

从入门到提高丛书

编程语言与应用程序服务器的完美结合

PHP 4.0 实例教程

熊松明 等编著

PH-43



浦东电子出版社

67

TP 312 PH-43
x68

PHP4 实例教程

熊松明 等编著

浦东电子出版社

系 列 名: 从入门到提高
书 名: PHP4 实例教程
文 本 著 作 者: 熊松明等
CD 制 作 者: 本社多媒体研究制作中心
责 任 编 辑: 秦华
出 版: 浦东电子出版社
地 址: 上海浦东郭守敬路 498 号上海浦东软件园内 201203
电话: 021-38954510, 38953321, 38953323 (发行部)
发 行 者: 新华书店上海发行所
CD 生 产 者: 上海金像光盘制作有限公司
文 本 印 刷 者: 商务印书馆上海印刷股份有限公司
开本 / 规格: 787×1092 毫米 16 开本 印张 17.5
版次 / 印次: 2001 年 2 月第一版 2001 年 2 月第一次印刷
印 数: 0001—— 8000 册
本 版 号: ISBN 7—900335 - 30 - 7
定 价: 26.80 元 (1CD, 含配套书)

前 言

网站设计制作是当前的热门话题，制作的方式已经从以前的静态页面转向基于各种网络数据库的动态页面，而在其中起重要作用的则是后台的脚本语言。这些脚本语言包括 ASP、Perl、PHP、JSP 等，而 PHP 就是其中的一支强大的生力军。

PHP 不是由任何单独的公司或组织开发出来的，它和 Linux 一样是免费的自由软件，是全世界许许多多开发者协同工作的结晶。它拥有广泛的平台兼容性并能迅速地升级，具有强大的网页制作功能。

本书主要讲解的是 PHP 4 版本，另外也对 PHP 4.03 版本的新增函数库进行了介绍。第 1 章总体地介绍了 PHP；第 2 章介绍 PHP 的安装；第 3 章介绍 PHP 的语法；这三章是使用 PHP 的基础。第 4 章介绍使用 PHP 构造网络数据库；第 5 章介绍 PHP 的编程技巧；第 6 章讲解网站应用程序的开发；这三章包括了许多 PHP 编程的实用技术。第 7 章是 PHP 4.03 版实用函数库的介绍，可供读者编程时进行参考。

本书将通过实例详细介绍如何用 PHP 开发基于网络数据库的站点，还包括许多 PHP 的实用技巧如文件传输、动态图像生成、Cookie 和 Session 的原理和使用等等。在讲解过程中兼顾基本概念的介绍和程序设计的实际技巧，适合于 Web 程序设计的入门者和有一定基础想进一步提高的程序员，对于电子商务，虚拟社区或者想制作功能较强的个人站点都会有所帮助。

本书写作的原则是学习和实践并重，因为编程是一项实践性很强的技术，只有在实际的编写过程中才能掌握相应的概念和技巧。希望读者在学习要掌握设计的思想，然后用自己的方法去实现，学以致用。

本书主要由熊松明编著。参加本书编写的还有陈平、屈博勋、乔嵩、邓淼、朱平、张云冰、张永军、许刚、赫骞、曾增、言金刚、俞涓、宋丽红、彭进展等。由于作者水平有限，书中难免有疏漏和错误之处，衷心希望广大读者不吝赐教，为本书提出宝贵意见，以便我们能够不断加以改进，使之逐步完善。

编 者
2000 年 11 月

目 录

第 1 章 概述.....	1
1.1 PHP 介绍	1
1.2 PHP 的特点和优势	3
1.3 如何学习 PHP	8
1.4 本章小结.....	8
1.4.1 主要内容.....	8
1.4.2 本章复习题	9
第 2 章 PHP 的安装和配置	10
2.1 Linux 系统下的安装	10
2.1.1 获得源代码	11
2.1.2 安装 Apache	11
2.1.3 安装 PHP	12
2.1.4 配置 Apache	17
2.1.5 其他安装方法	21
2.2 PHP 在 Windows 系统平台下的安装	22
2.2.1 PWS4 下安装 PHP	22
2.2.2 PHP 在 Windows NT/2000 下的 IIS 的安装	24
2.2.3 PHP 对 Windows 下 Apache 服务器的安装	25
2.3 PHP 的配置	27
2.3.1 配置文件.....	28
2.3.2 常规配置指令	28
2.3.3 邮件配置指令	31
2.3.4 安全模式配置指令	31
2.3.5 调试器配置指令	31
2.3.6 扩展加载配置指令	32
2.3.7 MySQL 配置指令	32
2.3.8 mSQL 配置指令	32
2.3.9 Postgres 配置指令	32
2.3.10 Sybase 配置指令	33
2.3.11 Sybase-CT 配置指令	33
2.3.12 Informix 配置指令	34
2.3.13 BC Math 数学函数配置指令	34
2.3.14 Browser Capability 浏览器兼容配置指令	34
2.3.15 Unified ODBC 配置语句	35
2.4 使用 Zend Optimizer 优化 PHP	35

2.5	测试 PHP	37
2.6	本章小结.....	39
2.6.1	主要内容.....	39
2.6.2	本章复习题	39
第 3 章	PHP 语法	40
3.1	语法概述.....	40
3.1.1	一个最简单的 PHP 程序的例子	40
3.1.2	嵌入方法.....	42
3.1.3	文件引用.....	42
3.1.4	程序注解.....	44
3.2	常量与变量.....	45
3.2.1	常量类型.....	45
3.2.2	变量类型.....	46
3.2.3	变量的使用	47
3.3	运算符.....	51
3.3.1	算术运算.....	52
3.3.2	字符串运算.....	53
3.3.3	设定运算.....	55
3.3.4	位运算	56
3.3.5	逻辑运算.....	56
3.3.6	其他运算符	57
3.4	流程控制.....	59
3.4.1	分支选择语句.....	59
3.4.2	逻辑循环语句.....	61
3.5	函数与对象.....	65
3.5.1	函数	65
3.5.2	对象	67
3.6	本章小结.....	73
3.6.1	主要内容.....	73
3.6.2	本章复习题	73
第 4 章	网络数据库.....	75
4.1	数据库系统.....	75
4.2	网络数据库.....	77
4.3	SQL 简介	79
4.4	PHP 和 MySQL 实现网络数据库	86
4.4.1	MySQL 介绍	87
4.4.2	MySQL 安装和使用	88
4.4.3	使用 PHP 对 MySQL 数据库进行操作	92
4.4.4	MySQL 数据库函数库.....	98

4.4.5	PHP 操作 MySQL 的技巧	108
4.5	PHP 和 Oracle 实现网络数据库	111
4.5.1	Oracle 介绍	111
4.5.2	Oracle 在 Linux 下的安装	112
4.5.3	Oracle 8 数据库函数库	117
4.5.4	使用 PHP 对 Oracle 数据库进行操作	136
4.6	本章小结	138
4.6.1	主要内容	138
4.6.2	本章复习题	138
第 5 章	PHP 网站编程技术	139
5.1	表单处理	139
5.1.1	HTML 中的表单元素	139
5.1.2	PHP 对表单输入的处理	143
5.2	Cookie 和 Session	145
5.2.1	Cookie	145
5.2.2	Session	148
5.3	PHP 图形处理	153
5.3.1	加入 GD 库	153
5.3.2	GD 库函数介绍	153
5.3.3	用 GD 库函数生成图像	162
5.4	文件传输和处理	165
5.4.1	文件上传表单处理	165
5.4.2	文件处理	167
5.4.3	FTP 文件传输	181
5.5	本章小结	187
5.5.1	主要内容	187
5.5.2	本章复习题	187
第 6 章	PHP 应用实例	188
6.1	数据库转换程序	188
6.2	投票系统	191
6.3	基于数据库的聊天室	204
6.3.1	聊天室工作原理	204
6.3.2	聊天室系统的实现实例	207
6.4	本章小结	234
6.4.1	主要内容	234
6.4.2	本章复习题	234
第 7 章	PHP4.03 函数库	235
7.1	PHP4.03 的新函数库	235
7.1.1	CCVS API 函数库	235

7.1.2	类/对象函数库.....	235
7.1.3	CURL, 客户端 URL 库函数库	240
7.1.4	Cybercash 支付函数库	243
7.1.5	DOM XML 函数库.....	243
7.1.6	ICAP 函数库	244
7.1.7	Verisign Payflow Pro 函数库	247
7.1.8	Pspell 函数库.....	250
7.1.9	GNU Readline 函数库.....	251
7.1.10	正则表达式函数库(POSIX 系统扩展).....	252
7.1.11	Shockwave Flash 函数库.....	254
7.1.12	YAZ 函数库.....	265
7.1.13	YP/NIS 函数库.....	268
7.2	本章小结.....	269
7.2.1	主要内容.....	269
7.2.2	本章复习题	269

第 1 章 概 述

本书的第 1 章对编程语言 PHP 作了概述，介绍了 PHP 的来龙去脉以及其发展历史、市场前景等，是学习和掌握 PHP 的第一步。阅读本章仅需要具备简单的 Internet 使用经验和 Web 编程的一些概念。对于经验丰富的程序员来说，本章只需要快速浏览一下 PHP 的主要特点就可以了。对于不太有经验的读者，其中的技术要点需要较仔细的了解。

1.1 PHP 介 绍

本节主要介绍 PHP 语言的概念和历史，并对 PHP 的强大功能和它在当前的网络中的应用做简单的分析。

PHP 是一种 HTML 内嵌式的语言（类似 IIS 上的 ASP），也就是一种超文本预处理语言（Hypertext Preprocessor）。PHP 独特的语法混合了 C、Java、Perl 以及 PHP 的新语法，可以方便快捷地编写出功能强大、运行速度快并可同时运行于 Windows、UNIX、Linux 等平台的 Web 后台处理程序。它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。

用比较简单的说法，PHP 是一种服务器端的嵌入式 HTML 脚本语言。但这个解释意味着什么呢？下面举一个最简单的例子（以下是该例的代码，例子的输出如图 1-1 所示）。

```
<html>
<head>
<title>Hello World!</title>
</head>
<body>
<?
    echo "Hello World!";
?>
</body>
</head>
</html>
```

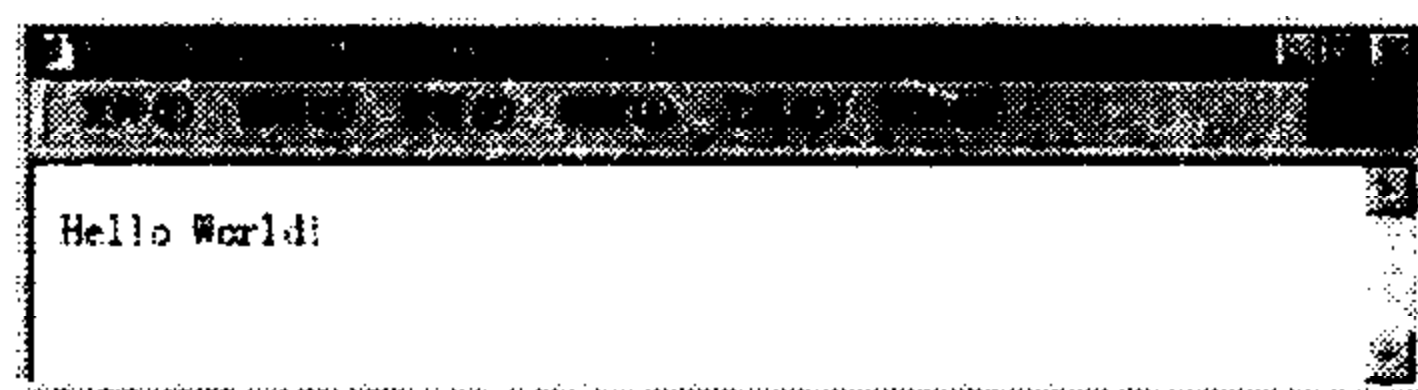


图 1-1 打印 Hello World!

PHP 的正式名称是“PHP: Hypertext Preprocessor”，即超文本预处理语言。1994 年秋季，Rasmus Lerdorf 开始构思 PHP，用在他的主页上，以追踪谁在看他的在线简历。1995

年年初第一版本出台, 当时 PHP 只被认为是个人主页开发工具。它由一个非常单纯的只能理解极少数特殊宏的分析引擎和一些用在主页后端通用的工具组成, 如留言簿、计数器和其他一些东西。这个分析器在 1995 年年中被重写并被命名为 PHP/FI 第二版。FI 来自 Rasmus 写的另外一个包, 用于解释 HTML 形式的数据。他结合了个人主页工具脚本和形式解析器, 并加上 MySQL 支持, 这样就产生 PHP/FI 了。PHP/FI 以令人惊奇的步调成长, 人们开始把自己的代码贡献给它。看来, PHP 的诞生充满了传奇色彩, 由于有一个天才人物的灵感和一大批对自由软件很热心的人们, PHP 便从无到有地产生了。在 1997 年中, 开始了第三版的开发计划, 开发小组新增了成员 Zeev Suraski 及 Andi Gutmans, 第三版就定名为 PHP3。目前, PHP 的最高版本是 PHP 4.0.3 pl1, 对应不同的操作系统也分为不同的版本。

初识 PHP, 可以看到许多 ASP 的影子, 觉得好像跟 ASP 没有什么两样。但这正说明 PHP 利用了 ASP 的长处。不同于其他像用 Perl 或 C 写的 CGI 脚本, 不需要在程序中写许多输出类似于 `print("<HTML>")` 的指令, 只需在 HTML 中加一些代码就可以了。PHP 代码被赋予特殊的起始和结尾标志, 以允许你做“PHP 模式”的交互。

PHP 和 ASP 一样, 区别于其他(像客户端 JavaScript)语言的地方是它的代码在服务器端执行。如果服务器上有一个类似这样的脚本, 客户端收到的将是这个脚本运行的结果, 而不知道背后的执行代码是什么。用户甚至可以配置 Web 服务器, 用 PHP 处理所有的 HTML 文档。

看起来, PHP 的外表似乎和 ASP 没有太多的区别。但 PHP 不同于 ASP, PHP 的语法借用 C、C++、Perl 等语言的语法, 而 ASP 完全用 Visual BASIC 的语法。PHP 在编程思想上采用的是面向过程和面向对象的混合, PHP 提供丰富的函数供开发者选择和调用, 编写属于自己的类库。而 ASP 则主要是面向对象的编程方法, 开发者主要围绕其 Session、Application、Request、Response 等对象做各种处理工作。

PHP 最大的魅力在于对大范围的数据库提供了支持, 这使得编写一个支持数据库的 Web 页面显得难以置信地容易。对于流行的多数数据库, PHP 都提供了丰富的操作函数, 用于数据库的连接, 数据记录的查询, 插入和修改等操作。与 PHP 相比, ASP 仅仅支持为数不多的几种数据库, 这些数据库也是因为 Microsoft 的支持才得以受到宠爱。

除了强大的数据库支持, PHP 还提供了丰富的网络函数库, 利用这些函数, PHP 能轻松地实现与其他协议, 像 IMAP, SNMP, NNTP, POP3, HTTP 的交互。如果用户喜欢做一些类似于底层的编程工作, 就可以利用 PHP 打开晦涩的网络接口和其他协议交互。PHP 的这一特点使得利用 PHP 开发一个很综合性的网站显得很容易。

和 ASP 一样, PHP 提供了一些常用的函数, 如数学、货币、日期时间、目录和文件操作等, 这些基本的功能为基本的开发工作提供了极大的方便。PHP 还支持动态生成图像, 可以创建 PDF 格式的文档, 这为某些有特殊要求的开发者提供了方便。

PHP 是电子商务开发者的利器。PHP 提供了加密函数库, 这些函数为开发对安全性有较高要求的电子商务网站提供了坚实的基础。和 ASP 不同, PHP 的开发者不用再利用较多的时间, 采用别的开发工具如 VC 或 VB 书写一个复杂的 ActiveX 控件或其他服务器端的程序来实现数据加密等安全性控制。PHP 只需调用几个简单的函数就可以完成这一切事情。当需要架构一个庞大的电子商务网站而解决数据加密等安全性问题时, PHP 的优势将

显得尤为突出。

PHP 还有一个很大的特点，PHP 是功能可扩展的，开发者如果觉得 PHP 功能不够强大，可以编写自己的 DLL，扩展自己定义的功能。如果有可能的话，一旦这些功能提交给 PHP 组织获准，这些功能将成为 PHP 标准功能的一部分。PHP 就是这么壮大起来的，这一点是 ASP 所不及的。

虽然与 ASP 相比，PHP 有着许多独特的优势，但是也并不是没有不足的地方。ASP 的受宠的原因在很大程度上来自于其面向对象的编程方法，其 Session、Application、Request、Response 等功能强大的对象使得许多应用开发显得极为简单。其 Session 对象的强大功用更是 PHP 一直难以比拟的特征。因此在开发电子商务应用时，和 ASP 相比，用 PHP 设计购物车一直是一件相对麻烦的事情。直到最近 PHP 在 4.0 版本中加入对 Session 的支持，情况才有所改观，PHP 也显得更加有吸引力。

PHP 至诞生以来，就一直处在发展之中。据估计到 1996 年末，整个世界至少有 1 万 5 千个网站在用 PHP/FI。到 1997 年年中，这个数字已经超过 5 万个了。而在此时 PHP 的发展也发生了变化，由 Rasmus 自己偏爱的和几个人开发的项目变成一个更有组织的团体项目。Zeev Suraski 和 Andi Gutmans 重写了解释器，这个新的解释器成为 PHP3 的基础。许多有用的代码从 PHP/FI 继承到 PHP3 中，PHP3 从 3.0.1 发展到现在的 3.0.16，其功能一直在不断完善和加强之中，其中很多是完全重写的。到 2000 年年初，采用 PHP 技术的网站已经超过了 120 万个，PHP 也完成了从版本 3 向版本 4 的升级，功能也变得更加完善和强大，同时 Zend 优化器的加入使得其运行速度大大提高。

PHP 的诞生和 Linux 大同小异，PHP 也是完全免费的自由软件，不存在 Y2K 问题。PHP 应用的最大障碍是一开始没有给用户提供一个方便的开发环境，这使得只有比较高级的网站开发者才会对 PHP 青睐有加。而 ASP 有 Visual InterDev 的强力支持，同时 Basic 的简单语法令人很容易上手，因此 ASP 拥有一个庞大的用户群。但目前面向 PHP 的开发环境已经可以找到，如 1999 年推出的 PHPED。

1.2 PHP 的特点和优势

本节介绍 PHP 语言的特点，并针对这些特点和其他的服务器端编程语言进行比较，介绍在稳定性、开发时间和系统安全等方面的优势。

PHP 作为一种较新的服务器端开发工具，主要具有以下的特点：

- 开放源码——所有的 PHP 源代码都可以得到。
- 没有运行费用——PHP 是免费的，任何人都可以访问 PHP Web 站点来获取 PHP 以及它的最新版本、源代码。
- 基于服务器端——由于 PHP 是在 Web 服务器端运行的，PHP 程序可以很大、很复杂而不会降低客户端的运行速度。
- 跨平台——PHP 程序可以运行在 UNIX、Linux、Macintosh 或者 Windows 95/98/ME、Windows NT/2000 操作系统下。PHP 与 Apache Web 服务器融为一体。Apache 也是一种免费技术，是在 Internet 上是最受欢迎的 Web 服务器之一。另一方面，PHP

在其他的 Web 服务器上同样能够良好运行，包括微软的 Internet Information Server。现在，PHP 也开始支持 ISAPI 和 NSAPI，这样就可以更好地与微软及 Netscape 的 Web 服务器融为一体。

- 嵌入 HTML——因为 PHP 语言可以嵌入到 HTML 内部，所以 PHP 很容易学习。
- 简单的语言——和 Java 和 C++ 不同，PHP 语言坚持以基本语言为基础，然而它的功能也强大到足以支持任何类型的 Web 站点。
- 效率高——和其他的解释性语言相比，PHP 系统消耗较少的系统资源。当 PHP 作为 Apache Web 服务器的一部分时，运行代码不需要调外部二进制程序，服务器解释脚本不需要承担任何额外负担。
- 分析 XML——用户可以组建一个可以读取 XML 信息的 PHP 版本。
- 数据库模块——PHP 支持的数据库见下表：

Informix 数据库

dbm 类数据库

filePro 数据库

InterBase 数据库

mSQL 数据库

SQL Server 数据库

MySQL 数据库

Sybase 数据库

ODBC 数据库链接

Oracle 8 数据库

Oracle 数据库

PostgreSQL 数据库

Solid 数据库

数据库抽象层 (DBA)

dBase 格式资料表

PHP 还可以通过 ODBC 模块支持任何 ODBC 标准的数据库。

- 文件存取——PHP 有许多支持文件存取的函数。
- 文本处理——PHP 有许多函数处理字符串，其中包括模式匹配的能力。
- 图像处理——用户可以使用 PHP 动态地创建图像。

最新的 PHP 4 是更有效的、更可靠的动态 Web 页开发工具。在大多数情况下运行得比 PHP 3.0 快，其脚本描述更强大并且更复杂，最显著的特征是速率的增加。PHP 4.0 这些优异的性能是 PHP 脚本引擎重新设计产生的结果：引擎由 AndiGutmans 和 Zeev Suraski 从底层全面重写。PHP4.0 脚本引擎——Zend 引擎——使用了一种更有效的“编译-执行”范式，而不是 PHP 3.0 采用的“执行-当解析时”模型。PHP 4 在 3.0 版的基础上增加或增强了许多有用的特征。主要有如下一些：

(1) 别名：在 PHP4 中，可以利用引用为变量赋值，给编程带来了很大的灵活性。

(2) 扩充了 API 模块：PHP 4.0 为扩展的 API 模块提供了扩展 PHP 接口模块，它比旧的 API 版本显著地快。PHP 模块已有的及最常用的接口多数被转换到使用这个扩

展的接口上。

(3) 自动资源释放: PHP4 增加了引用计数功能, 这种新技术的引入使 PHP4 具有了自动内存管理功能, 减轻了开发人员的负担。

(4) 布尔类型: PHP 4.0 支持布尔类型。

(5) 进程生成: 在 UNIX 环境下的 PHP 4.0 提供了一个很智能和通用的生成进程, 使用了一种名为基于 automake/libtool 的系统生成技术。

(6) COM/DCOM 支持: PHP 4.0 提供 COM/DCOM 支持 (仅用于 Windows 环境) 可以无缝地存取和访问 COM 对象。

(7) 与 PHP 3.0 相容性很好: PHP 4.0 与 PHP 3.0 代码向后兼容性接近 100%。由于 PHP 4 的改进的体系结构, 两者有一些细微的差别, 但是大多数人将可能永远不可能遇上这种情况。

(8) 配置: PHP4 重新设计和增强了 PHP.ini 文件, 这使得用 PHP.ini 来配置 PHP 显得极为容易, 这个文件可以在运行时被 Apache (UNIX 系统) 或由 Windows (Windows 环境) 注册。

(9) 加密支持: PHP4 实现了完整的加密, 这些加密功能是一个完整的 mycrypt 库, 并且 PHP 4.0 支持哈希函数, Blowfish, TripleDES, MD5, 并且 SHA1 是可使用的一些加密算法。

(10) 类型检查: PHP 4.0 支持同一操作符用于批类型检查: "===" (3 等号运算符), 为在两个值和其类型之间作检查。例如, "3" === 3 将视为假 (类型是不同的), 而 "3" == 3 (相等判断) 将视为真。

(11) FTP 支持: PHP 4.0 支持 FTP。通常, 你会为通过一个调制解调器连接下载一个大文件提供一个网络接口。然而, 如果你确有需要, 你可以用 PHP。

(12) PHP4 新增函数或功能增强函数: PHP 4.0 新增了许多函数, 同时也将许多现有的函数功能进行了增强。以下是一些例子: array_count_values (), eval (), foreach (), nclude (), ob_end_clean (), ob_end_flush (), ob_get_contents (), ob_start (), strip_tags (), unset ()。

(13) "Here" 打印: PHP 4.0 的 "Here" 打印是与 Perl 类似的, 尽管完全不相同。"Here" 是打印大容量文章的一个有用的方法, 例如在 HTML 文件中, 不会漏掉任何一个字符, 例如目录标记。

(14) HTTP Session fallback 系统: 为 HTTP Session 管理的一个 fallback 系统, 在 PHP 4.0 中被实现。默认情况下, Session 标识符由 cookies 存储。如果没有 cookies 支持或一项 cookies 任务失败, Session 标识符自动被创建并在 URL 的查询字符串中被携带。

(15) ISAPI 支持: PHP 4.0 能作为一个个性化的 ISAPI 模块作为 IIS 插件。这比 PHP 3.0 更有效, 它作为 CGI 运行 (一个外部的程序)。

(16) 内存: PHP 4.0 能更有效地使用内存, 导致较少的内存占用消耗, 这主要归功于引用计数技术的实现。

(17) 其他类成员函数: 在 PHP 4.0 中你能在成员函数本身的作用域或全局范围内调用其他类的成员函数。例如, 你能用一个子函数覆盖父函数, 并在子函数中调用父函数。

(18) 多维数组: 在 PHP 4.0 中, 利用 GET, POST, Cookies 进行的数据传输支持多

维数组。

(19) 个性化的 HTTP Session 支持: HTTP Session 处理, 包括 fallback 系统管理, 在 PHP 4.0 中被它的新库函数实现。在版本 3.0 中处理 Session 要求使用 PHPLIB 和第三方的库函数, 它比把 Session 直接地由 PHP 支持慢了许多。

(20) 个性化的 Java 支持: PHP 4.0 支持和 Java 的交互。这种个性化的 Java 支持为 PHP 在 Java 对象上创建和使用方法提供一个简单并且有效的工具。

(21) 对象和数嵌套组: PHP 4.0 实现了功能更加强大的对象, 移去了 PHP 3.0 中存在的句法限制。对象能在数组以内被嵌套并且反过来也如此, 可以根据你的需要实现嵌套。

(22) 面向对象的编程: PHP 4.0 为面向对象的编程和构造类及对象提供扩展的功能和新特征。PHP4 实现了对象重载、引用技术等新技术。

(23) 对象重载支持: 对象重载语法允许第三方的基于面向对象的类库使用 PHP4 的面向对象的特征存取他们自身的功能。使用这个特征的一个 COM 模块已经被实现了。

(24) 输出缓冲支持: PHP 提供了一个输出缓冲函数集合。输出缓冲支持允许你写包裹函数功能压缩缓冲区。在 PHP4 中的输出缓冲支持允许 HTML 头信息存放, 无论 HTML 的正文是否输出。头信息 (header (), content type 和 cookies) 不采用缓冲。

(25) 增加了 PCRE 库: PHP 4.0 包括一个 Perl 兼容的正则表达式 (PCRE) 库, 和正常 regex 库一起与 PHP 绑定。支持 split 和 replace PCRE 功能。PCRE 和 Perl 正规表达式之间有一些细微差别。

(26) php.ini 文件: php.ini 文件在 PHP4.0 中被重新设计, 使用 PHP 的配置 php.ini 更容易并且更有效。全部文件能被 Apache 在运行时间操作 (在 Apache 环境下) 或由 Windows 注册表 (在 Windows 下面)。被加入 php.ini 文件的配置指令自动地在所有相关的模块中被支持。

(27) 引用计数: PHP 4.0 为系统中的每个数值提供了引用计数, 包括资源。一旦一个资源不再被任何变量引用, 它自动地被释放以节省内存资源。利用这个特征的最明显的例子是一个内置 SQL 查询的循环语句。在 PHP 3.0 中, 每次递归, 另外的 SQL 结果集合重复申请内存, 直到脚本执行完毕, 这些结果集合占用的内存才被释放。

(28) 支持引用: 通过引用可以改变一个变量的值。

(29) 函数的运行时绑定: PHP 4.0 的运行时间绑定功能允许你在他们被声明以前调用, 无论声明是否在代码以后或是在运行时间。

(30) 类的运行时信息: PHP 4.0 支持在运行时刻存取下列类信息: 一个对象的类名, 一个对象的父类的类名字, 以及对象函数所在的名字。

(31) 服务器抽象层: 为支持 Web 服务器提供了增强型 SAPI (服务器 API) 接口, 是 PHP 4.0 不可分的一部分。这个服务器抽象层, 提供了通用的 Web 服务器接口支持, 支持多线程 Web 服务器, 为大多数的 Web 服务器提供透明的支持, 这些服务器包括 Apache、IIS (ISAPI) 以及 AOL 服务器。

(32) 语法的点亮显示: PHP 4.0 语法的点亮显示允许开发者看见源代码而不是脚本, 这个功能比 PHP 3.0 中的更有效。它跑得更快, 执行得更好, 并且产生更紧凑的 HTML 代码。

(33) 由引用改变变量的值: PHP 4.0 由引用支持可变的赋值, “关联”的 2 个变量之

中的任何一个值被改变，另外的变量的值同样被改变，这类似与 C 中的指针类型。

(34) 在引用字符串中的变量引用：PHP 4.0 增强了在引用字符串中的变量引用。例如，这个例子在 PHP 3.0 中是不可能做到的：`print "${a[0]}->bingo";`

在以前，Web 站点大多用静态 HTML 以及少量 CGI 脚本编写，这个时代已经过去了。现在，绝大多数的网站使用动态网页编写技术设计，然而，如何选择一门技术进行编写就成为需要思考的第一步。

Perl 很好地解决了 CGI 中的问题，它被用来编写复杂的 Web 网页；微软结合自己的 IIS 服务器系统推出了 ASP (Active Server Pages) 语言等等。在 ServerWatch.com 中列举了上百种 Web 技术，有些需要花费几十万元甚至更多，我们却从其中选择了 PHP。那么，为什么选择它呢？

其实理由很简单：因为 PHP 比它们都好。上一节已经列举了一些 PHP 的特点，它是免费的，编写代码较快，执行速度也快，而在不改变代码的前提下，可以在不同的操作系统下运行。

下表是 PHP 与一些脚本语言的比较：

表 1.1 脚本语言的比较

开 发 程 序	PHP	ASP	CGI	NSAPI	ISAPI
操作系统	均可	Win32	均可	均可	Win32
Web 服务器	数种	IIS	均可	Netscape Serve	IIS
执行效率	快	快	慢	极快	极快
稳定性	佳	中等	最高	差	差
开发时间	短	短	中等	长	长
修改时间	短	短	中等	长	长
程序语言	PHP	VB	不限	C/C++	C/Delphi
网页结合	佳	佳	差	差	差
学习门槛	低	低	高	极高	高
函数支持	多	少	不定	中等	少
系统安全	佳	差	最佳	佳	尚可
使用站点	超多	多	多	极少	少
改版速度	快	慢	无	慢	慢

PHP 可在多种 Web 服务器上使用；传统 CGI 就不限于哪种操作系统或 Web 服务器平台；NSAPI 一定要在 Netscape 的服务器（如 Netscape Enterprise Server 或 FastTrack Server）上才可以执行，但可支持多种操作系统（UNIX 或 Win32）；ASP 及 ISAPI 只在 IIS 上有完整的功能。

在稳定性上，由于 NSAPI 或 ISAPI 是动态连接的方式，因此执行时如果出现问题，会使 Web 服务器一起瘫痪。而 ASP 在实际应用经验中，使用一段时间（例如超过 48 小时）就会使系统不稳定，需要重新启动操作系统。PHP 在许多的站点使用中，不但长期使用都没有问题，而且程序的稳定性也不错。当然最稳定的还是传统的 CGI 程序，因为它由操作系统负责控制，不会因 CGI 程序的错误导致 Web 服务器的不稳定。

在开发及维护时间上，PHP 及 ASP 都有不错的表现。NSAPI 及 ISAPI 需要长时间的开发过程，在稳定工作后这两种却是效率最佳的方法。传统的 CGI 程序要由开发工具语言而定，用 Perl 或是 shell script 不需要编译的过程，直接就可以执行。用 Delphi 或 VC/BCB 甚至用组合语言等都要经过编译才能执行，至于用 VB 来写传统 CGI，其效果很不理想。

要比较和网页结合的能力，PHP 和 ASP 是并驾齐驱的，其他方式就不能内嵌 HTML 语法了。而这也是影响开发时间的因素之一。

然而，就系统安全性而言，ASP 是最差的，在没有经过微软的 IIS Service Pack 升级过的服务器上，使用 `::$DATA` 或者 `:%80` 就可以看到 ASP 的原始代码。当然，传统 CGI 的程序，由于是由操作系统直接管理，要破解的难度最高。

在新增功能及改版方面，传统的 CGI 由于不受任何语言限制，没有这方面的问题。PHP 是最有活力的，数天至数周就有一个新版本出现，每次的新版，就代表更多的功能及修正更多的错误。其他的 ASP、NSAPI、ISAPI 就由具体的 Web 服务器厂商和公司来决定了。

1.3 如何学习 PHP

本节简单介绍学习 PHP 的方法，并给出适当的建议。

学习一门新的编程语言不能只是通读一遍书本，就可以立刻用上所学的知识。阅读和思考仅仅是开始。实践是学习编程语言的最有效途径。读者可以尝试着照搬现成的代码。如果运行起来了，再做改动，逐步了解其中每一句的作用。循序渐进，你就能掌握它。

那么，在最开始，需要的是一个支持 PHP 的 Web 服务器系统以便能够运行 PHP 脚本。

网络上提供 Web 个人主页空间的站点已经很多了，其中也不乏可以找到支持 PHP 的站点空间。最直接的，就是去申请一个，然后在那里学习。当然，这并不是一个最好的办法，因为使用 Internet 上边的资源，我们还需要了解有关 TCP/IP 协议、FTP 协议以及一些服务器远程登录 (Telnet) 的相关知识。对于初学者来说，在学习 PHP 以前就已经为学会这些知识使用掉了不少时间。

那么，第二个方法是本书所推荐的。首先，要有一台可以自己使用的个人电脑。然后，根据这台电脑上所运行的操作系统选择一个 Web 服务器和 PHP 脚本解释语言。当这些都安装好后，读者就可以开始真正的学习并使用 PHP 语言了。

需要特别注意的是，如何安装一个包含 PHP 脚本支持的 Web 服务器系统这一门技术也是必要的。甚至它可以被包含到 PHP 中作为一门基本知识。是否能够快速、熟练地架设一个包含 PHP 脚本支持的 Web 服务器系统是至关重要的。

1.4 本章小结

1.4.1 主要内容

本章介绍了 PHP 的概念，使用一个简单的程序来示范 PHP 程序的编写；对 PHP 的特点、功能、应用范围做了讲解，并把 PHP 和其他服务器端的脚本语言进行了比较。最后，对读者学习 PHP 的方法提出了适当的建议。

1.4.2 本章复习题

- (1) PHP 有什么特点?
- (2) PHP 和其他的服务器端脚本语言相比, 有什么优势?
- (3) PHP 支持哪几种常用数据库?