

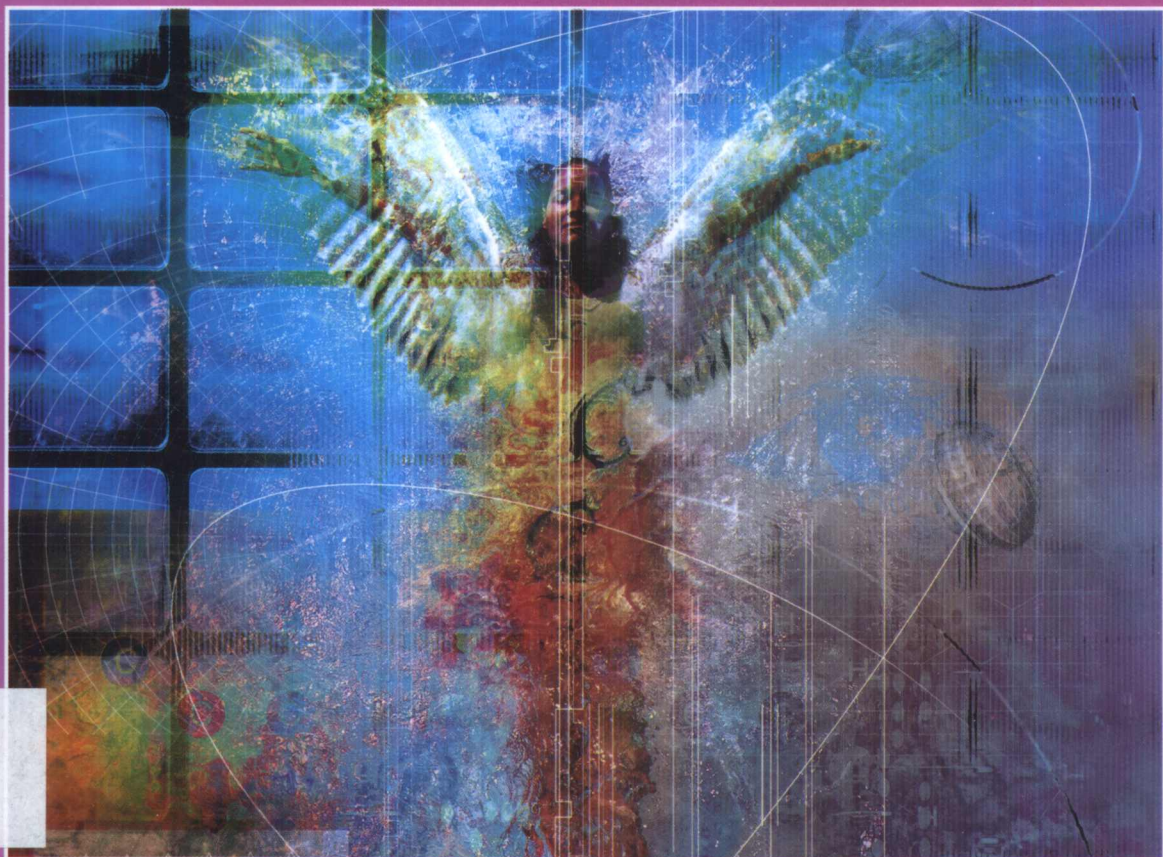
PHP

动态网站建设

高成 主编 董长虹 姚德忠 副主编
张华弟 段莉莉 马强 张乐乐 编著

NATIONAL DEFENSE INDUSTRY PRESS

国防工业出版社



图书在版编目(CIP)数据

PHP 动态网站建设/高成主编. —北京:国防工业出版社,2002.9

(动态网站建设丛书)

ISBN 7-118-02847-9

I. P... II. 高... III. PHP 语言 - 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2002)第 025919 号

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

三河市新艺印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 17 $\frac{1}{4}$ 403 千字

2002 年 9 月第 1 版 2002 年 9 月北京第 1 次印刷

印数:1—3000 册 定价:24.00 元

(本书如有印装错误,我社负责调换)

前 言

随着网络技术的迅猛发展和覆盖全球的独特优势，越来越多的人通过制作网页将自己的想象、思维和观点介绍给世界，他们使网页制作者的行列越来越壮大。毋庸置疑，很多优秀的网页制作软件也应运而生，如 DreamWeaver 等等。这些软件操作简单，制作的页面都可以达到漂亮的程度。但是，这些软件制作出的页面都存在致命的缺陷——没有与数据库连接，也称之为“静态网页”。静态网页所有内容的更新都要靠手工来完成，换言之，当页面内容发生变化时，或者用户对页面进行访问并要求获得某种服务，例如要求生成一个结果随页面变化而变化表单，它们是无能为力的。

对于大型网站来说，不具备自动更新内容的能力将是吞噬时间的噩梦，解决问题的方法就是制作动态网页。动态网页是相对于静态网页而言的，它们与数据库相关联，页面可以根据数据库中数据的变化而自动更新。目前可以应用的动态网页设计方案主要有两套：一种是采用 Microsoft 公司的操作系统和一些软件的组合，包括 Windows NT4.0/Windows 2000、IIS 以及 MS SQL，这样的组合性能优异，但费用也是相当昂贵的；另一套系统在近几年里得到了广泛的应用，因为该系统全部采用完全免费的自由软件，而且最重要的是其同样拥有强大的性能。它的操作系统采用 Linux/UNIX，服务器采用 MySQL，编程语言采用 PHP，服务器软件采用 Apache。

PHP 是英文 Personal HomePage Program 的缩写，它可以满足个人网页制作和大型网站建设的需要，本书对其基本知识及其应用进行介绍，希望对读者有所帮助。

本书的内容

本书共分为四个部分。

第一部分为入门篇，包括 1 章~6 章的内容，主要介绍了 PHP 的起源、特性、安装、使用费用和基本语法等知识，目的是通过这一部分的学习，读者将能够迅速地了解 PHP 并掌握简单的应用。

第二部分为数据库篇，包括 7 章~11 章的内容，主要介绍了 MySQL 数据库，动态网页的设计是离不开数据库的，如何实现与数据库的连接，如何向服务器发送数据，如何进行数据查询等等问题都可以在这部分找到答案，通过这一阶段的学习读者将基本能够掌握 PHP 与 MySQL 数据库的协同工作。

第三部分为会话篇，包括 12 章~15 章的内容，主要介绍了会话、客户/服务器交互技术、面向对象程序设计和 XML 基础知识等网站建立过程中的一些高级问题，通过这一部分的学习读者将对 PHP、对 HTML 乃至 HTTP 有进一步的了解。

第四部分为附录，介绍了 PHP 中的函数以及本书中的一些术语，方便读者在阅读和

使用过程中进行查询。

本书的结构

本书正文共 15 章的内容，每章以本章将讲述以下内容作为开始，将内容提要简明地呈现给读者；在详细讲解内容的同时，穿插特别提示，将提醒读者对某些问题引起注意；书中给出的示例代码和一些运行结果均有灰色背景，以示与其他文字的区别；最后以小结结束，总结一章的要点。

本书的适用对象

本书适用于那些有兴趣于动态网页设计的初学者。基于这样的目的，在内容设计上相对比较简单，只涉及到少量的高级话题，所以不适用于熟练的网页制作程序员；但对于初级的网站建设将有所裨益，如果读者希望实现更完美、性能更理想的网页，仍需要进一步的研究和提高。

学习本书需要的条件

尽管本书针对初学者安排内容，但读者至少要掌握一门高级语言的知识，并且应该对 HTTP 和 HTML 具有一定的了解，将有助于快速入门和新知识的领会。此外，最好拥有一台安装有 Linux/UNIX 操作系统的微机将方便练习和实践。

目 录



第 1 章 PHP 简介

1.1 PHP 起源与发展	2
1.2 PHP 特性	2
1.2.1 优点	3
1.2.2 缺点	3
1.3 费用问题	4
1.4 Apache 与 MySQL	4
小结	4

第 2 章 PHP 安装

2.1 组建主机	5
2.1.1 外部采办	5
2.1.2 用户自建	6
2.1.3 其他方法	6
2.2 安装 PHP	7
2.2.1 UNIX/Linux 平台	7
2.2.2 Windows NT/Windows 2000 平台	9
2.2.3 Windows 95/98/Me 平台	10
小结	11

第 3 章 PHP 的语法特点

3.1 PHP 语法的一般特点	12
3.2 数值	13
3.2.1 数字	13
3.2.2 字符串	14

3.3 常量	16
3.4 变量	16
3.5 注释	18
3.5.1 单行注释	18
3.5.2 多行注释	19
3.6 表达式	19
3.6.1 表达式的构成	19
3.6.2 优先级和结合次序	20
3.7 输出	20
3.7.1 echo 语句	20
3.7.2 print 语句	21
3.8 操作符	21
3.8.1 算术操作符	21
3.8.2 条件操作符	22
3.8.3 一元操作符	24
3.8.4 逻辑操作符	25
3.8.5 三元操作符	26
3.8.6 字符串连接操作符	26
小结	27

第 4 章 PHP 中的类型

4.1 PHP 中的类型简介	28
4.2 简单类型	29
4.2.1 整型	29
4.2.2 双精度型	30
4.2.3 布尔型	30
4.2.4 字符串型	30
4.3 复合类型	31

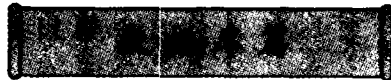
4.3.1 数组	31
4.3.2 对象	32
小结	33

第5章 程序控制

5.1 语句	34
5.2 判断语句	34
5.2.1 if 语句	35
5.2.2 switch 语句	39
5.3 循环语句	42
5.3.1 for 语句	42
5.3.2 while 语句	45
5.3.3 do-while 语句	45
5.3.4 循环嵌套	46
5.4 跳转语句	47
5.4.1 break 语句	47
5.4.2 continue 语句	48
小结	49

第6章 函数

6.1 函数定义	50
6.1.1 函数定义的一般规则	50
6.1.2 函数调用	52
6.1.3 return 语句	53
6.2 函数和变量的作用域	54
6.2.1 局部变量	54
6.2.2 全局变量	55
6.2.3 静态变量	56
6.3 一般函数问题	57
6.3.1 函数的返回值	57
6.3.2 参数传递	59
6.3.3 函数嵌套	60
6.3.4 递归调用	61
6.4 高级函数技巧	63
6.4.1 参数数目可变的函数	63
6.4.2 按值调用和按引用调用	66
小结	69




第7章 MySQL 入门

7.1 MySQL 的主要特征	72
7.1.1 一般特性	72
7.1.2 兼容性	73
7.1.3 表格特性	73
7.2 MySQL 的安装	74
7.2.1 安装 MySQL	74
7.2.2 关闭 MySQL	75
7.3 基本命令	75
7.3.1 创建数据库	75
7.3.2 建立数据表	76
7.3.3 填写数据表	77
7.3.4 SELECT 命令	78
7.3.5 UPDATE、DROP、DELETE 和 ALTER 命令	80

7.4 正则表达式	81
7.4.1 MySQL 正则表达式的 描述	81
7.4.2 正则表达式中的 保留字	81
7.5 安全问题	83
7.5.1 权限和口令	83
7.5.2 存储口令的位置	83
7.5.3 修改密码	84
7.5.4 备份	85
小结	85

第8章 PHP/MySQL 函数

8.1 基本函数简介	86
8.1.1 初始化 MySQL	86
8.1.2 创建数据库	86

8.1.3 连接数据库	86	10.1.1 没有连接	122
8.1.4 建立 MySQL 查询	87	10.1.2 权限问题	123
8.1.5 提取数据	87	10.2 语法错误	123
8.1.6 特殊用途函数	87	10.2.1 引号的问题	124
8.2 连接多个数据库	88	10.2.2 中断的 MySQL 语句	125
8.3 建立错误检查	89	10.2.3 拼写错误的名称	126
8.3.1 使用 Die 语句	89	10.2.4 逗号引起的错误	126
8.3.2 错误提示信息	89	10.2.5 未被引号引起来的字符串 参数	126
8.4 创建 MySQL 数据库	90	10.2.6 未被设定的变量	127
8.5 MySQL 函数	90	10.3 其他问题	127
8.5.1 MySQL 函数列表	91	10.3.1 数据太少或太多	127
8.5.2 MySQL 函数详解	92	10.3.2 分号的例外	128
小结	102	10.3.3 TEXT、DATE、和 SET 数据类型	128
第 9 章 PHP/MySQL 的 初步应用		10.3.4 通配符	128
9.1 查询数据库	103	10.3.5 NOT NULL 和空 记录	129
9.2 向服务器发送数据	106	10.4 条理清楚的检查	130
9.3 修改数据	108	小结	130
9.4 设置权限	113	第 11 章 PHP 和 MySQL 编程	
9.4.1 MySQL 用户管理	113	11.1 在 HTML 表中显示查询	132
9.4.2 创建用户并授权	114	11.2 应用实例——Weblog	134
9.4.3 连接的权限	116	11.2.1 最简单的 Weblog	135
9.4.4 用户权限与级别	117	11.2.2 通过 Web 页面来输入 信息	138
9.4.5 撤权并删除用户	119	11.2.3 添加数据库	140
9.4.6 设置用户权限的另一种 方法	120	11.2.4 搜索功能	149
小结	121	11.2.5 分页显示 MySQL 数据库 记录的类	150
第 10 章 MySQL 数据库 应用常见问题		小结	153
10.1 数据库连接	122		
第 12 章 会话		12.1.1 会话的概念	156
12.1 会话的概念	156	12.1.2 为什么要进行会话 跟踪	156

12.1.3 需要解决的问题	157
12.2 实现会话跟踪可能的手段	157
12.2.1 IP 地址	157
12.2.2 隐藏变量	158
12.2.3 cookie	158
12.3 会话的处理方法	159
12.3.1 确定会话的开始	159
12.3.2 变量传递	160
12.3.3 会话变量的存储位置	161
12.4 会话函数	161
12.5 常见问题	163
小结	163

第 13 章 客户/服务器

交互技术

13.1 CGI 的概念与编程	164
13.1.1 关于 CGI	164
13.1.2 CGI 应用程序	165
13.2 HTTP 标题	166
13.3 cookie	168
13.3.1 cookie 的问题	169
13.3.2 如何使用和设置 cookie	169
13.4 JavaScript	172
13.4.1 PHP 与 JavaScript 的关系	172
13.4.2 使用 JavaScript 的场合	172
13.4.3 如何联合使用 PHP 与 JavaScript	173
13.5 表单	173
13.5.1 表单简介	173
13.5.2 创建表单	173
13.5.3 INPUT 标识符	174
13.5.4 传递表单信息	178
13.5.5 示例	179
小结	180

第 14 章 面向对象程序设计

14.1 面向对象程序设计的特点及主要概念	181
14.1.1 面向对象程序设计的特点	181
14.1.2 面向对象程序设计的主要概念	182
14.1.3 PHP 中关于 OOP 的约定	182
14.2 类的基础知识	183
14.2.1 类的定义	183
14.2.2 属性	184
14.2.3 函数	185
14.2.4 继承	186
14.2.5 重载	187
14.2.6 类的作用域	188
14.2.7 别名(alias)	189
14.3 类与对象	191
14.3.1 对象建立	191
14.3.2 实例	192
14.4 与类相关的函数	198
14.4.1 打印对象	198
14.4.2 序列化	199
14.4.3 其他函数	199
小结	199

第 15 章 XML

15.1 XML 概述	201
15.1.1 XML 简介	201
15.1.2 文件的结构	203
15.1.3 XML 的语法规则	205
15.1.4 在 XML 文件中使用 HTML 标签	207
15.1.5 HTML 与 XML 的组合	209
15.2 数据结构的定义	212
15.2.1 DTD 的声明及引用	212
15.2.2 标签的类型	215

15.2.3 实体的声明与使用.....	219	15.2.5 属性的类型.....	230
15.2.4 标签的属性.....	224	小结.....	234



附录 A 名词解释	236	附录 B PHP 函数清单	245
-----------------	-----	---------------------	-----

入门篇

本篇包括第1章至第6章的内容，主要讲述PHP的基础知识。

第 1 章 PHP 简介

本章将讲述以下内容

- ◆ PHP 的起源
- ◆ PHP 的优、缺点
- ◆ PHP 的相关费用
- ◆ 与 PHP 配套的软件

PHP 是一种在服务器端运行的脚本语言，是编程语言和应用程序服务器的组合体。之所以这样来描述它，是因为它除了可以像一般的编程语言那样处理数值和变量外，还具有存取数据库数据、支持电子邮件和 HTTP 等 Internet 协议的特性。由于提供了对这些先进技术的支持，它使一个开发人员能在多种操作系统下迅速地完成一个小型的 Web 应用。这些突出的特点使 PHP 在短短的几年里得到了迅速的发展。

在自身发展的同时，PHP 在 GNU 的世界里找到了一个最佳的伙伴——Apache。Apache 不仅为 PHP 提供了理想的运行环境，而且由于两个产品都是开源代码的产品，所以开发人员可以使用它们的源代码将这两个产品完美的结合到一起。

1.1 PHP 起源与发展

PHP 是 Personal HomePage Program 的缩写，是由美国人 Rasmus Lerdorf 所创制的。在最初，它仅仅是 Lerdorf 利用 Perl 语言编写的一个程序，用来记录自己在线简历的访问者。后来，Lerdorf 又用 C 语言重新编写了该程序，并增加了对数据库的访问功能。在此期间，有很多人从 Lerdorf 那里拷贝了该程序供自己使用。因此，Lerdorf 专门编写了一些文档来介绍该程序，并发布了 PHP v1.0 版本。随着 PHP 使用人数和范围的不断增加，PHP 也增加了许多新功能，像循环语句、数组等，版本也从最初的 v1.0 版逐步升级，一直到了今天的 PHP v4.4 版。

PHP 作为完全免费公开代码的自由软件，它改变了自由软件技术要求高、代码记忆困难和使用不便等特点，一个会使用 Basic 语言的人就可以在短时间里掌握并应用 PHP 完成需要的任务。

1.2 PHP 特性

PHP 的发展史本身就充分说明了它的优点，然而“金无足赤”，PHP 也有它的不足之

处。充分了解了优点和缺点，就可以扬长避短，更好地应用此技术。

1.2.1 优点

➤ 跨平台 PHP 是一种跨平台的、有良好数据库交互能力的开发语言，它不仅可以在 Unix/Linux 平台下应用，也可以在 Windows NT 和 Windows 98 平台下应用。

➤ 源码开放 PHP 是免费的，所有的源码和文档都可以免费的复制和下载，用户可以对对自己的 PHP 程序做任意的处理。

➤ 语法简单、书写容易 PHP 是以基本语言为基础的，相对于 C++ 等传统语言来说，语法简单，程序的书写也很容易，同时 Internet 上还有大量的代码可以共享，所以对于想学习简单的 Unix 开发的人来说是一个好的入手点。

➤ 运行于服务器端 在开发 Internet 应用程序时，程序在哪一端运行是很重要的。运行于服务器端是 PHP 的一个重要优点，由于这个优点，即使一个 PHP 程序编写得很复杂很庞大，依然不会影响客户端的运行速度。

➤ 嵌入 HTML PHP 不需要编译成可执行文件，用户可以把 PHP 代码直接嵌入到 HTML 中混合使用，因此使用方便。

➤ 与 Apache 及其他扩展库结合紧密 PHP 与 Apache 可以以静态编译的方式结合，而与其他扩展库也可以用这样的方式结合（但在 Windows 平台除外）。这种方式最大化地利用了 CPU 和内存，同时极为有效的利用了 Apache 高性能的吞吐能力。外部的扩展也采用静态连编，可以达到最快的运行速度。由于与数据库的接口也使用了这种方式，所以是本地化的调用，这也让数据库尽可能发挥了效能。

➤ 良好的安全性 由于 PHP 本身的代码开放，所以它的代码在许多工程师手中进行了检测，同时它与 Apache 编译在一起的方式也可以让它具有灵活的安全设定。到目前为止，PHP 的安全性能得到了广泛的公认。

1.2.2 缺点

➤ 安装复杂 PHP 的每一种扩充模块并不是完全由 PHP 本身来完成的，它需要许多外部的应用库，如图形需要 Gd 库、LDAP 需要 LDAP 库等。只有将这些应用库都安装好，再连编进 PHP 中来才可以实现用户的需要。所以，一般开发人员在使用 PHP 前都会遇到这样复杂的安装过程，正是这样的问题让大家转而使用其他的开发语言。

➤ 数据库接口缺乏规范 PHP 的所有扩展接口都是独立团队开发完成的，没有统一的规范。虽然 PHP 支持多种类型的数据库，可是针对每种数据库的开发语言都完全不同。这种情况造成的结果就是针对一种开发的数据库，在升级后需要开发人员进行几乎全部的代码更改工作，这样的工作量是很大的。另外，为了让应用支持更多的数据库，就需要开发人员将同样的数据库操作使用不同的代码写出多种代码库，使得程序员的工作量又大大增加了。

➤ 缺少组件的支持 由于没有组件的支持，所有的扩充只能依靠 PHP 开发组给出的接口，而且这样的接口还不是很多。同时对于大型的站点或企业级的应用来说，难以将集群、应用服务器这样非常必要的特性加入到系统中去。

尽管 PHP 目前还存在这样或那样的缺点，但是随着 Linux 系统的普及和 PHP 自身的

发展，相信这些缺点将会逐步得到改善和修正，总的来说它是一种优秀的具有良好发展前途的语言。

1.3 费用问题

如果操作系统采用 Linux、服务器软件采用 Apache、数据库采用 MySQL、编程语言采用 PHP 这套方案来建立一个 Web 服务器，整个系统根本不需要支付任何软件费用。就 PHP 本身来说，它的完全免费意味着所有的源代码和文件都可以免费的复制、下载和分发，这一点尤其适用于经济并不宽裕的国内用户。

如果换一种组建方法，操作系统采用 Windows NT 4.0 或 Windows 2000，服务器软件采用 Ms IIS，编程语言采用 MS SQL，这种组合的软件费用是相当高昂的！加之网站建设的其他费用，实现大型站点的梦想是需要强大的经济后盾的。

1.4 Apache 与 MySQL

可以配合 PHP 共建网站系统的两个合作伙伴是 Apache 和 MySQL。

Apache 与微软公司的 Ms IIS 类似，同属于一种 Web 服务器。根据 Web 服务器调查公司 Netcraft (<http://www.netcraft.co.uk/Survey/>) 的调查，在 Internet 上有 50% 以上的 Web 服务器使用 Apache。可以说，Apache 是目前最流行的 Web 服务器。

MySQL 是一个非常精巧的数据库服务器软件，对于中、小型系统应用是非常理想的。它除了支持标准的 ANSI SQL 语句外，更重要的是 MySQL 还支持 Windows/Linux/Unix 等多种平台。而在 Unix/Linux 系统上，MySQL 支持多线程运行方式，从而能获得相当好的性能。此外，与 PHP 和 Apache 两个软件一样，它也是属于开放源代码软件。

小 结

本章是对 PHP 的一个简单概括介绍。介绍了关于 PHP 的起源，PHP 的优点和缺点，使用 PHP 的费用以及与 PHP 配套使用的软件等问题。

第 2 章 PHP 安装

本章将讲述以下内容

- ◆ 主机的组建
- ◆ PHP 的安装

Web 虚拟主机可以采用外部采办、自建或者采取介于两者之间的折中方案；而 PHP 的安装包括 Windows NT/Windows 2000 平台、Unix/Linux 平台以及 Windows 95/98 平台下的安装。

2.1 组建主机

在安装 PHP 之前，用户必须考虑如何建立自己的 PHP 主机。一般地，建立 PHP 主机可以由用户自己选择。

2.1.1 外部采办

现在许多公司都开始提供启用 PHP 的 Web 站点，因此租用远程虚拟主机已经成为一种非常流行的方式。这种外部采办主机的方式大大方便了用户，而且大多数情况下，远程虚拟主机的成本效益是非常高的，对于中、小型站点来说这一优势就更加明显。

除了成本效益高以外，外部采办虚拟主机最主要的优点就是用户完全无须关心管理上的问题。管理的问题不仅相当繁琐，甚至难以解决。主要表现在以下方面：

- 硬件升级
- 软件升级
- 数据备份
- 邮件服务器
- 电源
- 带宽
- 潜在的黑客攻击

对于外部采办的虚拟主机来说，所有以上这些头痛的问题都不需要用户自己去解决的，完全交给了提供虚拟主机服务的公司。用户的站点出现的任何问题，都由 ISP 的技术人员来处理。

当然外部采办虚拟主机也有自己的缺点。例如，用户的需求与提供虚拟主机的 ISP 的便利之间往往会发生冲突。原因是作为用户来讲，总是希望得到最快、最高效的运行效果。而作为 ISP 来说，更希望管理简单和方便。用户与 ISP 之间的冲突便不可避免。

因此采用租用虚拟主机的方法，用户应该尽量使自己的需求符合一般的要求。需求越简单越常规，就越容易找到虚拟主机服务。如果用户有很多特殊的需求，尤其是站点中包含某些特殊内容。例如暴力、可能引起法律诉讼的内容、安全信息等，要寻找到虚拟主机服务是非常困难的。在这种情况下，用户就应该考虑自建主机。

☞ 特别提示：

基于 Linux 平台的 PHP 虚拟主机非常容易找到，而且收费比较低廉；而基于 Windows NT 平台的 PHP 虚拟主机则很少，费用较高。

2.1.2 用户自建

当用户有很多特殊要求或者要创建一个大型的 Web 站点时，就有必要自己建立主机。自己建立主机同样有很多的优点：

- 随着连接费用的降低，在创建大型 Web 站点时总体的费用也会降低；
- 只要有足够的技术实力，则当网站出现任何问题都可以迅速解决，而不必浪费大量的宝贵时间与 ISP 的技术人员交涉。

但是，自己建立主机与租用虚拟主机相比，要做的工作会更多，尤其是管理方面的问题。对于大多数的网站来说，日常管理的费用要比采购硬件的费用高得多。一旦网站建立起来后，必须要保证它能够正常、良好的运行。而在网站的运行过程中，可能出现各种各样的问题。防范黑客攻击就是一个关于安全性的最常见问题，这些问题都需要自建主机的用户自己解决，这就要求自建主机必须有多方面的技术力量。

所以，要自建主机并使其正常运行，除了有必须的资金以外，充足的技术储备和良好的管理能力也是必须的。

2.1.3 其他方法

除了租用虚拟主机和完全用户自建外，还有几种其他方法，用户可以根据自身的能力找到最适合的解决办法。

➢ 托管 采用托管的方法，用户需要自己购买主机，对主机进行配置，办理各种手续，然后将主机连接到提供虚拟主机的公司的网络上，并由该公司负责对主机的运行进行监视。用户必须自己对主机的软件和硬件进行维护，一旦出现了问题，提供虚拟主机的公司并不负责解决，而只负责及时通知用户，用户需要自己解决所有的技术问题。如果需要对方提供技术支持和其他服务，就需要额外付费。

从上面的过程可以看出，采用托管的方式，用户并不能节约资金。但是可以省出一些不必要的监视主机运行的时间。

➢ 专用服务器 由用户提供资金，主机服务公司负责购买服务器并将其配置得符合用户的需要，这台服务器由用户专用。除此之外，主机服务公司通常还负责对服务器的软、硬件进行维护和提供必要的技术支持。采用这种方法，用户提供的资金除了用于购买硬件外，还包括主机服务公司的管理费用。在长期运行的情况下，使用这种方法的用户会更节省资金。对于要建立大型的 Web 站点而又没有足够的技术实力和管理经验的用户来说这种方法尤其合适。

2.2 安装 PHP

确定如何建立 PHP 主机后，我们就可以来看看如何安装 PHP 了。采用外部采办建立 PHP 的用户完全可以不考虑这个问题，下面的内容只适用于要自建主机和采用托管方法建立主机的用户。

在开始安装之前，让我们首先了解一下保证 PHP 正常运行所需要的软件和硬件两个方面的开发环境。推荐的开发环境是：

- 软件 Apache 1.3.12 版、MySQL 3.22 版；
- 硬件 CPU 为 P III 500、512M 内存和 18GB 硬盘空间。

☛ 特别提示：

在现有条件下，这种软、硬件要求是很容易满足的。如果达不到这个要求，最低配置的开发环境是：Apache 1.3.12 版、MySQL 3.22 版、CPU 为 P 200、64MB 内存和 9GB 的硬盘空间。

满足上述条件后，我们就来看看在各种平台上 PHP 的安装方法。

2.2.1 UNIX/Linux 平台

要在 UNIX/Linux 平台上安装 PHP，需要具备以下条件。

- 已安装了 UNIX/Linux 操作系统的服务器。
- PHP 源代码发行版，可以到下面的网址下载：
<http://www.php.net>
<http://www.phpuser.net>（中文）
- Apache 源代码发行版，可以到下面的网址下载：
<http://www.phpuser.net>（中文）
<http://www.chinaasp.com>
- MySQL 数据库，可以到下面的网址下载：
<http://www.mysql.com>（可以查找在中国的镜像网站）
<http://www.phpuser.net>（中文）
<http://www.chinaasp.com>
- Bison 和 flex，可以分别在以下地址下载：
www.gnu.org/software/bison
<ftp://ftp.ee.lbl.gov>
- PHP 的支持软件，包括邮件服务器和 BC 数学包等
- ANSIC 编译器。
- Gnu make。
- HTTP 配置文件的路径。
- Apache 源文件的位置，通常情况下是/user/local/apache_1.3.x。
- Apache 守护进程，即 httpd 的位置。

具备了上述条件后，就做好了安装前的准备工作，可安装 PHP。下面具体介绍安装步骤。

1. 解压 Apache

将下载的 Apache 源代码版本进行解压。通常情况下将其解压到 `/user/local/` 位置。在 shell 提示符下输入以下代码片段：

```
gunzip -c apache_1.3.x.tar.gz
tar -xf apache_1.3.x.tar
cd apache_1.3.x
./configure
```

2. 解压 PHP 源代码发行版

将下载的 PHP 源代码版本进行解压。通常情况下将其解压到 `/user/local/` 位置。在 shell 提示符下输入以下代码片段：

```
gunzip -c php-4.x.tar.gz
tar -xvf php-4.x.tar
cd php-4.x
```

3. 配置编译连接

配置 PHP 是非常重要的步骤。通常情况下选择编译连接为一个 Apache 模块，并带有 MySQL 数据库支持。在 shell 提示符下输入以下代码片段：

```
./configure --with-apache=../apache_1.3.x --with-mysql --enable-track-vars
```

4. 进行编译连接并安装 PHP 模块

输入如下代码：

```
make
make install
```

5. 配置并编译连接 Apache 守护进程

在下面的代码中，`/etc/httpd` 是 HTTP 配置文件的路径。如果配置文件在其他路径下，用正确的路径将其替换即可。

```
cd ../apache_1.3.x
./configure --prefix=/etc/httpd
--activate-module=src/modules/php4/libphp4.a
make
```

6. 停止并替换 httpd 可执行文件

输入如下代码：

```
cd src/support
./apachectl stop
cd..
cd httpd /usr/local/bin
```

7. 设置 php.ini 文件

可以在任意的文本编辑器中对其进行编辑。

8. 设置 PHP 文件扩展名类型

这一步的目的是告诉服务器什么扩展名的文件是 PHP 文件，可以在 `httpd.conf` 文件