

利用 PHP 在 Internet/Intranet 上 开发动态 Web 页的技术

王亚斌 刘明杰 (北京理工大学机电工程学院 82 教研室 100081)

摘要: 本文阐述了目前 Internet/Intranet 上新一代动态网页开发技术 PHP 的运行机制, 以较丰富的实例论述了 PHP 的工作环境, 语法结构和 PHPLIB 的基本组成及其开发方法。

关键词: PHP 动态网页 嵌入式脚本 电子商务

1 引言

近年来, 随着互联网技术的成熟和发展, 人们越来越多地通过浏览器查看 Internet/Intranet 上的网站资源, 其中如何根据用户请求实现动态数据交换成为一个网站建设者首先需要考虑的问题, 如网上数据检索、基于电子商务的商品订购、身份认证等, 这些应用都和数据库密切相关。目前实现这种机制的方法主要有: 传统的 CGI (公共网关接口), 与 ASP (Active Server Pages), PHP 及基于 Java 的 JSP。

但是传统的 CGI 和 ASP 存在着自身无法克服的缺陷:

(1) 传统的 CGI 一般用 C/C++、Perl 等语言编写, 开发周期长, 难度大, 与网页的结合性能差。而且在工作时, 它根据浏览器的 HTTP 请求激活响应过程, 每一个请求对应一个进程。这样当存在很多请求时, 就会因进程过多, 而占用大量的系统资源, 导致数据传输不稳定, 效率低下。

(2) ASP 目前仅支持 Windows 系列操作平台。一些采用 Unix 和 Linux 服务器的大型网站还无法使用 ASP 作为网站的开发工具。而且在安全性方面, ASP 存在着很大的隐患, 它没有经过微软公司的 IIS Service Pack 处理, 在 ASP 的网址后加上 “: ; \$DATA” 的字符串, 就可以看到 ASP 的源代码。这一方面使得源代码容易被他人抄袭, 另外一个更严重的问题是, 这些源码中可能包含不应公开的字符、甚至密码, 如果让黑客窃取这些信息后, 就可能造成无法弥补的损失。

PHP 是由一些开放源代码程序员开发出的一种全新的动态网页开发技术。它支持多种系统平台, 包括:

Windows 系列、Unix 和 Linux 等, 这使得采用 PHP 语言编写的程序可以很方便地在不同系统平台之间进行移植。PHP 具有强大的数据库支持功能, 它支持当前流行的各种数据库, 并为它们分别设计了专门的函数, 使得操作这些数据库变得非常方便。而且, PHP 还内置了对文件上传、密码认证、Cookies 操作、邮件收发、动态 GIF 生成等功能的支持。更为重要的一点是, PHP 具有很好的安全性, 它使用了操作系统的安全特性, 如果有黑客要对运行 PHP 的 Web 服务器进行攻击, 他必须先攻击操作系统, 而攻击操作系统要比攻击一个网络服务进程困难得多。

2 安装调试 PHP 的工作环境

一般来说, 进行 PHP 程序的开发工作需要这样几个环境: 程序源码编辑环境、支持 PHP 的浏览器环境、PHP 自身的工作环境。绝大多数操作系统都为 PHP 提供了很好的程序源码编辑环境和浏览器环境, 在此就不赘述。下面分别就 Windows 和 Linux 平台介绍 PHP 工作环境的安装和调试:

2.1 在 Windows 下工作环境的构建

先去 PHP 的官方网站免费下载一份 PHP 在 Windows 下的安装程序, 将安装程序解压缩到本地磁盘的一个目录下, 例如 C:\PHP 目录下。在这个目录下, 有许多带有 .dll 后缀名的库文件, 将这些库文件复制到 C:\Windows\system\目录下 (这是在 Windows 9x 中的情况, 在 Windows NT 中, 则是在 Windows\system32 下), 只有放置在这个目录下, 这些库文件才能够被 Windows 所调用, PHP 程序才可以正常运行。

在 PHP 目录下还有一个 php.ini-dist 文件，这是 PHP 的配置文件。将这个文件移动到 Windows 目录下，并改名为 php.ini。打开这个文件，修改“extension_dir=.”为“extension_dir=c:\Windows\system\”。修改“;extension=php_mysql.dll,”除去前面的注释号。在 [mysql] 栏下修改

```
mysql.default_port = 3306
mysql.default_host=localhost
mysql.default_user=root
```

现在就完成了 php.ini 的配置工作，通过配置 php.ini 可以保证 PHP 程序的正常运行，并使 PHP 支持目前互联网上最流行的 MySQL 数据库。接下来配置 Web 服务器，使其支持 PHP 的程序。如果使用的是 IIS 服务器，可以直接用鼠标右键单击 php_iis_reg.inf 文件，从弹出的快捷菜单中选择“安装”。如果使用的是 PWS 服务器，修改 pws-PHP4.reg 文件，将“.php = [*****] \\ PHP 4isapi.dll”改为“.php=c:\\php\PHP 4isapi.dll”。加入行“.php3=c:\\php\PHP 4isapi.dll”。再用鼠标右键单击 pws-PHP 4.reg 文件，选择合并，这样就完成了 PHP 在 Windows 平台下工作环境的构建。进行完这些设置之后，重新启动系统，使更改的配置生效。

2.2 在 Linux 下工作环境的构建

在 Linux 下，PHP 的发行和安装一般是发布源码，然后从源码开始编译，编译后再安装，这样虽然有些复杂，但它允许用户对整个 PHP 安装进行定制，选择自己需要的功能，使 PHP 服务变得短小精湛，提高执行效率，获得更高的安全性。

将预先准备好的 PHP 4.0.x 源程序和 Apache 1.3.x 源程序复制到 /usr/src 目录下，按照下列步骤编译、安装：

```
tar -zxvf php-4.0.xbs.tar.gz // 解开压缩文件
tar -zxvf apache_1.3.x.tar.gz
killall httpd // 中止 httpd 服务进程
cd /usr/src/apache_1.3.x
./configure -prefix=/www/apache // 将 Apache 安装在 /www/apache 目录下
cd ../php-4.0.x
./configure --with-mysql --with-apache=../apache_1.3.x --enable-track-vars // 编译 PHP 将其作为 Apache 的一个模块运行，并支持 MySQL 数据库
make
make install
```

```
cd ../apache_1.3.x./configure -prefix=/www/apache -
activate-module=src/modules/PHP 4/libPHP 4.a
make
make install
cd ../php-4.0.x
cp php.ini-dist /usr/local/lib/php.ini //将 php.ini-dist 文件复制为 /usr/local/lib/php.ini 文件
```

这样基于 Apache 内嵌模块的 PHP 安装完毕，接下来对 PHP 和 Apache Web 服务器进行配置。用编辑软件打开 /usr/local/lib/php.ini 文件，对 [mysql] 栏下内容进行与其在 Windows 下相似的改动，保存文件。打开 /www/apache/conf/http.conf 文件，去掉“# AddType application/x-httpd-php.php”行的“#”号，加入行“AddType application/x-httpd-php .php 4”。改变完这些配置后，重新启动计算机，使这些配置生效。

3 PHP 的运行机制

PHP 是一种跨平台的服务器端嵌入式脚本技术。它和 HTML 语言共同存在于一个 .php 文件中，当客户浏览器通过 HTTP 向 Web 服务器提出一个请求时，Web 服务器启动 PHP 解释器，将 .php 文件中的 PHP 脚本解释为 HTML 语言，然后向客户浏览器返回一个自己生成的 HTML 文件。整个 PHP 脚本的解释过程均在服务器端进行，客户浏览器端只能看到从服务器返回的 HTML 页面。图 1 说明了这一工作过程：

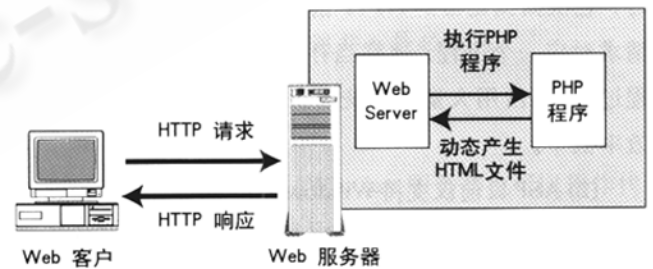


图 1

4 PHP 程序设计语言的语法

PHP 文件是一个文本文件，主要由文本、HTML 标志、脚本命令三部分组成。编写一个 .php 文件十分简单，只要将 HTML 文件的后缀名改为 .php 就得到了一个没有脚本命令的 .php 文件。它不需要经过编译、连接等复杂的过程，只要将它保存在 Web 服务器指定的目录下，确保

该目录有读 (READ) 和执行 (EXECUTE) 的权限, 就可以让远程用户访问。

但是一个真正的.php文件除了包含HTML中所含有的文本、HTML标志外, 还应包含PHP脚本命令, 以便实现与客户端进行动态交互的功能。在一个.php文件中, 有四种方法可以将PHP脚本嵌入HTML文档, 下面通过一个例子加以说明:

```
<HTML>
<HEAD><TITLE>嵌入方法</TITLE></HEAD>
<BODY>
<? echo (" 这是一个 PHP 语言的嵌入范例\n"); ?>
<?php echo(" 这是第二个 PHP 语言的嵌入范例\n");
?>
<?script language="php"&GT
echo ("这是类似JavaScript及VBScript语法的PHP语言嵌入范例");
/script&
<% echo ("这是类似ASP嵌入语法的PHP范例"); %>
</BODY>
</HTML>
```

其中在嵌入符号中的语句就是PHP脚本, 它由Web Server所提供的脚本引擎 (Scripting Engine) 解释, 在Web Server端处理, 生成的HTML语言被送回客户端供客户浏览。编写PHP程序的最好方法, 就是先处理好纯HTML格式的网页文件后, 再将需要使用变更或其他处理的地方改成PHP程序。

5 PHP 的包含文件

PHP有一个和C语言非常类似的地方, 它可以包含文件。这使得开发者能够将以前做过的工作结合在一起, 很轻松地组建出新的功能。PHP包含文件的方法有两种: require 和 include, 它们的使用方法如下:

(1) require("xxxx.php")。其中的“xxxx.php”代表任意的一个包含了基本功能的PHP文件。它的含义是在PHP程序执行之前, 首先读入require所指定的文件的内容, 然后将这些内容和网页中的PHP代码段一起解释, 形成HTML格式的文件。

(2) Include("xxxx.php")。其中“xxxx.php”的用法同上。这条语句的含义是在PHP程序解释过程中, 才读入相关的部分, 形成一个新的PHP文件, 然后再继续解释执行。所以该函数一般在流程控制的代码段中调用, PHP解释

引擎在读到include的文件时, 才将文件读进来。采用这种方式可以简化程序执行的流程。

6 使用 PHPLIB

PHPLIB是一种功能强大的PHP基库, 它集成了大量的类及其变量、方法, 为Web网络资源服务及管理提供了强大的技术支持, 大大方便了网页的编写, 特别是在客户身份认证、数据库访问、数据驱动等方面有较大的实用意义。应用这些特性, PHPLIB可以开发象搜狐那样大中型的、数据库支持的Web应用。所谓的大中型的数据库支持的Web应用主要有以下一些特征:

- (1) 在应用程序中包含多重条件的数据库查询;
 - (2) 程序要求从数据库中查询数据并将数据生成表格;
 - (3) 程序需要设计一个用户界面来组织生成SQL查询语句;
 - (4) 需要一个界面友好的途径来维护网页功能的完备。
- 其次, “数据库支持的Web应用”是指在应用程序中需要利用一套数据库系统进行协同工作, 该数据库系统需要满足以下条件:

- ① 可被PHPLIB支持;
- ② 基于SQL语句进行查询。

程序对数据库系统进行访问, 利用访问结果生成HTML网页, 同时还可以反过来使用HTML的窗体来驱动数据库的事务处理过程。它们相互之间的关系如图2所示:



图 2

为了真正掌握PHPLIB, 了解PHPLIB的核心结构是最重要的。

6.1 DB_Sql 类

该类用于对数据库连接进行管理, 连接设置使用默认值, 自动对结果数据进行管理。使用该类对数据库进行查询的步骤为:

- (1) 创建该类的子类来提供一次数据库连接的合适的参数。例如:

```
class DB_SAMPLE extends DB_Sql
{
```

```
var $classname = "DB_SAMPLE";  
var $Host = "sonic.bit82";  
var $Database = "sample_1";  
var $User = "wang";  
var $Password = " ";  
}
```

(2) 创建该子类的一个实例来管理访问该数据库的所有的查询。例如:

```
<?php  
$q = new DB_SAMPLE;  
$query = sprintf("select * from article like '%%%s%  
%'", $searchword);  
$q->query($query);  
while($q->next_record())  
{  
printf("<tr><td>%s</td><td>%s</td></tr>\n", $q->  
f("article_id"), $q->f("article"));  
}  
?>
```

执行完上述语句后,就将数据库 DB_SAMPLE 中表 article 中的 article_id 和 article 字段的内容显示在 Web 页面上。

6.2 Session 类

该类管理了一些会话变量,同时也提供了一组函数,这组函数可以用来从数据库中加载和向数据库中保存这些变量。Session 类依赖于 DB_Sql 类,因此在引用时,一定要提供一个 DB_Sql 的子类,用来定义访问用户指定的数据库所需要的参数。

6.3 Auth 类

该类管理会话过程中的用户身份认证。这种认证居于数据库技术,通过数据库用户名和密码相匹配的方法实现对会话用户身份的认证。它依赖于 DB_Sql 类和 Session 类。

6.4 Perm 类

该类在已经认证的会话页面中对用户许可权限进行检查。保护仅供用户访问的页面不被任意修改,保证用户享有但不超越其应有的权限。该类依赖于 DB_Sql 类、Session 类和 Auth 类。

6.5 User 类

该类管理由用户自己定义的变量。这些自定义的变量与特定的用户 id 号相关联,而不是像会话变量那样是

和特定的会话 id 号码相关联。这些变量可以在多个会话期间持续存在,但是只有通过身份认证的用户才能获得它们。该类依赖于 DB_Sql 类、Session 类、Auth 类。

6.6 Car 类

该类用来进行处理简单的商店购物车的应用。这个购物车可以把东西放入车内,也可以把东西从购物车内清除,并且还可以列举购物车内的物品。Car 类中含有一系列实例方法,对编程人员开发一个基于电子商务架构的购物车十分有用,下面简单加以介绍:

(1) check(\$art)。检查一个由商品号码变量 "\$art" 指定的商品是否在这个购物车中。函数返回一个布尔类型的数组和一个整型数字。若布尔值为真,表示小车内存在这种商品。

(2) add_item(\$art,\$num)。向购物车内加入 "\$num" 件商品,返回由 "\$art" 所指定的商品在购物车中的位置。

(3) remove_item(\$art,\$num)。将购物车中 \$num 件 \$art 所指定的商品移出,如果购物车中没用足够的这种商品,则该函数返回布尔值“假”。

(4) show_all()。显示当前购物车中所有的商品。

(5) show_item(\$art,\$num)。这个函数由用户提供。使用 HTML 标记语言来显示一个购物车内由 \$art 所指定商品的数量。

(6) show_car_open()。这个函数由用户提供。用 HTML 标记语言来为显示购物车的商品列表附加一个序言。

(7) show_car_close()。这个函数由用户提供。用 HTML 标记语言来为显示购物车的商品列表附加一个尾声。

6.7 Template 类

该类管理模板和变量替换。模板可以存储在文件中,它们按照要求载入页面,同时替换掉文件中的变量。

6.8 HTML widgets

“HTML widgets”即 HTML 窗口小部件,它是一组类的集合。这些类可以生成一些用来显示 GUI 元素的 HTML 代码,通常用来生成窗体或者表格。■

参考文献

- 1 杨世明.PHP4编程指南.北京:中国电力出版社,2000.
- 2 蒋长浩.PHP专家指南.北京:中国电力出版社,2000.
- 3 网胜工作室.自由网页编程高手PHP&MySQL.北京:北京希望电子出版社,2000.