

# 第三部分

---

# 将Perl用于CGI

第17学时 CGI 概述

第18学时 基本窗体

第19学时 复杂窗体

第20学时 对HTTP 和CGI进行操作

第21学时 cookie

第22学时 使用CGI程序发送电子邮件

第23学时 服务器推送和访问次数计数器

第24学时 建立交互式Web站点

## 第17学时 CGI概述

毫无疑问，人们普遍认为 Internet的爆炸性流行主要是因为有了 World Wide Web。自从1993年第一个图形 Web浏览器的问世以来，Internet便以惊人的速度迅速发展，1993年前后Internet上的主机数量每20个月翻一番，而目前则每12个月翻一番。专用网络即 Intranet的增长速度甚至更快。

1993年以来，Web的内容已经变得越来越杂，Web用户希望每个Web页不只是能够显示静态（不变的）Web内容。成功的 Web站点必须显示动态 Web页，也就是能够提供最新信息的Web页。要使复杂的Web页能够跟上内容的迅速变化，这几乎是不可能的，因此出现了公用网关接口（CGI）。



为了学习后面7个学时的内容，你必须具备关于超文本标记语言（HTML）的某些知识。如果你对 HTML不熟悉，不必担心，它学习起来并不困难，也不需要通过本书来学习更多的这方面的内容。

HTML是一种标记语言，常用于创建 Web页。HTML由纯文本组成，其格式化代码嵌入文本之中，以指明 Web浏览器应该如何显示文本。例如，HTML is *<I>not</I>* hard to learn这句话是个普通文本，而 *<I></I>* 这些标记则不属于普通文本。它们称为标记，用于描述应该使用何种格式来显示文本。在上面这个例子中，Web浏览器应该用斜体字来显示单词 not。（请记住，并非所有浏览器都具有图形显示功能。）

关于HTML的详细说明，不属于本书要讲解的范围。介绍并不困难，但是有大量的资料需要加以说明。HTML的技术规范由World Wide Web集团（W3C）负责维护，该机构的网址是 <http://www.w3c.org>，可以通过该网址找到许多很好的教材。《HTML 24学时教程》是介绍HTML的一本好书。

在本学时中，你将要学习：

- Web是如何运行的。
- 在编写CGI之前你应该具备什么知识。
- 如何编写你的第一个CGI程序。

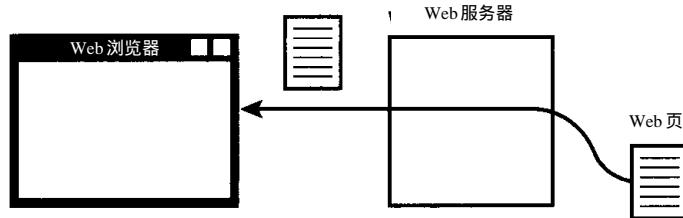
### 17.1 浏览Web

也许你已经知道，Web是指试图进行数据交换的两个不同系统之间进行的交互操作。试图抓取Web页的系统称为客户机系统。客户机系统通常运行一个称为 Web浏览器的程序，比如Netscape、Internet Explorer和Opera等，这是你习惯于日常使用 Web的应用范围。Web浏览器配有浏览按钮和书签，用于在屏幕上绘制 Web页。

在Web的另一端是称为 Web服务器的系统。该系统负责接收客户机查看 Web页的请求，从本地磁盘上检索 Web页，并将 Web页发送给客户机系统，即你的 Web浏览器。图17-1显示了这

种交互操作的情况。

图17-1 Web浏览器检索一个Web页



### 17.1.1 检索一个静态Web页

当客户机需要检索一个Web页时，它要查看统一资源定位器（URL），以确定使用什么协议、服务器，以及在该服务器上提出的是什么请求。典型的URL类似下面的形式：

`http://www.google.com:80/more.html`

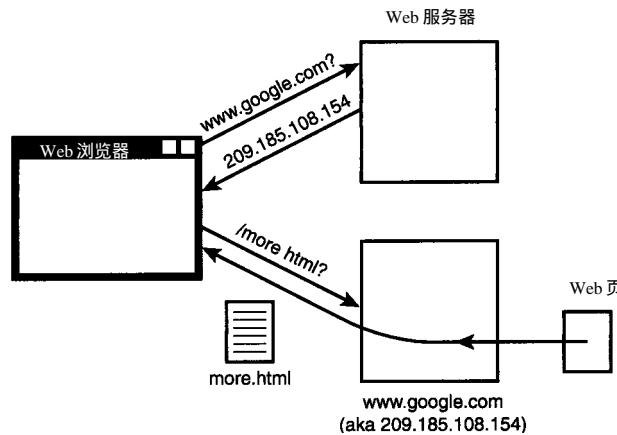
URL可以分割成下列部分：

- http 这个部分是指使用的协议。HTTP即超文本传输协议，它是传送Web页时使用的协议。你也会看到文件传输协议（ftp）或保密HTTP（https）等协议。
- www.google.com 这部分是服务器名，也称为主机名，它包含你想要的文档。有时，这部分不是主机名，而是个IP地址，通常写作4个数字，数字之间用圆点隔开，比如209.185.108.147。不过这些地址不如主机名那样可靠。
- : 80 这部分是个端口号，用于确定你的客户机与服务器是在哪个端口上互相进行连接。这部分通常是可有可无的。使用的协议决定了使用什么端口。http协议通常使用端口80。
- more.html 这部分是指对服务器提出的请求。通常这是你想检索的一个文档。有时它写作一个路径名，例如/archives/foo.html，也可以用其他字符作为结尾（？&），不过它基本上指客户机要求向服务器检索的文档。

这时客户机为http执行下列操作步骤（见图17-2）：

- 1) 主机名（www.google.com）转换成IP地址。
- 2) 使用IP地址和端口号与www.google.com上的服务器建立连接。
- 3) 向服务器提出检索Web页more.html的请求。客户机等待服务器应答。
- 4) 服务器发出应答，在上例中，服务器发出more.html的内容，然后断开与服务器的连接。
- 5) 客户机在屏幕上显示服务器应答的Web内容。

图17-2 客户机向服务器提出检索Web页的请求



下载

客户机与服务器之间进行“通信”的详细情况将在第20学时中介绍。

### 17.1.2 动态Web页——使用CGI

在检索通常的Web页时，服务器只是根据你想要的文档并从它的磁盘存储器上检索这个文档，然后将它发送给客户机，如图17-3所示。

图17-3 检索静态Web的示意图

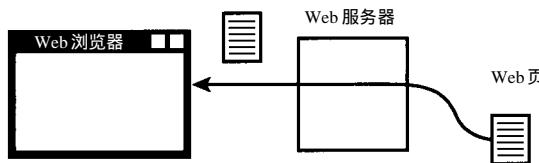
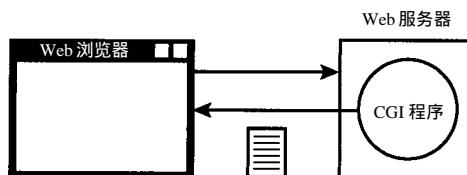


图17-3中的服务器根本不对数据进行任何处理，它只是查看客户机提出的请求，并将请求的数据传送给客户机。

在Web上创建动态内容的方法之一是使用CGI程序。CGI是Web服务器用来在服务器上运行程序以便生成Web内容的公认的方法。当URL指明CGI应该在它上面运行CGI程序来生成Web内容的服务器时，该服务器就启动该程序运行，该程序则生成Web内容，然后服务器将内容传递给客户机，如图17-4所示。

图17-4 CGI脚本生成的Web页



每当客户机请求检索一个实际上是CGI程序的Web页时，便出现下列操作：

- 1) 服务器启动CGI程序的一个新实例。
- 2) CGI程序使用它需要的信息生成一个Web页，或者生成另一个应答。
- 3) 该Web页被送回给客户机。
- 4) CGI程序退出。

CGI程序可以是任何类型的程序。它可以是个Perl脚本，就是你将要在这里学习的一项内容。它也可以是用C、UNIX shell，pascal，LISP、TCL或任何其他编程语言编写的程序。而许多CGI程序是用Perl编写的，这完全是一种巧合。Perl恰好非常适合编写用于文本处理的程序，而CGI程序的输出常常是文本。

CGI程序的输出几乎可以是以任何形式的信息。它可以是图形、HTML格式的文本、压缩文件、流式视频信息，或者你在Web上找到的任何其他类型的内容。总的来说，你编写的CGI程序将生成HTML格式的文本。



CGI不是一种语言，它与Perl之间并不存在特殊的关系，与HTML语言也没有任何关系，与HTTP之间也没有多少关系，它只是Web服务器与代表服务器运行的程序之间的一个公认接口。CGI的技术规范由美国国家超级计算机应用中心维护，该中心的网址是[http://www.ncsa.uiuc.edu/cgi\\_interface.html](http://www.ncsa.uiuc.edu/cgi_interface.html)。你可以在后面7个学时中了解CGI的详细特性。

## 17.2 不要跳过这一节内容

当你准备编写 CGI程序时，首先必须搞清几个问题，否则，第一次编写 CGI程序的经历一定不会使你感到愉快。你预先查找这些信息是比较容易的，而在调试程序时要搞清这些问题则要困难得多。

若要使用CGI，必须拥有Web服务器。CGI编程新手遇到的常见问题是他们试图在 Web服务器没有正确安装的情况下就编写 CGI程序。若要获取 Web服务器，可以从两种方法中选择一个。可以在商用 Web服务器上租用一定的空间，也可以运行自己的服务器。决定权在你的手里，可以根据你愿意支付的费用，需要的带宽，以及你的技术熟练程度来作出决定。

若要获得商用 Web服务器，你可以搜索 Web并查找一个。这些商用服务器常常称为“ Web托管”公司，它们收取的费用和提供的特性可以根据情况而千差万别。如果你打算编写 Perl CGI程序，应该确保Perl 5将作为CGI的编程语言。很少有 Web托管公司不支持Perl 5作为CGI编程语言或者根本不允许CGI编程的。应该避免使用不支持Perl 5的公司提供的服务，有许多其他公司可以供你选择。

还应该确保 Web托管公司允许你使用自己的脚本。有些公司声称它们允许使用 Perl CGI程序，但是接着又要求你使用它们公司的程序，有时要收取一定的费用。你也应该避免使用这些公司提供的服务。

另外还有一些公司，它们收取一定的费用来“审查”你的CGI程序，它之所以要收取费用，原因是你可以使用这些程序。如果你选择这些公司中的一个，应该安装你自己的服务器，以便进行相应的测试，因为“审查”是非常昂贵的。

如果你具备某些专门技能并愿意阅读全部说明的话，运行你自己的个人 Web服务器并不十分困难。首先，必须选择一个 Web服务器。如果你运行Windows，可以从几十种免费的或者接近免费的Web服务器中选择你要安装的服务器。一定要确保它们支持Perl作为CGI脚本语言。少数商用 Web服务器也可以用于Windows，比如Microsoft的Internet Information Server(IIS)。

如果你拥有一台 UNIX计算机，也可以使用少量商用 Web服务器。请与 UNIX供应商联系索取Web服务器清单。

Internet上最流行的 Web服务器是Apache，它是完全免费的。如果你拥有一个 C编译器，那么Apache Web服务器是很容易安装的。如果你习惯于编辑配置文件，那么它的运行也是非常容易的。Apache甚至可以用于 Miorosoft Windows平台。关于 Apache的信息，请访问 <http://www.apache.org>。

如果你运行自己的 Web服务器，在试图编写CGI程序之前，应该确保Web服务器运行正确，并且能够为静态 Web页提供服务。如果你的 Web服务器不能为静态 Web页提供服务，这说明你的CGI程序很可能无法运行。

还应该检查 Web服务器的配置，以确保你已将CGI脚本正确地激活。如果不激活这个特性，那么CGI编程的初学者就会遇到非常头痛的麻烦。

### 检验表

无论你是运行自己的 Web服务器，还是租用商用 Web服务器上的空间，都必须花费一点时间来完成下面这个检验表中的操作，确实如此。请将这些信息写下来，以后你就可以省去很多麻烦。

- 如果你是从商用 Web服务器主机上租用空间，那么该主机将为你提供所有的信息。这些信息可能位于主机的 Web站点上的FAQ中，也可能包含在你建立帐户时发送给你的文档中。如果你没有接收到这些信息，可以与 Web托管公司联系索取这些信息。如果你想使CGI程序正确运行，获取这些信息是非常重要的。
- 如果你已经配置和安装了自己的Web服务器，那么这些信息应该是配置进程的组成部分。如果你遇到了问题，请查看一下是否能够找到一个回答这些问题的FAQ，或者是否能够找到要检验的配置文件。

如果你要进行CGI编程，必须知道下列信息：

- Web服务器上Perl的位置 你必须知道Perl解释程序安装在Web服务器上的什么位置。由于你必须修改程序中的#！行代码以便反映该路径的情况，所以必须知道这个信息。如果你的Web托管公司运行Microsoft公司的操作系统，那么你可以不需要这个信息。
- Web服务器日志文件的位置 如果不知道Web服务器的错误日志保存在什么地方，就很难调试你的CGI脚本程序。你应该设法找到这个位置，这很重要。
- 用于CGI程序的扩展名 Web服务器有时要将服务器上保存的普通静态Web页与根据文件名来运行的CGI程序区分开来。CGI程序的扩展名通常是.cgi或.pl。有时则根本不用扩展名。
- CGI程序目录的位置 Web服务器有时需要CGI程序文件名的扩展名，有时则需要将文件放在一个专门的目录中。（很少同时需要这两者。）该目录通常称为 /cgi-bin，并且位于Web站点的顶层目录中（或者靠近顶层的目录中）。
- CGI目录的URL 许多情况下，你使用的Web服务器的URL中CGI目录附加在它的结尾处，例如：

http://www.myserver.com/cgi-或者http://www.myserver.com/cgi/.

### 17.3 编写你的第一个CGI程序

了解了上述关于CGI编程的有关说明、注意事项、检验表等信息后，你就可以准备键入你的第一个CGI程序了。程序清单17-1显示了这个程序。

键入这个程序并将它保存为hello。如果在检验表中必须将某个扩展名用于CGI程序（正如“用于CGI程序的扩展名”项中所说的那样），那么请使用该扩展名。这样，如果必须使你的CGI程序的文件名带有.cgi扩展名，那么将该脚本程序保存为hello.cgi。如果必须使用扩展名.pl，请将该脚本程序保存为hello.pl。

程序清单17-1 你的第一个CGI程序

```
1:  #!/usr/bin/perl -w
2:  use CGI qw(:standard);
3:  use strict;
4:
5:  print header;
6:  print "<B>Hello, World!</B>";
```

第1行：这一行是个标准#！行。你必须替换检验表中的“Web服务器上的Perl位置”信息项中的路径，使该脚本程序能够运行。当然-w用于激活警告特性。

第2行：CGI模块纳入了该程序。qw（:standard）使得一组标准函数从CGI输入到你的程

序中。

第3行：use strict是个很好的编程命令，对于CGI程序来说也是一样。

第5行：从CGI模块中输入header函数。它输出一个标准标题，服务器（和客户机）必须看到它后才能处理CGI程序的输出。

第6行：当标题输出后，所有输出就会正常显示在浏览器中。在本例中，当 CGI运行时，浏览器将显示Hello world。

这就是各个代码行的具体内容。

不过，事情并没有结束，你还必须安装这个CGI程序并对它进行测试，你的工作只完成了半。

### 17.3.1 在服务器上安装CGI程序

究竟如何安装CGI程序，主要取决于你拥有何种服务器，你是否能够在本地访问它，或者是否只能用FTP将文件发送给该服务器。下列各节将介绍如何为不同的环境安装CGI程序。

#### 1. 本地访问UNIX Web服务器上的文件系统

如果你能够使用telnet、rlogin或其他方法登录到UNIX Web服务器上去，请使用下列说明来安装CGI程序：

1) 使用FTP，将CGI程序hello.cgi（或hello.pl）放在UNIX服务器上。也可以使用vi将该程序写入服务器，这也是个好方法。

2) 使用mv或cp命令，将CGI程序转到正确的目录中。你应该在“CGI程序目录的位置”下的检验表中找到正确的目录。

3) 在UNIX下，必须使该程序成为可执行程序。可以使用下面这个命令来执行这项操作：

```
chmod 755 hello.cgi
```

如果该程序的名字是hello.pl，则在该命令中使用该名字。该命令使得文件所有者能够写入该文件，而其他人则可读取和执行该文件（对于CGI程序来说，这是正确的）。

#### 2. 只能用FTP来访问UNIX Web服务器

如果你只能使用FTP来访问服务器，请按下列说明来安装CGI程序：

1) 使用你的FTP客户程序将hello.cgi（或hello.pl）程序转入CGI程序目录。你应该已经在“CGI程序目录的位置”下的检验表中找到正确的目录。务必以文本方式或 ASCII方式来传送文件，不要使用二进制方式将CGI程序传送到服务器中。如果使用文本方式的FTP实用程序，那么它的默认方式通常是文本方式。

2) 必须使CGI程序成为可执行程序。对于仅为文本的CGI程序，下面这个命令应能运行：

```
quote site chmod 755 hello.cgi
```

如果hello.pl是该程序的名字，则上述命令应该使用hello.pl。该命令使得该文件可供文件所有者写入，而其他所有人则可以读取和执行该文件（对于CGI程序来说，这是正确的）。

3) 如果你拥有一个图形FTP程序（如Cute-FTP），必须找到Set Permissions（设置访问许可权），Change Mode（改变方式），Set File Attributes（设置文件属性），或Set File Access Mode（设置文件访问方式）等选项卡，以便设置访问许可权。

不管用何种方法设置访问许可权，文件所有者需要读/写/执行权限，用户组需要读/执行权限，其他用户需要读/执行权限。如果该程序需要数字式访问权限，请使用755。

**下载**

### 3. 本地访问NT Web服务器上的文件系统

如果你能够本地访问NT Web服务器的文件系统，请使用NT的Explorer或文件拷贝实用程序将CGI程序放入正确的目录，即“CGI程序目录的位置”中指定的这个目录。

### 4. 只能使用FTP来访问NT Web服务器

如果你只能使用FTP来访问NT Web服务器，请使用FTP客户程序将hello.cgi（或hello.pl）程序放入CGI程序目录。你应该已经找到“CGI程序目录的位置”下的检验表中的正确目录。务必用文本方式或ASCII方式来传送文件，不要使用二进制方式将CGI程序传送到服务器。如果使用文本方式的FTP实用程序，其默认方式通常是文本方式。

## 17.3.2 运行你的CGI程序

若要了解你的CGI程序是否能够运行，请打开浏览器，并将它指向你在检验表中设定的地址，即CGI目录的URL，并将CGI程序名附加在该URL的后面。例如，可以输入下面的URL：

`http://www.myserver.com/cgi-bin/hello.pl`

当保存CGI程序时，应该使用hello.cgi或你用于CGI程序的任何名字。

这时会发生下列两种情况中的一种：

- 1) 你的浏览器加载一个带有Hello, world消息的Web页。
- 2) 它没有加载这个Web页。

如果你的CGI程序不能运行，不管原因是什么，请查看下一节的说明。下一节专门介绍如何查找类似这样的程序问题。CGI程序的安装和调试过程非常困难，而且的确很难。不过你不要灰心，因为CGI程序的运行不会是一帆风顺的，你应该坚定信心。一旦排除了CGI程序的故障，你将不必重新对它进行调试。

如果你的CGI程序能够按照要求来运行，那就太好了。这说明你已经成功地安装了你的Web服务器和CGI程序，并且使它们能够正确运行。不过你仍然应该浏览下一节的内容。总有一天，你的某个CGI程序可能发生故障，你至少应该熟悉诊断程序故障的操作步骤。

## 17.4 CGI程序无法运行时怎么办

下面几节为你提供一个CGI程序的通用调试指南。在你阅读所有这些内容以便找出你的第一个CGI程序中的问题之前，请回头看一看前面的内容，以确保没有跳过任何步骤。当你学到本学时结尾的时候，应能发现你的CGI程序中存在的任何问题。

这几节中介绍的诊断操作均假设你要调试的CGI程序名是hello.cgi。如果你的程序使用别的名字，请改过来。

### 17.4.1 这是你的CGI程序吗

- 第一个需要解决的产生问题的原因是CGI程序本身。如果CGI程序不能运行，那么调试Web服务器的配置是毫无意义的。

CGI程序可以像所有Perl程序那样以交互方式来运行，用交互方式来运行CGI程序对于程序的调试来说是非常有用的。若要运行你的CGI程序，请在命令提示符处输入下面这个命令，将它启动：

```
perl hello.cgi
```

这时Perl解释程序应该输出下面这行信息作为应答：

```
(offline mode: enter name=value pairs on standard input)
```

这个提示行表示CGI模块试图获取你的CGI窗体的值。这些值将在第18学时中介绍。

看到这个提示后，你应该输入文件结束字符作为应答。在UNIX下，它是Ctrl+D，你只需按下Ctrl键并键入D。在Windows中，可以按下Ctrl+Z。然后Perl应该输出下面这两行消息：

```
Content-Type: text/html
```

```
<B>Hello, World!</B>
```

Content-Type : text/html这条消息表示后面的信息应该转换为文本或HTML。这条消息的含义将在第20学时中全面介绍。现在，你只需要知道重要的是这条消息是你的程序用header函数输出的“第一个”信息，并且这条消息是必须输出的。如果在Content-Type消息之前输出了别的什么消息，那么CGI程序的运行将会失败。

问题：Perl应答的语句有误。

解决办法：找出语句错误。

问题：Perl应答的信息是Can't locate CGI.pm in @INC...(在@INC中无法找到CGI.pm...)。

解决办法：你安装的Perl不完整。Perl配有默认的CGI模块。如果你想要安装它，请参阅本书的附录。

#### 17.4.2 服务器存在的问题

当排除了你的脚本程序是问题的根源之后，就应该检查脚本程序的安装和服务器的配置是否正确。

问题：服务器应答的消息是Not Found(未找到)或404 Nat Found(404未找到)。

解决办法：这些消息通常表示存在下列问题之一：

- 你使用的URL不正确。当你应该键入http://www.server.com/cgi-bin/hello.cgi时，你却键入http://www.server.com/cgi/hello.cgi。请返回到检验表，核实你的CGI目录的URL是否正确。
- 你将脚本程序放入Web服务器上的目录不正确。请核实检验表，确定CGI程序的目录是否正确。如果不正确，请将脚本程序转到正确目录中。

问题：你的脚本程序的文本显示出来了。

解决办法：之所以显示该程序，原因是Web服务器认为该程序实际上是个文档。

- 你使用的CGI程序扩展名错了。你没有使用.pl，而是使用了.cgi或者其他错误的扩展名。请查看检验表，确保你使用了正确的CGI程序扩展名。
- 你将脚本程序放入了不正确的目录中了，同时使用了错误的URL来访问它。请将脚本程序放入正确的CGI程序目录中，并且确保你使用的URL是正确的。
- 服务器配置有误。如果你是使用自己的Web服务器，请重新阅读它的文档，并核实你的Web服务器安装是否正确。有时安装服务器时包括了一个测试用的CGI脚本程序。如果是这样，请测试这个CGI脚本程序。如果你使用一个商用Web托管服务器，请核实你将脚本程序放入了正确的目录之中，否则与Web主机联系，请求其帮助。

问题：服务器应答的信息是Forbidden(禁止)或403 Error(403错误)。

解决办法：对CGI程序的访问权限设置不正确。这个问题最有可能出现在 UNIX Web服务器上。

可以查看对hello.cgi程序的访问权限，方法是在命令提示符处键入 ls -l hello.cgi。如果你拥有对服务器的FTP访问权，可以查看文件访问权，方法是键入 dir。该访问权限应该类似下面的形式：

```
-rwxr-xr-x    1 user          93 Aug 03 23:06 hello.cgi
```

访问权限是左边的字符 rwxr-xr-x。如果不是这样，请回到安装说明，详细了解如何正确地设置对CGI程序的访问权限。

#### 17.4.3 排除服务器内部错误或500错误

如果服务器应答的消息是 Internal Server Error ( 服务器内部错误 ) 或 500 Error ( 500 错误 )，这意味着你的CGI程序运行失败了。这个通用故障消息是由许多不同问题产生的。

检查“ Internal Server Error ”时使用的最重要工具是服务器的日志文件。当 Web 服务器收到客户机要检索 Web 页的请求时，它就会将每个请求写入一个文件，供以后分析时使用。服务器遇到的任何错误也会记录在这个文件中，包括 CGI 程序生成的出错消息。

请查找服务器的出错日志文件的位置，你在检验表中应该看到了这个文件位置。日志文件的编写通常是将新的项目附加在日志文件的底部。若要查看 UNIX 下的最后几个日志文件项目，请在提示符后面键入下面这个命令以便查看日志文件的底部的项目：

```
tail server_log
```

有些Web服务器配有一个实用程序，它常常是 CGI 程序本身，用于查看日志文件。如果你只拥有对服务器的FTP访问权，那么必须下载该日志文件，并在你的本地 PC 上查看该日志文件，以便找出错误项。

如果你无权访问服务器的错误日志文件，那么就存在一个很大的隐患。查找“ Internal Server Error ”将是一件漫无边际的工作。按照下面显示的检验表，最终你应该能够找到存在的问题。（你在服务器的日志文件中找到的消息是不精确的信息，不同的服务器的消息文本各不相同。）

日志项：No such file or directory : exec of /cgi-bin/hello.cgi failed ( 没有这个文件或目录：/cgi-bin/hello.cgi运行失败 )

出错的原因：

- 脚本程序的 # ! 行可能不正确。应该确保 # ! 行中 Perl 的位置与检验表中 Web 服务器上的 Perl 位置相一致。通过使用 FTP 中或本地的 ls 或 dir 命令，核实 Perl 实际上已经安装在该位置上了。
- 如果你使用 FTP 将 CGI 程序传送到服务器，可能没有使用 ASCII 方式进行传输。用二进制方式将 Windows 中编写的脚本程序转移到 UNIX 服务器（并反方向传送），这是行不通的。
- 对 CGI 程序的访问权设置不正确（ UNIX 下）。请在“ 服务器存在的问题 ” 这一节中查看关于 Forbidden 的说明。

日志项：Can 't locate CGI.pm in @INC...(@inc中找不到CGI.pm)。

出错的原因：

- 安装的Perl不完整，受到了破坏，或者太旧了。显然Perl无法找到CGI模块，CGI模块是Perl的标准组成部分。你必须重新安装该模块，或者与系统管理员联系，请他重新安装Perl。安装方法请参见本书附录。

日志项：Syntax error, warning, Global symbol requires, etc(语句错误、警告、需要全局符号等)。

出错的原因：

- 你的Perl程序显然存在键入错误或语句不正确的问题。请按“这是你的CGI程序吗”这一节中的说明，确定问题的性质。

日志项：Premature end of script headers (脚本程序标题过早结束)

出错的原因：这个出错消息说明了这样一种情况，即你的脚本程序在运行，而CGI模块的header函数输出的Content-Type标题并不是脚本程序发出的第一个消息。有时在日志文件中的这个消息前面或后面还会出现一条辅助消息。这个辅助消息更有助于确定出错的原因。你可以试用下面的方法来确定出错的原因：

- 在调用header函数前，务必不要输出任何信息，包括出错消息。在header函数之前输出的任何东西都会导致这个错误。



在程序的开始处而不是在调用header函数时，你会看到Perl CGI程序输出“Content-Type : text/html\n\n”这条消息。显然输出这个消息和调用header函数被认为是做同样的事情，但实际并非如此。header函数要考虑这样一个问题，即\n\n在每个服务器上并不总是表示相同的意思，它会为该服务器输出相应的序列。

- 一个称为输出缓冲的问题会导致system函数在header函数输出之前产生输出，并在输出中出现反引号(`)。若要确保header函数的输出总是显示在前面，可以将Perl CGI程序的开始部分重新编写为下面的形式：

```
#!/usr/bin/perl -wT
use strict;
use CGI;

$|=1;      # ensures that header's output always prints first
print header;
```

## 17.5 课时小结

在本学时中，我们介绍了CGI程序如何运行的基本知识，讲述了静态Web页与动态Web内容之间差别，并且在后面几个学时中还要进一步阐明这些问题。你还编写了第一个CGI程序并且使它运行了起来。

此外我们还提供了如何调试CGI程序的指南，这对今后几个学时的学习来说是非常有用的。

## 17.6 课外作业

### 17.6.1 专家答疑

问题：我没有加载CGI模块，是否必须使用该模块？

**下载**

解答：坦率地说，你确实必须使用该模块。CGI并不是个很容易使之正常工作的模块。目前已经发布的许多程序试图仿制CGI模块的功能，可惜都不太成功。它们存在着大量的安全漏洞，并且无法实现兼容。此外，它们不符合Internet标准。在第16学时中，我们讲述了为何“仿制车轮”是件并不高明的事情。CGI是个很难仿制的车轮，我们都无法第一次或者第100次使之恢复正常。

标准产品中包含的Perl CGI模块已经被成千上万个程序开发人员测试过，非常耐用，你可以放心地使用它。

本书附录中讲到，如果必要的话，你可以安装只供你自己使用的CGI模块。你没有理由不使用这个模块。本书中的所有代码例子都需要使用CGI模块，有关说明的前提是你已经安装了该模块。

问题：我拥有cgi-lib.pl的拷贝，可以使用它吗？

解答：你不应使用它。cgi-lib.pl的所有函数都在CGI模块中。cgi-lib.pl库非常老，并且得不到维护。

问题：为什么人人都必须将Perl用于CGI？为什么不使用C或TCL？

解答：Perl的特性对CGI特别有用。这些特性主要包括：

- Perl具有非常出色的文本处理功能。
- 你很快就会了解的Perl的出色功能将使它成为编写CGI程序的安全语言。
- Perl是一种优秀的“胶水”语言，它非常适合将操作系统的实用程序、数据库和CGI等不同技术组合在一起。
- Perl很容易使用。

问题：如果我遇到了关于Perl和CGI方面的问题，是否应该将一条消息发送给comp.lang.perl.misc新闻组？

解答：也许不合适。更合适的新闻组是comp.infosystems.www.authoring.cgi。不过首先你应该查看<http://www.w3.org/CGI/>网址上的FAQ。

### 17.6.2 思考题

- 1) CGI程序可以用下面的语言编写：
  - a. 只能用Perl、UNIX Shell或C语言。
  - b. 只能用C语言。
  - c. 能够在服务器上运行的所有编程语言。
- 2) Web是在Perl之前问世的。
  - a. 是。
  - b. 否。

### 17.6.3 解答

- 1) 答案是c。Perl并不是编写CGI程序时使用的惟一语言，它的某些特性在编写CGI程序时更加容易并且更加可靠。
- 2) 答案是b。Perl是在1987年开发而成的，而Web直到1991年才在CERN问世。

#### 17.6.4 实习

对“Hello, world!”程序稍作修改，增加一些功能。使用localtime输出当前时间，并且用HTML标记给它增添某些颜色和一两个表格，要有创意。请记住，在你的Perl程序中输出HTML，可以在加载Web页时使该程序出现在最终的Web页中。