给各位推荐的是WebSite for win95&NT。就是说两个操作系统都可以使用它。现在的版本是2.1，有17M。显然是一个很大的软件，不过正如我说的，喜欢一样东西是有代价的。你可以在<http://www.windows95.com>里找到它。WebSite支持winCGI，也支持perl写的CGI程序。显然这是一个好东西，可以满足我们的要求。虽然大了些，不过要是你能找到website 1.1的话也不错，它大概有4M左右。不过我在网上从没发现有下载的地方。

2. Perl for win32

Perl, 让人激动的名字。我想再也找不到一个编程语言可以如此出色的完成基本CGI程序的开发了。我感谢那些为perl移植做出贡献的朋友们，是他们让我们可以在win平台上使用如此如此强大的Unix工具。你可以在perl的主页上找到perl for win32版本，当前版本是5.02。你也可在 <http://download.ihw.com.cn>里找到perl for win32版本。大概是2.5M。

3. VB

VB是我们写WinCGI程序最好的选择。我推荐各位使用vb4。不是因为vb5不好，只是在Internet上支持它的文档和免费代码太少了。

4. C&C++

这个教程里几乎没有用C&C++写的代码，因为它更适合高级程序员而不是我们这些业余选手。不过你要是一定使用它我也没有意见。

5. 安装。

好了，我们开始建立基本环境。就是说我们要在自己的机器上建立一个WEB SITE!首先安装Website。按照它标准的安装界面，你可以轻松的完成这个工作。缺省情况下它将被安装到c:\website下。我提醒各位注意这几个目录：

cgi-shl:这个是基本的cgi目录，也就是各位朝思暮想的地方！

cgi-win:这个cgi目录里放的是WinCgi程序。其他的各位自己研究吧。

好了，在装完Website后请确认它已经运行。在状态栏地右下角会有一个齿轮地图标它表示website正在运行。接下去请打开你地浏览器，在url输入你机器的ip地址，如果没有ip，请输入:127.0.0.1。注意，做这些动作的时候，最好请你使用netscape，而且你不必连上网络！如果一切顺利的话，你将看到website的标记，那证明你已经成功了！

接下去我们安装perl。下载后的perl是一个自解开文件，它会问你装到哪个目录。缺省是c:\perl，接着它展开并产生几个自目录。在安装结束的时候，它会修改

autoexec.bat，把c:\perl\bin加如 path，请一定要这么做，道理我会在下面解释。最后

请随便写一个以.cgi或.pl结尾的文本文件。直接双击它，并为它加上关联。.pl

或.cgi的关联文件是c:\perl\bin\perl.exe。这步很重要，不然你无法完成接下去的学习。那么为什么我们要加路径呢？因为perl是一种文本解释型语言，就是说它没有编译的过程，只有在运行的时候解释。解释一句运行一句。所以我们必须让系统知道perl.exe在哪里。

6. 测试

最后我们进行一个测试。请把下面的两个程序下载并运行它们：

1.html文件：这是测试程序的客户端内容，请以test.htm保存它。并把它放在c:\website\hotdocs\ 下，那是你的website缺省的根目录。

test.htm:

```
<html>
<title>你的第一个perl测试程序</title>
<body>
<h1>你的第一个perl测试程序</h1><br>
<h2>win 平台的环境变量测试</h2><br>
<form method="get" action="http://127.0.0.1/cgi-shl/env.pl">
<input type=submit value="查看CGI环境变量">
</form>
</body>
</html>
```

2.Perl文件：请使用任意一个文本编辑器，输入后以env.pl保存到c:\website\cgi-shl\下。

```
#! perl
print"Content-type:text/html\n\n";
print"<title>Windows环境变量</title>";
print"<body>恭喜你的website已经成功<br>";
foreach $key(sort keys %ENV) {
```

```
print"$key=$ENV{$key}<br>"; }  
PRINT"</body>";
```

在完成上面的工作后，请打开浏览器，输入<http://127.0.0.1/test.htm>。当然如果你是以tcp/ip方式连在局域网上，请把127.0.0.1改成你的ip地址（还有test.htm里的。）然后按下那个键，当你的浏览器去向另一个页面的时候，今天的任务就算是完成了。

CGI简明教程 第五章

服务器与CGI程序交换信息的协作方式是通过环境变量实现的。无论什么请求，CGI程序总能在特定位置找到某些信息。无论环境变量怎样定义，总有一些变量有着特定含义。环境变量是保存用户信息的内存区。例如，所有的机器都有一个PATH环境变量，当在当前目录找不到文件时就要查找PATH变量。当服务器收到一个请求后，它首先要收集它能得到的所有相关信息，并把它放入内存。那么，服务器要收集什么信息呢？

1. 关于服务器自身的详细信息
2. 关于用户的信息信息
3. 关于用户请求的信息

服务器不知道CGI程序到底需要那些信息，所以它把这些信息一起收集，那么重要的东西就会遗漏了。为了向你展示服务器收集那些环境变量，在上一章的教程里我已经为各位写了一个程序，该程序将几乎所有的信息都反映在了浏览器里。所以在进一步学习变量的时候，请首先把程序调试好。环境变量

1. 与服务器相关的环境变量

| | |
|-------------------|-------------|
| GATEWAY_INTERFACE | 服务器遵守的CGI版本 |
| SERVER_NAME | 服务器的IP或名字 |
| SERVER_PORT | 主机的端口号 |
| SERVER_SOFTWARE | 服务器软件的名字 |

CGI简明教程 第六章

对于本周的程序——选票程序，我不想多说什么。我只是想在本程序内教给大家如何

使用perl语言或C语言如何处理字符串的实例。让我们先看看代码：

页面文件——pote.html

```
<HTML>  
<TITLE>投票程序</TITLE>  
<h1>投票程序</h1>
```

```

<FORM METHOD="post" ACTION="/cgi-bin/vote.pl">
一. 你喜欢谁?<br>
<input type="radio" name="idol" value="A" checked>张德陪<br>
<input type="radio" name="idol" value="B">阿加西<br>
<input type="radio" name="idol" value="C">桑普拉斯<br>
<input type="radio" name="idol" value="D">贝克<br>
<input type="radio" name="idol" value="D">顾懿<br>
<input type="submit" value="执行">
<input type="reset" value="取消">
</form>
查询<a href="/cgi-bin/vote.pl?command=view">选举投票结果</a>
</html>

```

CGI程序: --vote.pl

```

#!/perl
print"Content-type:text/html\n\n";
print"<titel>投票系统</title>";

if($ENV{'REQUEST_METHOD'} eq "POST") {
read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});
} elsif ($ENV{'REQUEST_METHOD'} eq "GET") {
$buffer=$ENV{'QUERY_STIRNG'};
}

@pairs=split(/&/, $buffer);
foreach $pair(@pairs) {
($name, $value)=split(/=/, $pair);
$value=~tr/+///;
$value=~s/%([a-f A-F 0-9][a-f A-f 0-9])/pack("C", hex($1))/eg;
$FORM{$name}=$value;}

$filename="/vote.dat";
%NAME=("A", "张德陪", "B", "阿加西", "C", "桑普拉斯", "D", "贝克", "E", "顾懿");

if($ENV{'REQUEST_METHOD'} eq "POST") {
print"Content-type:text/html\n\n";
print"<titel>投票系统</title>";
print"<h1>投票系统的例子</h1>";
open(FILE, "<$filename") || die"不能打开文件, 请和管理员联系\n";

```

```

for($i=0;$i<2;$i++){
$file[$i]=<FILE>;
$file[$i]=~s/\n$//;
}
close(FILE);

@item=split(/:/,$file[0]);
@vote=split(/:/,$file[1]);

for($i=0;$i<@item;$i++){
if($FORM{'idol'}eq$item[$i]){
$vote[$i]++;
last;
}
}
open(FILE,">filename")||die"Can't Open the file";
$item=join(":",@item);
$vote=join(":",@vote);
print FILE "$item\n";
print FILE "$vote\n";

close(FILE);

print"<h2>您是投票给$NAME{$FORM{'idol'}},感谢您的选票!<h2>";
print"查询<a href=\"/cgi-bin/vote.pl?command=view\">投票结果系统</a>";

}

if($FORM{'command'}eq"view"){
print "HTTP/1.0 200\n";
print "Content-type:text/html\n\n";
print"<title>投票结果</title>";
print"<h1>投票结果</h1>";
open(FILE,"$filename")||die"文件打开错误";

for($i=0;$i<2;$i++){
$file[$i]=<FILE>;
$file[$i]=~s/\n$//;
}
close(FILE);

```



```

@item=split(/:/,$file[0]);
@vote=split(/:/,$file[1]);

print"<table border=1>";

for($i=0;$i<@item;$i++){
print"<tr><td>姓名</td><td>$NAME{$item[$i]}</td><td>票数</td>,td>$vote[$i]
</td><tr>";

}
print "</table>";
}

```

这个程序是要各位学习Perl的分解字符串的功能。在Perl中，字符串操作是非常简单的。我对几句重要语句做一个分析：

```

if($ENV{'REQUEST_METHOD'}eq"POST"){
read(STDIN,$buffer,$ENV{'CONTENT_LENGTH'});
}elseif($ENV{'REQUEST_METHOD'}eq"GET"){
$buffer=$ENV{'QUERY_STRING'};
}

```

这是一个非常常见的功能块，几乎所有的CGI程序都会用到它。它判断页面使用何种方式提交变量。如果是post，就从STDIN里把变量读出，存到变量buffer里。注意在perl里，变量是用\$开头的。而\$ENV{'CONTENT_LENGTH'}则是读出该变量的长度，请注意CONTENT_LENGTH是一个环境变量。第二个if则处理get情况，在get情况下，页面提交的信息是存放在环境变量QUERY_STRING中的。所以\$buffer就是页面提交的信息。环境变量REQUEST_METHOD表示方式，它的值是一个字符串，前面加上\$ENV则表示读出该变量的值。

```

@pairs=split(/&/,$buffer);
foreach $pair(@pairs){
($name,$value)=split(/=/,$pair);
$value=~tr/+///;
$value=~s/%([a-f A-F 0-9][a-f A-f 0-9])/pack("C",hex($1))/eg;
$FORM{$name}=$value;}

```

以上功能块是一个分解过程。页面信息的提交往往是以“名称=值”的形式，比如本例子中就是以“idol=顾懿”这样的形式提交的，所以我们要去掉字符串中的“=”等等信息，但同时要保留“idol”和“顾懿”之间的对应关系\$FORM{\$name}=\$value做到这一点，这是

一个关联数组。具体的语法，请各位自行查书。接下来的就是文件的操作，但那已经不是本课的重点。所以就不多说了。