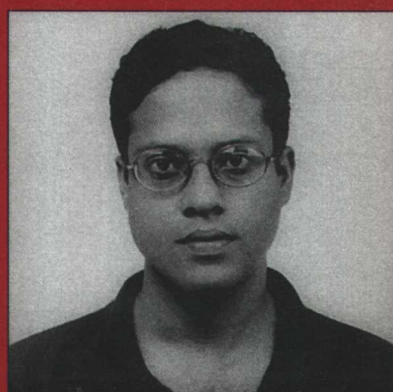
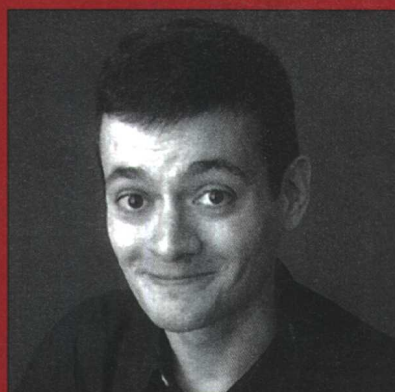
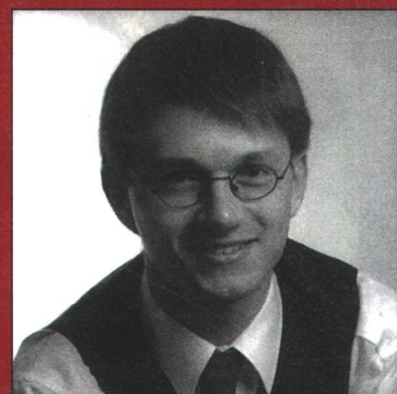
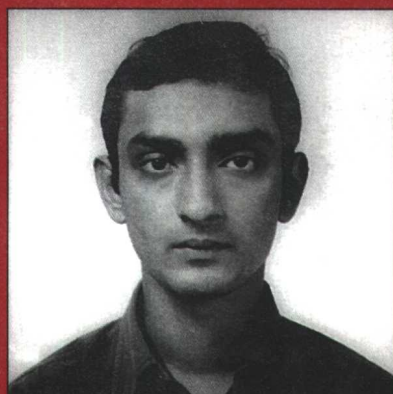
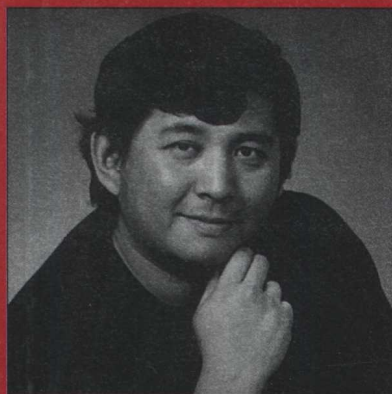




PROGRAMMER TO PROGRAMMER™

Professional PHP Programming

Wrox 程序员参考系列



PHP

高级编程

(美) Jesus Castagnetto 等著

薛忠胜 石征 杨登峰 郑丰 等译

p2p.wrox.com 提供在线支持



机械工业出版社
China Machine Press



乐思

本书介绍PHP的基本知识与高级特性，指导读者从入门到精通掌握PHP技术。内容包括PHP的变量、函数、语句、表达式，PHP对文件的处理，PHP的模块，如MySQL、XML、LDAP、GD，用PHP发送电子邮件，PHP的安全特性等。本书还介绍了五个范例应用程序，结合实际应用程序来讲述编写PHP代码的特点。本书的附录还介绍了关于PHP的问题解答、技巧和文章。本书的作译者均具有丰富的实际应用经验，赋予了本书极有价值的参考信息。

Jesus Castagnetto et al: Professional PHP Programming.

Authorized translation from the English language edition published by Wrox Press.

Original copyright © 1999 by Wrox Press. All rights reserved.

Chinese simplified language edition published by China Machine Press.

本书中文简体字版由英国乐思出版公司授权机械工业出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2000-1163

图书在版编目（CIP）数据

PHP高级编程/（美）卡斯塔格尼托（Castagnetto, J.）等著；薛忠胜等译.-北京：机械工业出版社，2001.3

（Wrox程序员参考系列）

书名原文：Professional PHP Programming

ISBN 7-111-08618-x

I. P… II. ①卡… ②薛… III. PHP语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字（2001）第00963号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：徐燕红

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2001年3月第1版第1次印刷

787mm×1092mm 1/16·51.75印张

印数：0 001-5 000册

定价：78.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

译者序

当Rasmus Lerdorf在1994年用一些Perl脚本来跟踪那些看他的简历的人时，他也许不会想到会造就今天PHP的辉煌。根据英国Net Craft公司（<http://www.netcraft.com/Survey/>）的调查，到2000年9月份，PHP在全球3 602 795个域名、681 637个IP地址上得到了使用。E-Soft公司的调查数据则说明PHP是最受欢迎的Apache模块。应该知道Apache占有全球Web服务器60%以上的市场份额。Apache+PHP+MySQL已经成为在Linux平台上建立动态网站的首选组合；事实上，它们不仅可以用在Linux平台上，还可以用在其他的类UNIX系统甚至Windows平台上。现在，除了Apache之外，PHP还支持AOL公司的AOLserver、微软公司的PWS和IIS、O'Reilly公司的WebSite、Roxen公司的Roxen、Netscape公司的Netscape Enterprise Server等。总的来说，PHP具有如下特点：

- 1) 是开放源代码软件。
- 2) 具有很强的跨平台特性，而无需修改代码。
- 3) 简单易学，任何有C或Java语言基础的人都能很快学会PHP。
- 4) 基于模块化设计，扩展性强。
- 5) 支持十几种数据库，并且都是直接的驱动程序，效率很高。

6) 支持动态图像的生成和处理、数据加密、日历、XML、电子邮件（POP和IMAP协议）、电子现金支付、信用卡验证、COM、CORBA、PDF文件、Java（Servlet）代码的调用、动态创建Shockware(Flash)文件等等。

更重要的是新的功能还在不断地增加。

全书可以分为四个部分。第一部分是关于PHP的基础知识，包括PHP的安装，PHP中的变量、函数、语句、表达式等；第二部分讲述PHP的高级内容，例如对文件的处理，对HTML表单的处理；以及对PHP的一些有名的、比较突出的模块的介绍，如MySQL（数据库）、XML、LDAP（目录访问协议）、GD（图像处理），用PHP发送电子邮件；还讲了如何提高PHP和Web服务器的安全性；第三部分是范例部分，结合实际程序来讲述编写PHP代码的特点。共有5个范例：购物车程序、论坛程序、电子邮件程序、数据库浏览器、Session应用程序。研究这些范例，可以很好地消化前面的基本知识，并提高自己的编程水平。最后一部分是附录部分，主要是PHP的函数列表，一些有代表性的问题解答，技巧和文章，以及关于PHP的网络资源。

本书的英文原版由美国Wrox(<http://www.wrox.com>)出版社出版，在国外是最受欢迎的PHP专业书籍。译者在翻译这本书的时候，发现有些东西与现实不相符了，PHP已经有了不小的发展。本着对读者负责的原则，在征得出版社的同意后，译者对原书的部分内容做了修改（例如安装部分），并增加了一些内容（例如PHP 4新增的foreach语句、Session等），但总的来说，译文还是忠于原书的。

本书由薛忠胜（网名Hunte，PHP中文用户网站站长）组织翻译，石征、杨登峰、郑丰等参

与了翻译工作。本书所涉及的内容比较广泛，译者的水平有限，疏漏之处在所难免，还望广大读者不吝指教，更欢迎大家就本书或者PHP脚本语言，到PHP中文用户网站（<http://phpuser.com>）上畅所欲言。

2000年10月

作者介绍

JesusCastagnetto

JesusCastagnetto是一个化学博士，在Scripps Research Institute工作。该研究院是属于金属蛋白质结构和设计集团(<http://www.scripps.edu/research/metallo/>)的博士后研究团体。在这里，他们用自己内部的程序（Java Applet/Servlet）、SQL数据库和PHP3开发金属蛋白质的数据库和浏览器(<http://metallo.scripps.edu/0>)。由于在计算、合成、医药化学等方面的研究，他从纽约大学获得了博士学位。他现在研究的课题是：金属蛋白质。生物信息和化学品的计算模式是他最感兴趣的两个领域。另外，他还精通awk、PHP。

Harish Rawat

Harish Rawat是Oracle公司的一位软件工程师，在系统编程方面有七年的工作经验。他感兴趣的技术领域包括XML、Java和网络协议。

Sascha Schumann

Sascha Schumann是PHP开发小组和Apache软件基金会的一名成员。他目前正在Gymnasium Letmathe学习。Sascha是许多PHP模块（如DBA、Session管理、mcrypt和mhash）的设计师，他使PHP能够用在AOLserver上，并给使用PHP的组织提供综合的支持和模块的定制。Sascha还维护mhash——一个用于加密的开放源代码软件。

Christopher Scollo

Christopher Scollo是Taurix公司的合作创始人和现任总裁。Taurix公司是一家软件开发公司，位于新泽西州的New Brunswick。在他不开发基于Web的应用程序时，他教授Web方面的课程，给Web期刊撰稿。除了计算机编程语言，他对人类的语言也很有兴趣。他在不同程度上掌握英语、德语、法语和意大利语。

Deepak Veliath

Deepak Veliath目前就职于印度Bangalore的Oracle公司。他学习PHP已经有一年了，最喜欢的操作系统是Linux。

前言

在过去的几年里，因特网(Internet)已经从学术界向商业界迈进。这主要是由于Web的快速发展，包括Web上的图形浏览器和多媒体的集成。由于脚本技术的引入，从静态网页到动态网页，用户交互操作已经取得很大的改变。

和网页上普通的描述语言一样，脚本语言可让客户端从服务器端取得特定的信息，同样服务器端可以接收用户的输入，处理和显示用户要求的数据。

在技术最前沿的是开放源代码群体，他们把解决基于Web的问题仅仅看作是对技术的纯粹的爱好。开放源代码软件不仅仅是自由软件（虽然不是所有的都是免费的），而且从它的名字就可以看出，开放源代码就是公开程序的源代码。

开放源代码意味着程序的作者必须按照一定的规则来做事。如果程序缺乏维护，就可能会有问题，按规则来做事的优点是，如果作者没有修正，别的人也可修正。很多独立的程序员懂得程序能做什么，并能确保程序遵循一定的准则，防止程序作者控制软件的用户，而这在商业软件中是经常发生的。

本书覆盖的内容

PHP是本书的主题，它是一种服务器端的开放源代码的脚本语言，它把Web开发带到了一个新的时代。

在这本书里，我们将向你介绍什么是PHP，它是怎样简化服务器端的脚本编程的，以及它是怎样给你的Web页面增加新的功能的。

具体来说，我们将讲述以下几方面的内容：

- PHP的语言结构和语法，包括数据类型、操作符、表达式和函数。
- 在服务器上安装PHP脚本引擎的过程。
- 可大大提高PHP性能的扩展模块。
- 服务器端的编程。
- PHP的案例研究。

本书读者对象

这本书是为那些要实现从简单的静态网页到基于Web应用程序的人写的。如果你正准备建立一个数据库驱动的网站，或者你不了解整个动态网页产生的过程，或仅仅是做一些好看的服务器端脚本，请读本书。如果你是富有经验的开发者，直到最近还只能用它在Kewl-OS操作系统上产生简单输出，但觉得有必要构造更轻便、能够在短时间内开发出来的解决方案，也可参考本书。

本书假定读者有Web技术的一些基本知识，可以轻松写出HTML，有用脚本语言或编程语言写程序的经验，懂得基于Web的信息系统的工作原理。也假定读者对数据库有概念上的了解，当

然，对这些数据库系统了解得越深，就越容易理解本书所用到的一些例子（特别是理解数据库和购物车应用的那些章节中的例子）。

环境需求

要做服务器端的编程，你的电脑上需要一个Web服务器。它可以是Windows平台上的IIS，或者其他操作系统上的服务器，像Apache、Xitami。Apache是目前用得最多的Web服务器，占有60%的市场份额。

要用服务器端的脚本工具，你需要在服务器上安装和配置PHP语言解释器。我们会在本书的前面一部分介绍。在客户端，你可以自由选择用IE浏览器或者Netscape Communicator，也可以是其他的Web浏览器，如Opera等。

如果你想运行全部的代码，你应该有一个可用的关系型数据库。这方面可选择的范围很广，这将在本书的后面介绍。本书选用MySQL关系数据库。

对编程本身，你所需要的是一个好的文本编辑器，像Windows的写字板、gnotepad或Emacs（译者注：推荐使用EditPlus 2，它可以在<http://www.editplus.com>上得到。另外，在<http://phpuser.com/msic/phpeditor.php>可以看到更多的关于编辑器的信息）。脚本都是简单的文本文件，通常嵌入在生成网页的描述语言里。

请把你的想法告诉我们

我们尽力把书编得更有用。使书更符合你的期望和要求。

请告诉我们你对本书的看法，告诉我们本书的哪些内容写得好，哪些写得不好。这不是商业上的花言巧语，我们的确会在众多的电子邮件中找出你的想法。如果你不相信，那么给我们留言，我们会答复而且在书的下一版中参考你的意见。最方便的途径是用电子邮件：feedback@wrox.com。你也可以在我们的网站上了解关于Wrox出版社更多的细节。你可以看到我们最新的书里面的代码，可以预览即将出版的书目，和一些有关著者编者的信息。

客户支持

如果你在本书中发现了错误，请先看我们网站上关于这本书的勘误表。如果你找不到解答，告诉我们问题的位置，我们会尽快尽我们所能来解决。

邮件地址：

support@wrox.com

网站：

<http://www.wrox.com/contacts.asp>。

目 录

译者序		第3章 在Web环境下编程	42
作者介绍		3.1 为什么用PHP	42
前言		3.2 Web方面的技术	43
第1章 什么是PHP	1	3.2.1 静态发布	43
1.1 PHP简介	1	3.2.2 动态站点	43
1.1.1 PHP的发展历程	2	3.3 编写PHP页面	47
1.1.2 PHP的现状	3	3.3.1 PHP和客户端代码	49
1.2 本书的风格和编排	4	3.3.2 PHP变量	49
1.3 小结	4	3.3.3 与用户交互	50
1.4 一些有用的网站	4	3.3.4 处理表单数据	52
第2章 安装和配置PHP	6	3.4 一个工作申请表	53
2.1 安装	6	3.4.1 URL编码的介绍	54
2.1.1 平台和Web服务器	6	3.4.2 注释代码	58
2.1.2 PHP的模块	6	3.4.3 字符转义	59
2.1.3 CGI解释器和Apache模块	7	3.5 小结	61
2.1.4 获得PHP	8	第4章 变量、常量和数据类型	63
2.1.5 在UNIX下建立和安装PHP	10	4.1 数据类型	63
2.1.6 测试和基准测试	15	4.2 数据的值	63
2.1.7 在Windows下安装PHP	16	4.3 常量	63
2.1.8 在Microsoft PWS下安装PHP	19	4.3.1 定义常量	64
2.1.9 在OmniHTTPd下安装PHP	20	4.3.2 内置常量	64
2.1.10 在Microsoft IIS 5.0下安装PHP	22	4.4 变量的声明和初始化	65
2.2 配置	23	4.5 类型转变和类型强制	65
2.2.1 在编译时进行配置	24	4.5.1 字符转变和类型转变	66
2.2.2 用Apache命令进行配置	26	4.5.2 类型强制	66
2.2.3 利用php.ini文件进行配置	26	4.6 变量的变量	66
2.3 MySQL	35	4.7 关于变量的一些有用函数	67
2.3.1 安装MySQL	36	4.8 建立一个在线工作申请表	69
2.3.2 配置系统的访问权限	39	4.9 小结	72
2.4 Boutell的GD库: 支持创建GIF图片	40	第5章 操作符	73
2.5 Expat: XML分析支持	41	5.1 算术运算符	73
2.6 小结	41	5.2 赋值符号	73
2.7 建议读物	41	5.3 比较符号	73

5.4 逻辑操作符	75	8.3 多维数组	118
5.5 字符串连接符号	76	8.4 排序函数	119
5.6 三元操作符	77	8.5 在表单元素中使用数组	122
5.7 位运算符	77	8.6 小结	123
5.8 变量赋值的简洁方式	78	第9章 面向对象的编程	125
5.9 其他类型操作符	79	9.1 什么是面向对象的编程	125
5.9.1 变量操作符	79	9.1.1 分类	126
5.9.2 对象操作符	79	9.1.2 实例化	130
5.9.3 错误抑制符号	79	9.2 继承	132
5.10 操作符的优先级和组合	80	9.3 小结	135
5.11 建立一个在线的工作申请表	81	第10章 字符串处理和正规表达式	136
5.12 小结	83	10.1 基本的字符串函数	136
第6章 语句	84	10.2 正规表达式	139
6.1 条件语句	84	10.2.1 基本模式匹配	140
6.1.1 if语句	84	10.2.2 正规表达式函数	144
6.1.2 分支条件语句	85	10.3 建立在线工作申请表	147
6.1.3 switch语句	86	10.4 小结	154
6.2 循环	88	第11章 文件处理与数据存储	155
6.2.1 while循环	88	11.1 文件处理	155
6.2.2 do...while循环	89	11.1.1 打开文件	155
6.2.3 for循环	90	11.1.2 关闭文件	156
6.2.4 foreach语句	91	11.1.3 显示文件内容	156
6.3 在PHP脚本中包含文件	91	11.1.4 从文件中读取内容	156
6.4 从PHP脚本中退出	92	11.1.5 把数据写入文件	157
6.5 建立一个在线工作申请表	92	11.1.6 文件的遍历	158
6.6 小结	96	11.1.7 拷贝、删除和重命名文件	159
第7章 函数	97	11.1.8 决定文件的属性	159
7.1 函数如何工作	97	11.1.9 目录的处理	160
7.2 变量的作用范围和生命周期	101	11.2 从客户端上载文件	162
7.3 函数的嵌套和递归	102	11.2.1 处理上载文件	162
7.4 把函数赋给一个变量	104	11.2.2 确定文件有效	163
7.5 建立一个在线工作申请表	104	11.3 完善工作申请表	164
7.6 小结	109	11.4 非关系型数据库	170
第8章 数组	110	11.4.1 程序说明	170
8.1 简单数组	110	11.4.2 地址簿的用户界面	171
8.1.1 初始化数组	110	11.4.3 设计总体查看的表单	171
8.1.2 数组的遍历	111	11.4.4 数据库查询	174
8.2 字符串索引的数组	115	11.4.5 删除条目	175

11.4.6 显示条目	176	13.4.4 修改函数	245
11.4.7 编辑地址簿	177	13.4.5 PHP的LDAP应用程序	246
11.4.8 更新数据库	178	13.5 小结	258
11.4.9 增加新条目	178	第14章 XML	260
11.4.10 从CSV文件输入数据	180	14.1 什么是XML	260
11.4.11 综合起来	181	14.2 XML语言	262
11.5 小结	183	14.2.1 XML文档的结构	263
第12章 PHP与SQL数据库	184	14.2.2 文档类型定义	264
12.1 为何要使用数据库	184	14.3 PHP对XML的支持	268
12.2 结构化查询语言	186	14.4 PHP的XML函数	268
12.2.1 PHP写的SQL代码测试器	186	14.4.1 生成XML解析器	270
12.2.2 SQL语言参考	190	14.4.2 注册回调函数	271
12.3 PHP对数据库连接的支持	198	14.4.3 解析XML文档	277
12.3.1 MySQL数据库	198	14.4.4 释放解析器	278
12.3.2 PHP中对MySQL支持的函数	200	14.4.5 解析器选项	278
12.4 一个PHP和MySQL应用程序的例子	215	14.4.6 实用函数	278
12.5 小结	227	14.5 PHP的XML应用程序	280
第13章 PHP与LDAP	228	14.5.1 支持Web的企业级应用程序	280
13.1 目录服务与LDAP	228	14.5.2 聪明的检索	280
13.1.1 LDAP特性	229	14.5.3 转换XML到HTML	280
13.1.2 LDAP与大规模数据库	230	14.5.4 相同数据的不同外观	281
13.2 LDAP的工作	230	14.6 PHP XML应用程序示例	281
13.2.1 LDAP组件	231	14.7 小结	292
13.2.2 LDAP的数据组织	231	第15章 图像生成和处理	294
13.2.3 LDAP说明	232	15.1 图像和服务端脚本	294
13.2.4 目录与数据库	233	15.2 PHP中的图像处理	297
13.2.5 LDAP的操作	233	15.2.1 安装具有图像支持的PHP	297
13.2.6 LDAP的方言——LDIF	234	15.2.2 PHP的图像函数	298
13.2.7 LDAP的一些高级特性	236	15.3 使用图像函数的应用程序	319
13.3 LDAP软件	237	15.3.1 使用PHP的示例图像应用	319
13.3.1 LDAP软件的选择	237	15.3.2 显示计数器的页面	320
13.3.2 安装与配置LDAP服务器	238	15.3.3 记录点击次数	320
13.3.3 运行slapd服务器	239	15.3.4 生成图像	322
13.4 PHP对LDAP的支持	241	15.4 小结	323
13.4.1 PHP提供的LDAP API	241	第16章 网络连接	324
13.4.2 连接函数	242	16.1 客户和服务端	324
13.4.3 查询函数	242	16.2 使用TCP	324

16.3 whois 函数	325	19.3 错误处理	366
16.4 前台的界面	327	19.3.1 抑制错误信息	367
16.5 扩展前台界面	328	19.3.2 错误之后的完美恢复	367
16.6 我的域名仍然可用吗	329	19.3.3 错误检查的定制	368
16.7 Web客户端	330	19.4 记录错误	368
16.8 小结	334	19.5 一些常见的错误	373
第17章 发送和接收电子邮件	335	19.5.1 已发送头信息	373
17.1 发送电子邮件	335	19.5.2 未定义函数	373
17.1.1 mail() 命令	335	19.5.3 字符转义错误	374
17.1.2 mime_mail 类	336	19.5.4 最大执行时间	375
17.1.3 使用SMTP	340	19.6 小结	375
17.2 接收电子邮件	344	第20章 安全问题	377
17.3 小结	347	20.1 安全的重要性	377
第18章 cookie	349	20.2 从头开始	377
18.1 什么是cookie	349	20.3 保护Web服务器	378
18.2 cookie的限制	349	20.4 使PHP更为安全	381
18.3 PHP中的cookie	350	20.5 使用安全模式	383
18.3.1 开始	350	20.6 用户识别和验证	383
18.3.2 你的名字是什么	352	20.7 使用加密技术	386
18.4 存取一个cookie	352	20.8 使用SSL	388
18.5 设置一个cookie	353	20.8.1 下载OpenSSL和ModSSL	389
18.5.1 多值cookie	353	20.8.2 编译和安装OpenSSL库	389
18.5.2 设置失效日期	353	20.8.3 编译和安装mod_ssl	392
18.5.3 限制cookie的范围	354	20.8.4 SSL基本配置	394
18.6 删除一个cookie	355	20.8.5 安装专用密钥	395
18.7 一般未知危险	355	20.8.6 创建证书请求和临时证书	396
18.8 发送HTTP头	356	20.8.7 取得签名证书	397
18.9 小结	357	20.9 使用Apache的suEXEC机制	397
第19章 调试及错误处理	358	20.10 创建安全的PHP脚本	398
19.1 错误的类型	358	20.11 小结	401
19.1.1 语法错误	358	第21章 字符的自动转义	402
19.1.2 语义错误	360	21.1 问题所在	402
19.1.3 逻辑错误	361	21.1.1 转换过程	402
19.1.4 环境错误	362	21.1.2 定义使用范围	403
19.2 PHP的出错信息	362	21.2 使用自动转义	403
19.2.1 PHP的错误级别	363	21.3 两个有用的函数	405
19.2.2 设置错误报告的级别	366	21.4 可选方式	407

21.5 小结	407	24.3.10 搜索	499
第22章 模板	408	24.4 小结	501
22.1 通用的模板技术	408	第25章 案例3——电子邮件程序	503
22.2 在应用程序中使用模板	408	25.1 程序功能	503
22.2.1 内嵌PHP	408	25.2 程序实现	503
22.2.2 分离通用部分的代码	409	25.3 小结	512
22.2.3 快速模板	410	第26章 案例4——数据库浏览器	513
22.2.4 网站框架	415	26.1 设计应用程序	514
22.3 现有代码与快速模板	417	26.2 数据库图解	515
22.4 小结	417	26.2.1 Protein表	515
第23章 案例1——购物车应用程序	418	26.2.2 位置表	516
23.1 需求分析	418	26.3 脚本的整体设计	517
23.2 选择应用程序所需的软件	419	26.4 简单SQL查询脚本	517
23.3 最终用户和购物车应用程序的交互	420	26.5 包含外部函数模块	520
23.4 设计购物车应用程序	421	26.5.1 生成SQL语句	520
23.4.1 数据库设计	422	26.5.2 数据中间层脚本	523
23.4.2 中间层的设计	425	26.5.3 显示结果	525
23.5 最终用户特性的实现	426	26.6 查询标识字段	527
23.6 应用程序代码	432	26.7 扩展查询界面	532
23.7 管理界面的实现	457	26.8 一些想法	541
23.8 小结	475	26.9 小结	542
第24章 案例2——Phorum论坛程序	476	第27章 案例5——Session应用程序	543
24.1 为什么使用PHP	476	27.1 关于Session	543
24.1.1 数据库支持	476	27.2 在PHP3中实现Session	544
24.1.2 在HTML中内嵌PHP	477	27.2.1 安装和初始化	544
24.1.3 异常错误的处理	479	27.2.2 基本使用方法	545
24.2 Phorum怎样运行	479	27.2.3 实地演习	546
24.3 代码研究	482	27.3 PHP4中的Session	547
24.3.1 代码重用	482	27.3.1 从这里开始	547
24.3.2 数据库的抽象层	482	27.3.2 与Session相关的php.ini配置	549
24.3.3 动态生成PHP代码	483	27.3.3 PHP4中的Session函数	550
24.3.4 在cookie中存放用户数据	484	27.4 Session的定制	552
24.3.5 核心文件forums.inf	485	27.5 小结	558
24.3.6 论坛列表	488	附录A PHP函数	559
24.3.7 列出消息	489	附录B 关于PHP的10个常见问题	759
24.3.8 阅读消息	493	附录C 关于PHP的10个技巧	765
24.3.9 张贴消息	494	附录D 关于PHP的10篇文章	770

第1章 什么是PHP

互联网在许多方面的发展是非常迅速的。很多物理学家用我们都不熟悉的，名叫HTML的标记语言在CERN服务器上连接学术文档的事好象就发生在昨天。但在今天，用简单的程序就可以在世界的任一角落阅读这些文章，而且文章中的链接可以方便地把你带到另一个相关的网站上。

这些很快被应用到其他的科学领域。文本界面是标准，可以方便获取信息的内容。要和同事们共享你的知识，你所要做的事就是准备有一套标记符的文本文档和简单的服务器配置，而不管在另一头的化学家用的是他可信赖的MAC机，或者理论学家在用他的UNIX机器，或者还是贫穷的研究生用的运行TCP/IP协议软件的二手、三手的PC。

今天我们希望的近非如此，我们希望网站有更多的信息，更好的界面，但我们不希望被不同的界面分散注意力。信息应该容易找到，而且是最新的，整洁的动态网站对使用者和信息提供者都是有用的。幸运的是那些在页面上放置过份装饰的、闪烁的内容或者大量动画的网站已经远去了。我们要的是信息，我们在五分钟前需要它，并且以我们喜欢的方式得到它。

现在的网站不仅仅是一个Web服务器那么简单了，它还包括怎样储存数据，查询数据（后台可能是SQL数据库），怎样处理用户的请求，怎样创建包含正确信息的文档。对于网站开发者有很多选择，但不是所有的选择都是合适的。我们不仅要考虑手头正在创建的动态网页的任务，我们还要确保不会因为软件或硬件的改变而停止提供预定的内容。

我们尽量面对将来科技的变化，尽量减少许可证费用，控制我们的硬件预算不超支，而且在不同的服务器和操作系统之间保持可移植性。我们也要确保能消除一些在我们服务器或脚本环境下发现的程序错误，理解脚本的工作原理，并且能够为不同的特殊需要修改我们的服务器和脚本。开放源代码软件可以使你的应用程序现在运行在Super-Turbo Hexium IX机器上，将来会运行在Nanotech Cube Aleph上。

1.1 PHP简介

PHP(PHP Hypertext Preprocessor的首字母缩写)是一种在服务器端执行的脚本语言。这意味着它可以很好地和HTML融合，产生需要的文档。你可以把你的网站转为Web应用程序，而不仅仅是一组难以经常更新的静态页面。静态网页对于个人网页可能是合适的（是的，我们也做过个人网站），但对于那些要用于商业或教育的网站就不合适了。

你可能会问，为什么要用PHP，还有很多选择，像ASP、Cold Fusion、Perl、Java、Python、甚至更好的老式脚本语言，如shell、awk、sed等，答案是：由于它的简单性，用近乎自然的方式使用数据库，保持系统平台的无关性。

我提到它是开放源代码的吗？

当然，大多脚本语言或编程语言，像Perl、Python等，都是与平台无关，且是开放源代码的。它们是很好的编程语言，但有时它对你的需求来说太强大了，有“杀鸡要用牛刀”之嫌。PHP

被设计成专门用于Web。它在这个领域非常优秀，通常完成简单的数据库连接和查询只要2行或3行代码。PHP引擎对Web应用程序的响应时间作了优化，甚至成为Web服务器本身的一部分，这大大提高了Web的性能（吞吐量）。

如果仅仅是要提高脚本的执行速度，那么PHP是不错的选择。但PHP的功能远非如此。它简单易学，是一种强壮有力的脚本语言。它是联系日益增长的数据库服务器系统，更短的开发周期，更容易地创建模块（归功于PHP语法和结构）及可重用组件的纽带。

使用PHP可以执行很多任务，从简单地创建反馈表单，发送电子邮件给网站管理员，到全部由数据库驱动的文档管理系统（如Midgard，<http://www.midgard-project.org/>），到帮助系统或程序排错系统（如Keystone，<http://www.stonekeep.com/keystone.php3>），到购物车应用程序（如FishCartSQL，<http://www.fni.com/fcsql/>），到可称为“中间件”的不需要额外语言或框架支持的程序包，再到能够在开发中灵活、快速使用的类库（PHPLIB，<http://phplib.netuse.de/>）。

还有许多来自广泛分布的交互合作的社区的支持，包括有一些源代码仓储（像PHP代码交换，<http://px.sklar.com>；Berber的WeberDev，<http://www.weberdev.com>）；许多网站还有教程（如PHPBuilder，<http://www.phpbuilder.com>；PHPWizard，<http://www.phpwizard.net>）和大量丰富的邮件列表资源。

我提到过它是开放源代码软件吗？

不必等下个版本加进新特性或修正错误，你所要做的是拿到源代码，按自己的要求修改定制、安装，完全地控制它。不要怀疑特别的函数特性不可靠，代码是不会撒谎的。也许可能有一天你所做的修改变得很流行，其他的人也要用到你修改的版本（你立即出名了）。你不可能再找到开发的总费用比Linux+Apache+MySQL+PHP的组合更便宜的环境了。它不仅比其他平台便宜，而且更稳定、更健壮。就像Eric Raymond所说的：“只要给足够的注意力，所有的错误都将暴露无遗”。

1.1.1 PHP的发展历程

PHP是Rasmus Lerdorf（可以在<http://www.zend.com/zend/hof/rasmus.php>上了解更多的情况）在1994年构思出来的，他用一些Perl脚本来监视什么人在看他的在线简历。逐渐地，人们开始对他的脚本感兴趣。不久，他以“个人网页（Personal Home Page）”工具包的名字发布了PHP（PHP的第一个含义）。由于自身的兴趣爱好，他写了一个脚本引擎和用于分析网页表单输入的工具：FI（Form Interpreter），即表单解释器。1995年年中，Ramus重写了整个解释器，并取名PHP/FI或者PHP 2。

不久，人们开始使用这些工具来做更复杂的东西，PHP的开发也由Ramus一个人变成由一组核心开发人员负责的团体。这就是PHP3的开始，这个开发小组（Rasmus Lerdorf、Andi Gutmans、Zeev Suraski、Stig Bakken、Shane Daraveo和Jim Winstead）增强和扩展了PHP脚本引擎，增加了一些简单的API函数，并允许其他程序员自由地编写PHP模块来增强PHP的功能。语言的语法也被重新定义，使那些使用面向对象或过程语言的人不感到陌生。如果你懂得C、C++或Java，或者已经写过一些Shell、awk脚本，或Pascal、VB程序，那么学起PHP来将会感到很轻松。

PHP语言是由控制结构、操作符、变量类型、函数定义和类/对象声明组成的，这跟我们熟悉的编译或解释型语言很相似。当然PHP有自身的特点。例如，C语言中你可以使用指针，但在脚本语言中使用指针是很麻烦的，甚至是不可能的，但在PHP中，这仅仅是用变量的变量（在本书后将详细介绍），如下所示：

```
$peru = array("domain"=>"pe", "capital"=>"Lima");
$japan = array("domain"=>"jp", "capital"=>"Tokyo");

function show ( $country ) {
global $$country;
    echo "Internet domain = ".${$country}["domain"]."\n";
    echo "Capital city = ".${$country}["capital"]."\n";
}

show ("peru");

// Prints:
// Internet domain = pe
// Capital city = Lima

show ("japan");

// Prints:
// Internet domain = jp
// Capital city = Tokyo
```

技巧是在\${\$country}[]中的调用，它和用\$peru[]或\$japan[]等价。但也不全是，如果你想写一个函数，使用另一个函数来比较一对元素。和上面例子相似的，是我们可以主函数参数里传送不同类型的参数。

```
function bigger($x, $y, $comp_func) {
    if ( $comp_func($x, $y) ) {
        $out = "Item ".$x." is bigger than ".$y;
    } else {
        $out = "Item ".$x." is not bigger than ".$y;
    }
    return $out;
}

bigger(2, 1, "num_comp");
// will use the function num_comp() to compare the numbers

bigger("epsilon", "gamma", "greek_comp");
// will compare the strings as names of greek alphabet characters
```

这样写处理程序很简单。上面的代码里没有排印错误，用\$comp_func而不用\$\$comp_func。想一想你就会明白为什么了。

如果暂时不明白上面的例子，不要太担心。上面的例子仅仅是向你展示一下本书中将要涉及的内容，而且将会很详尽地解释。

1.1.2 PHP的现状

PHP4是基于Zend引擎的(详见<http://www.zend.com>)。Zend引擎从一开始就设计成很容易和

不同的应用程序相结合。PHP是第一个应用Zend引擎的应用程序，但其他程序包也可以使用Zend引擎，如在MySQL里（该软件很好地应用了存储过程）。

PHP4的正式版已出来了，我建议读者访问Zend网站以获得更多信息，特别引人注目的是用Zend引擎有可能可以使用COM甚至COBRA（对COM的支持已经实现了）。

非常明显，越来越多的网站开始用PHP来实现自己的需要。统计数据（可以在PHP官方站点的访问统计页面<http://www.php.net/usage.php>和Netcraft免费网站<http://www.netcraft.com>上找到）表明，越来越多的网站把PHP作为Apache的内部模块（大约1000000个虚拟主机）。也有很多以CGI方式使用PHP的网站，但这些数据没有出现在Netcraft的统计中。

值得一提的是mod_PHP(即Apache模块)是因特网上最受欢迎、网站用得最多的模块（E-Soft公司调查，其网址为<http://www.e-softinc.com/survey/>）。保守的预测是PHP的使用会稳定增长；如果在PHP4版本中有Zend引擎所讲的全部功能，用的人会更多。

1.2 本书的风格和编排

本书简明易懂而不深奥，尽量使用现实生活实际的例子，而不是抽象的例子（如，"Hello World", "\$foobar=1"）。本书的目的是为读者提供有用的代码（做尽可能少的修改），而不是复杂的技术研讨。这些例子尽可能保持浏览器的无关性，这也是开发强壮的Web应用程序的好方法。客户端不需要特定平台的浏览器，例如，PHP生产Javascript代码时，它将正常执行，而不管用户是运行Windows下的IE，还是Solaris窗口下的Netscape Navigator。

虽然，PHP的开发者还需要用到其他的一些技术，如HTML、SQL、HTTP服务器，但本书不会详细介绍上面的内容。只是在用到与PHP有关的内容时才会提到。例如，我们不会非常详尽地讨论XML的基本知识和规范，但我们会示范如何使用PHP和MySQL来开发一个基于Web的数据库应用程序。

我们把本书分成几个部分，第一部分通过简明的现实生活中例子介绍PHP语言基本知识；第二部分包括使用数据库，创建图像等复杂的PHP应用；第三部分将详尽介绍用PHP写的应用程序（如购物车程序）；最后一部分以附录的形式出现，包括PHP函数，对开放源码和相关内容的讨论。

1.3 小结

本书覆盖了PHP的核心内容，包括PHP的安装、配置方面的问题，实际应用的演示等。本书的重点是放在一般的商业需求上，如数据库应用和电子商务。最后，我们的宗旨是给你（作为一个Web应用开发者）提供一个功能强大的工具，可以在很短时间内建起一个更好的项目。我们很高兴你选择PHP来开发你的项目。甚至决定不只是考虑使用PHP，而且在实际中使用，参加到其他PHP开发者的行列中，你将会得到很多人的帮助。

1.4 一些有用的网站

这里是一些你可能在开发过程中要随时了解的关于PHP、Apache和MySQL的网站。完整的

列表在本书后面的附录中。

- PHP: <http://www.php.net/>
- Apache: <http://www.apache.org/>
- MySQL: <http://www.mysql.com/>
- PHPBuilder: <http://www.phpbuilder.com/>
- PHP中文用户: <http://phpuser.com/>