Mastering PHP 4.1

PHP 4.1 从入门到精通

〔美〕 Jeremy Allen
Charles Hornberger 等著
毛 选 等译

使用最新版PHP服务器端嵌入式脚本语言，
构建交互式多功能Web应用程序

SYBEX
PHEI
電子工業出版社
Publishing House of Electronics Industry
http://www.phei.com.cn

Mastering PHP 4.1

PHP 4.1从入门到精通

〔美〕 Jeremy Allen
Charles Hornberger 等著

毛 选 等译

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 提 要

PHP是一种HTML嵌入式脚本语言，容易使用、威力强劲，是开发万维网应用程序的最佳工具。本书面向万维网开发人员，在介绍大多数基本而又通用的编程原理以及PHP 4.1全面知识的同时，利用大量的实例总结了万维网开发经验，涉及会话过程、数据库交互、可移植性、调试与排错等。

无论是编程的初学者，还是具有多年编程经验的老手，都能从本书中获取有益的知识。

Copyright©2002 SYBEX Inc., 1151 Marina Village Parkway, Alameda, CA 94501. World rights reserved. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior agreement and written permission of the publisher.

本书英文版由美国SYBEX公司出版，SYBEX公司已将中文版独家版权授予中国电子工业出版社及北京美迪亚电子信息有限公司。未经许可，不得以任何形式和手段复制或抄袭本书内容。

版权贸易合同登记号：01-2002-0898

图书在版编目（CIP）数据

PHP 4.1从入门到精通/（美）阿伦（Allen, J.）等著；毛选等译. - 北京：电子工业出版社，2002.8
（从入门到精通）
书名原文：Mastering PHP 4.1
ISBN 7-5053-7833-3

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

中国版本图书馆CIP数据核字（2002）第052059号

责任编辑：李 莹
印 刷：北京天竺颖华印刷厂
出版发行：电子工业出版社 http://www.phei.com.cn
北京市海淀区万寿路173信箱 邮编：100036
北京市海淀区翠微东里甲2号 邮编：100036
经 销：各地新华书店
开 本：787×1092 1/16 印张：38.625 字数：980千字
版 次：2002年8月第1版 2002年8月第1次印刷
定 价：58.00元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换，若书店售缺，请与本社发行部联系。联系电话：（010）68279077

献给Erin：你的耐心与细心让一切成为可能。

——Jeremy Allen

献给Charles Semmelman

——Charles Hornberger

致　　谢

一切都从我的父母开始，所以首先感谢他们把我带到这个世界上来！其次，感谢Erin，谢谢你放弃了那么多和我相聚的时间，而且谢谢你的耐心与理解。

感谢参与本项目的整个团队！虽然Tom Cirtin没有直接参与本项目，但仍要感谢他给我提供了首次从事专业写作的机会。接着要感谢的是Diane Lowery，谢谢你一直与我一道工作。感谢Charlie Hornberger，谢谢你在本书中倾注了大量的心血。感谢Pete Gaughan，谢谢你不知疲倦地编辑本书和提供了独到见解。感谢Liz Burke为本书做了统稿工作。Sybex的团队精神一直是本书得以出版的关键。

感谢elliptIQ的开发团队，谢谢你们对我撰写本书的支持。

——Jeremy Allen

我想感谢许多人，谢谢他们为本书的出版所给予的帮助，尤其是我的合作者Jeremy Allen；感谢Liz Burke使一切都保持得井井有条，即使在我遇到困难的时候；感谢编辑Pete Gaughan在本书上所倾注的心血；感谢Diane Lowery给予我的指导。

——Charles Hornberger

简　介

10年以前，World Wide Web（万维网，简称Web）崭露头角。那时创建的万维网是用来共享信息的一个完全静态的世界，而且是一条单行线：阅读者没有任何办法与站点进行交互。如果内容需要更新，它要求内容创作者的直接干预。动态万维网站点的概念实际上是前所未有的。那时，当今所发生的计算机革命还不完全是我们日常生活的一部分。

现在，世界上似乎充满了可轻易访问到的信息。当我们说“因特网”时，并没有把它看做满嘴咒语的神秘巫师。因特网是一个重要的经济力量——一个动态的、活跃的、充满多媒体内容的地方和一个需要我们永远去领会的词汇。现在，编程人员拥有大量可自由支配的编程语言。编程语言的选择余地也非常大，而且这些编程语言在能力、使用方便性以及支持硬件的抽象层次方面也有极大的不同。

欢迎使用PHP，它是一种HTML嵌入式脚本语言，容易使用，威力强劲，并融合了现代编程语言的一些最佳特性，进而提供了编写万维网应用程序的一种独特而又使人振奋的方法。PHP是大家要选择的最佳工具吗？如果读者正在开发万维网应用程序，它就是你的最佳工具。尽管有大量可以使用的语言，但PHP的原设计意图就是面向万维网开发的。所有这些因素结合起来使PHP变得非常简洁和容易理解，进而成为了一种使应用程序开发变得极其轻松的语言。尽管优良的设计和正确的管理对开发高质量的软件始终是至关重要的，但选择正确的工具也同样重要，就万维网开发来说，PHP就是最佳的工具。

数据驱动的动态应用程序和万维网站点已经成为实现各种商务和交流的一条重要途径。PHP在几乎所有主流硬件平台上都能运行，并且是独立于它们的。PHP可以与各种可供利用的开放式源代码关系数据库管理系统（RDBMS）之一（比如MySQL、PostgreSQL或Interbase）组合起来，构成一个强有力的、完全独立的开发平台。为了使应用程序变得更好，所有这些应用程序的“内脏”（即它们的源代码）也是可利用的。另外，PHP也可以从个人项目缩放到容量极大的应用。

PHP的发展历程

1994年，万维网仍处于起步阶段。可用来生成动态内容与数据驱动应用程序的服务器端脚本描述语言及工具的选择余地还十分小，而且这些可利用的语言及工具也十分简单。很显然，任何试图给网站添加动态内容的开发人员，都特别需要灵活的服务器端脚本描述语言。PHP的原开发者Rasmus Lerdorf把PHP（代表Personal Home Pages）设计为Perl脚本的一个小子集，以收集关于查看其简历的那些访问者的数据。这个初始版本可以看做某个黑客写给其他黑客的一组编程工具，用于使站点变得更有用、更具魅力。

1995年前后，公开发布了PHP的一个版本。在拼凑了一个用于翻译HTML窗体数据的实用程序（称做FI）之后，诞生了PHP/FI。在添加了对mSQL（另一个源代码开放RDBMS）的

此为试读，需要完整PDF请访问：www.ertongbook.com

支持之后，PHP/FI才开始变得真正有威力。此后，PHP/FI的使用人数急剧增加，到1997年中期，PHP引擎经过了彻底的重写，并形成了版本3.0。PHP不再是其创始人的一个业务爱好，而成为了一个有许多志愿者的大型而又不断增长的软件开发项目。PHP在专业软件开发人员之间的使用与“智力共享”仍在不断增长。

PHP 3

从PHP/FI 2到PHP 3的升级，标志着一群开发人员对PHP的彻底重写。这次重写提高了PHP的性能，扩充了可用于PHP的API（应用编程接口），并把这一核心语言改进成为一种更一致、更灵活的语言。大量数据库API增加到PHP 3上，进而给开发人员提供了更大的灵活性和选择余地。PHP 3被赋予了更强有力的语法规则，用于表达更传统的编程结构；此外，它增加了一组数量有限的面向对象编程（OOP）特性。虽然PHP决不是一个纯粹的OOP环境，但这些新特性已使熟悉OOP的开发人员能够在较熟悉的参数内做开发。

当PHP 3中掺杂了大量的改进以及其他许多编程语言中所没有的特性之后，它已经成为一种非常令人激动而又强有力的语言。从版本3开始，PHP的正式名称不再代表Personal Home Pages，该语言现在直接叫做PHP：Hypertext Preprocessor（超文本预处理器）。

PHP 4.1

在PHP 4及现在的PHP 4.1中，改进的程度不如PHP/FI 2到PHP 3的改进那么大，但这两个版本中的改进也是显著的和值得注意的。PHP的效率已经很高，但在PHP 4.x中，最显著、最重要的特性之一是运行效率极高的Zend脚本描述引擎。Zend Technologies公司开发了大量的PHP附件，其中的某些附件不是免费的，并且已得到商业公司的支持。不过，大多数的修改是源代码开放的，并且是免费的。另外，PHP 4.x中还包括了几个新的库和API，本书将对它们做较深入的探讨。

PHP已经发展成为一个真正的电子商务解决方案，而且它现在还拥有“最公用的Apache模块”的桂冠。根据E-Soft公司所做的一项调查显示，从2002年1月开始，PHP作为一个模块已安装在了46%以上的Apache安装上。在服务器端编程工具中，最强大的竞争对手是mod_perl，调查结果显示近20%的服务器上已安装了mod_perl。尽管E-Soft的调查结果不是结论性的，但这些统计数字极具迷惑力，因为Apache给60%以上的万维网站点提供着服务。

注意： 没有任何方法可用来实际检验这些统计数字的真实性，而且这些数字只用于反映PHP的估计增长量。要想详细了解这项调查的方法论述，请参见www.securityspace.com/s_survey/data/index.html网页。

版本4.1中的新特性

PHP的最新版本含有几个明显有别于版本4.0.x的变化，其中最显著的是与安全性有关的变化，这些变化已经改变了用户输入的存取方法。在这个新的模型中，“外部”值（像经由窗体提交的和包含在cookie中的数据）现在可以通过专门的全局数组来获取：

数组	内容
$_GET	使用HTTP GET方法所提交的任何值
$_POST	使用HTTP POST方法所提交的任何值
$_COOKIE	HTTP Cookie报头内的任何值
$_SERVER	万维网服务器所创建的任何名称/值对
$_ENV	来自万维网服务器环境中的任何变量
$_REQUEST	来自HTTP Request中的任何值
$_SESSION	任何会话变量

上述新变量的功能与那些旧的$HTTP_*_VARS变量（譬如$HTTP_GET_VARS，$HTTP_POST_VARS等）相类似，并且在任何作用范围内都是自动有效的。做这些修改的目的是为了鼓励编程人员停止使用register_globals配置选项来依赖外部变量在全局范围内进行自动注册，因为这种用法往往会导致安全性问题。欲知详细情况请参见www.php.net/release_4_1_0.php。

这个新版本还包含了总体的性能改进，并大大提高了PHP的Windows版本的运行速度和可靠性。

注意：2001年12月，发布了PHP的一个维护性版本4.1.1，目的是为了更正4.1.0初始版本中的少量问题。

PHP的优点

PHP提供了什么特性呢？动态内容可以通过各种服务器端解决方案来提供。重要的是要保证为特定的问题选择正确的工具。PHP为万维网应用提供了一个特性丰富的开发平台，此外还提供了一组用途广泛的API，并且能与大量源代码开放的和商业的数据库进行通信——总之，它是可互操作的，并且已被设计成能与大量技术协同工作。PHP是源代码开放的，这意味着其代码的核心部分可免费使用。PHP还是可移植的，并且能与各种万维网服务器和操作系统协同工作，其中包括Linux和Windows。

虽然PHP具有许多优点，但要重点注意的是，在选择开发平台时应对应用程序的各个方面都有充分的了解。应该反复权衡和考虑每个待选平台，以便做出最佳的选择。很显然，几乎没有什么万维网应用程序是用C++写成的，因为它的设计意图不是面向万维网应用程序开发的。但是，像PHP、JSP、ColdFusion和ASP之类的服务器端技术却是专为这类开发设计的。事实上，这些技术中的许多技术都与万维网服务器有某种关联。

有些技术比其他技术更容易学习和使用。许多语言为了获得使用的简便性而牺牲了高级功能。这种易使用性只是选择编程语言来开发应用程序时要考虑的最重要因素之一。如果开发团队的开发人员不能有效地学会和使用某种编程语言，该应用程序将会受到一定伤害。

PHP的发展方向

那么，PHP是怎么成为这样一种力量的？它的未来会怎么样呢？PHP获得成功的关键因素之一是它拥有开放的源代码，而且是完全免费的！假如一个产品具有这样的特性和功能，

那么花费数千美元来获得具有类似特性和功能的企业级解决方案就非常值得了。虽然一个开发平台的价格只是一个企业级解决方案的总成本中的因素之一，但PHP已经连续获得了成功，因为它是一项已得到实践检验的技术，但因其价格而未流行起来。PHP赢得了商业的支持，并吸引开发商开发了它的附件，但它的核心仍是100%免费的。

目前，PHP还没有作为一个独立使用的服务器来运行。严格地说，它还不是一个应用程序服务器。PHP的设计意图不是面向企业级应用。所谓“企业级”指的是能够给大型应用程序中的关键功能提供动力的那种能力，而这类大型应用程序又允许成千上万执行复杂任务的用户同时使用它们。给应用程序中的关键功能所提供的这种动力必须从一开始就得到很好的规划，以便能实现企业应用程序所需要的规模级别。在选择用来开发应用程序的平台时，需要重点了解这些局限性。为了让PHP进入企业领域并在这个应用领域内具有真正的竞争力，PHP体系结构中必须发生几个变化。

将来会增加到PHP上的特性和变化，随着重要性以及与其能力和当今用户的相关性而变化。对PHP能够或可能向何处发展，我们有一些想法。在下面所说的这些特性中，任一特性在PHP开发团队将来决定采用PHP以前只是想想而已。如果这些特性得到实现，无论他们决定做什么，结果都将是十分激动人心的。

作为应用程序服务器的PHP

使PHP能够成为一个企业级解决方案的第一步是让PHP成为一个真正的应用程序服务器。如果PHP拥有多个线程，那么，该应用程序服务器只是一个固定的服务器进程。如果把PHP作为一个独立使用的应用程序服务器来运行，它就可以更有效地处理数据库连接。这将使一定数量的持久性数据库连接能够经常处于打开状态，并由该假想的PHP应用程序服务器用来处理请求。从这样一种设计中得到的性能收获将是十分显著的。

拥有一个固定的服务器进程在把PHP与一个远程过程调用（RPC）协议（比如SOAP或一个更低级的协议）一起使用时也是有帮助的。当出现了一个RPC请求时，服务器用该远程过程调用的结果来响应它。这使PHP既能提供内容，又能处理来自万维网浏览器的请求，以及来自知道如何发送RPC请求的任一应用程序的请求。

例如，该PHP应用程序服务器可以接受来自某个财务程序的远程调用并处理结果，然后把结果返回给财务程序。在该PHP服务器正在处理来自一个内部财务应用程序的一个请求的同时，它也可以通过返回一个XHTML文档来响应来自某个万维网浏览器的请求。这种情况似乎有点牵强，但在实际生活中十分常见，而且优点也十分明显。整个开发方法可以从严格处理一个万维网浏览器变化到处理几乎任一类型的客户端软件。

拥有一个固定的服务器进程也使布局编程逻辑层变得更加容易。在应用程序内使用相互独立的逻辑层，可以使编程任务的相互依赖性（和交叉性）变得更微弱。一个团队只需要知道如何与业务逻辑层交互，并创建该万维网浏览器界面，而使用相同信息的另一个团队则可以实现一个与财务应用程序接口的界面。这就隔开了该界面与编程逻辑。与此同时，另一个开发团队可以开发该应用程序中提供全部逻辑并为所有不同客户界面返回所需数据的那一部分，然后由这些不同的界面来处理这些数据，生成必要的信息，并把这些信息返回给发出请求的客户。

把PHP重写成一个应用程序服务器意味着该语言的大部分将要彻底重写。优点很多，也很激动人心。但是，PHP将向何处发展是一个争论激烈的话题。

其他改进

把PHP变成一个真正的应用程序服务器无疑是改进这个资源的最大任务之一，但这肯定不是惟一可能的改进。随着应用程序在大小和复杂性方面的增强，更友好的调试界面是必不可少的，而且更严谨的数据类型化也是十分必要的。提供一个可选但严谨的数据类型模型，可以使应用程序在大小和复杂性增加时更不易出错。

另一个有趣的特性是把PHP脚本编译成能在.NET Common Language Runtime上运行的.NET能力，按照这个相同的思路，或许我们将实现把PHP脚本编译成Java字节码的能力，以使PHP脚本能在Java虚拟机上运行。如果PHP是一个应用程序服务器，则这些已编译的对象就可以在服务器中得到缓存，进而在理论上允许PHP极大地提高性能。

注意： 还有一些得到极高赞誉的技术，这些技术已把PHP页面缓存为立即编译的对象。Zend Cache允许脚本缓存，并省去了脚本在每次执行时的编译过程。有关这项技术的进一步信息，请访问www.zend.com站点。

关于本书

在撰写本书时，我们的目标之一是提供实际的真实例子，从概念到实现全面说明PHP中的功能。提供高质量的真实例子，将使读者更容易把从本书中学到的概念和思想运用到自己的应用程序中。我们的另一个目标是用一种精确而直观的方式提供信息，以便轻松地掌握用PHP 4.1做万维网开发的概念，同时轻松地使用这些概念。

我们的愿望是与大家共同分享我们的万维网开发经验，并介绍如何避免编程时易犯的常见错误——从编程风格问题到具体的PHP编程难点。本书中的某些题目似乎超出了讲解PHP的范围，但我们认为事实并非如此。要想高效地编写出软件应用程序，必须先正确地设计它，然后根据设计进行编程，以便应用程序在半年之后仍有意义。任何一个项目必须是可管理的，并设计成允许团队协作。我们将避免在与PHP无关的题目上花费过多的篇幅，但我们将详细叙述与PHP有关和影响PHP使用的那些题目。这种解释方法将既兼顾与更有效的万维网开发有关的更广泛知识，又兼顾关于PHP 4.1的全面知识。

在本书中，我们为初次接触编程的绝对初学者到已具有数年编程经验的老手，都提供了相应的题目和例子。我们在本书的前几章中介绍了大多数基本而又通用的编程原理，但在全书的每个例子中，共同的主题都是万维网开发。

除了与万维网应用程序的成功开发密切相关的那些方面外，本书还涉及了其他几项技术和几种编程语言。本书假设大家已经具备了HTML、SQL及配置万维网服务器的一些基本知识。本书还讨论了与PHP一起使用这些语言的基本知识，并提供了几点防止误解的说明，但想用一本书的篇幅来详细而全面地讨论这些内容是完全不可能的，因此我们不打算这么做。本书始终把重点放在用PHP实现实际的解决方案上。

选配光盘上的内容

全书中所使用的全部源代码都放在了本书选配光盘上，其中包括PHP的源代码及PHP中所使用的任何模块。选配光盘上还包含了源代码形式和Windows二进制形式的PHP和Apache的完整发行包副本，PHP开发者们所编写的整套PHP文档资料，以及供读者用于数据库编程练习的MySQL发行包副本。

本书中所使用的平台

Red Hat Linux 7是本书所使用的主平台。所有代码都已用PHP 4.1做过测试，并且是在Red Hat 7的一个洁净安装上从源代码中编译而来的。PHP的原设计平台是UNIX，而且使用这种类型的平台可以得到最大的稳定性和性能。本书不可能介绍Linux服务器的安装，但我们将稍微涉及安装和运行PHP的难点问题。

Windows安装上的许多具体选项和技巧也在本书中有所涉及。PHP在Windows上的性能仍比不上在UNIX产品上的性能，但对该语言来说，这是一个正在成长的领域。

在全书中，我们虽然指出了某些特性要求指定安装或编译选项的地方，但附录A更完整地说明了如何安装和配置PHP及几个相关的工具，其中包括Apache万维网服务器本身在Linux和Windows 32位版本（98、Me和2000）上的安装与配置。

与作者的联系方式

如果大家对本书有疑问、意见、建议或反馈信息，我们十分愿意接受。请用下列E-mail地址与我们联系：

Jeremy Allen：jallen01@mindspring.com

Charlie Hornberger：charlie@nothingspecial.com

目　录

第一部分　PHP基础 1

第1章　PHP概况 1

针对万维网的开发 1

了解PHP 3

第2章　变量 16

变量理论与命名 16

PHP与数据类型 18

标量数据类型 19

复杂数据类型 24

特殊数据类型 27

常数 28

其他类型 29

使用变量、PHP风格 33

第3章　运算符与表达式 41

运算符 41

表达式 54

第4章　流程控制与函数 60

程序中的抉择 60

循环 67

利用函数 77

第5章　串与数组 91

串 91

数组 128

第二部分　程序设计原理与PHP 139

第6章　面向对象的程序设计 139

OOP的基础知识 139

PHP的OOP实现 142

第7章　调试与错误 159

错误类型 159

调试方法 172

了解错误级别 174

处理错误 176

此为试读，需要完整PDF请访问：www.ertongbook.com

可视化调试程序 179

第三部分 数据流程 185

第8章 SQL与数据库交互 185
规范化数据库 185
关系数据库管理系统（RDBMS） 193
SQL基础 194

第9章 窗体与用户交互 228
窗体的基础知识 228
利用PHP的窗体支持 243
设计易使用的界面 254
保存与编辑数据 261
高级窗体处理 271

第10章 数据有效性检查 279
编写一个有效性检查类 279
用内部PHP函数检查数据 291

第11章 会话 297
cookie 297
会话的复杂问题 301
会话的基本知识 302
购物车 314
购物车实现 315

第12章 安全性 319
认证用户身份 319
杂凑数据 334
授权用户 336
加密数据 341

第13章 文件与联网 347
处理文件 347
目录 361
联网技术 367

第四部分 如何连接PHP 373

第14章 万维网应用程序开发 373
为什么重要 373
布置万维网应用程序 374
源代码管理 375
使用CVS 378
程序设计风格 388

第15章　XML与XHTML 391
XML概述 391
PHP与XML 401
用XSLT变换XML 426
用XML做数据交换 431
XHTML 439
第16章　LDAP 446
LDAP浏览 446
PHP与LDAP 449

第五部分　实际运用PHP 465

第17章　PDF 465
PHP与PDFlib 466
创建PDF文档 469
在现实世界中使用PDF 493
第18章　生成图形 504
设置PHP来创建图像 504
图像基础 506
绘制基本几何插图 510
绘制图表 513
应用中的注意事项 523
第19章　电子邮件 524
了解电子邮件 524
用PHP发送电子邮件 527
从头开始编写一个POP客户软件 528
安装IMAP扩展 531
用IMAP扩展阅读电子邮件 532
Web邮件应用举例 546

附录 563

附录A　PHP安装简明教程 563
附录B　PHP配置选项 586
附录C　XHTML实体 593

第一部分　PHP基础

第1章　PHP概况

为万维网或万维网类的运用（如企业内部网）而开发应用程序和站点，已经成为计算和编程工作的最主要方面之一。如果可以数字地实现万维网应用程序，那么某些人正在某处设法使它适应万维网浏览器。作为一名开发人员，了解万维网活动行为的各种对象（静态与动态页面，客户端与服务器端系统等）是提高开发灵活性必不可少的一步。

PHP以大家熟悉的C、Java和Perl之类的编程语言为基础结构。它通过提供轻松操纵动态HTML内容所必需的工具来帮助创建这种内容。PHP正在变成提高网页表达能力的卓越工具之一，因为它不但使用起来容易，而且也十分有威力。建立一些基本的脚本、测试来回转移数据的两种主要方法、并学会注释PHP代码，就可以说明PHP的特性是多么容易掌握。

针对万维网的开发

术语“万维网开发”的含义十分广泛，而且是分类各种开发活动的一个通用术语。万维网开发可以指从放置一个静态HTML页面到一个小型万维网站点上，到开发一个处理关键商务通信的大型洲际企业内部网的任何事情。但是，这些开发活动又确实可以分成便于管理的几大类别。

万维网应用程序

为了深入探讨万维网应用程序开发的话题，必须先弄清楚术语“应用程序”：应用程序是什么？应用程序应该做什么？

应用程序就是任何一个开发用来简化或执行任务的软件。任务的级别从非常具体变化到更加普通。一个接受学生6门课的分数、求平均分并以报表形式总结分数的程序是一个简单但作用有限的应用程序。相反，一个提供通信方法的应用程序则要复杂得多，而且实现一个更普通的目标，如在线群件应用程序（允许用户协调其工作流程的程序）。尽管群件应用程序的应用范围要比上述分数计算程序的应用范围广泛得多，但这两者都是应用程序。

那么，万维网应用程序具体指什么呢？一般说来，一个万维网应用程序就是一个利用因特网所提供的通信普遍性与方便性的应用程序。万维网应用程序的较严格定义，也是本书中将要使用的定义是：一个万维网应用程序指的是一个使用万维网浏览器作为客户端软件的

应用程序。大多数万维网浏览器可以使用的客户端技术有许多。一般说来，影响最深远、最容易访问的万维网应用程序是那些使用简单而标准的超文本标记语言（HTML）的应用程序。与术语“万维网应用程序”严格相符的几个例子是基于万维网的银行系统、拍卖程序以及新闻站点。

静态与动态网站

静态站点具有不变的内容，除非站点创作者更新这些内容，而且这些站点满足许多人的要求，因为它们允许信息共享。但是，静态站点不提供与访问者的交互，而且也不按可编程方式完成任何任务。

动态站点允许用户交互。尽管动态站点像静态站点一样，也在客户界面使用HTML，但它允许用户采取个人的和可自定义的操作，比如预约和定购某个特定航班甚至座位。在线订票系统的用途是显而易见的：给用户提供便利的易使用界面。如果这样一个系统可得到全球万维网终端的访问，那么买票的任务就会分散开来，而且完成起来也更容易。

HTML是一种基于文本的标记语言。从理论上讲，HTML用来定义内容并构造内容的结构，而级联格式表（CSS）用来定位并格式化内容。当然，由于向后兼容性和客户软件的多样性，选用CSS来定位内容不是最佳的。此外，由于HTML的静态特性（也就是说它只是一种简单的文本语言），因此在想让内容变化和演化时，它本身有一定的局限性。

HTML提供了与各种万维网客户软件共享内容的一种优秀手段，但它也有几个缺点。当向万维网服务器请求一个HTML文档时，万维网服务器仅给请求者返回这个HTML文档。因此，这只是出版内容的方法，而决不是创建、控制、组织或自定义内容的方法。HTML就像它当今被使用的那样，往往把焦点放在内容的可视化品质上，而对内容的具体结构却漠不关心。

服务器端与客户端

HTML是一项客户端技术，因而意味着HTML文档完全由客户软件来处理。万维网服务器并不根据HTML文档内所含有的代码来采取相应的操作。万维网服务器只是提供客户软件所请求的文件，然后由客户浏览器来做出如何再现它们的决定。HTML不是一种编程语言，它不给任何类型的数据处理提供任何结构。

相反，PHP是完全服务器端的。当PHP脚本执行时，它不直接与浏览器交互，只是PHP脚本的最终产品（通常是HTML文档）才由发出请求的浏览器来处理。如果浏览器接收到一个未处理过的PHP脚本，它将尽力把该PHP脚本再现为正常的HTML。浏览器执行不了PHP脚本。

HTML是万维网应用程序开发的一个完整部件。PHP代码可以直接嵌入和混合在HTML中。当客户软件向万维网服务器请求一个HTML文档时，万维网服务器通过直接给客户软件发送该文档来作为响应。图1.1显示了一个客户软件请求一个HTML文档时服务器应该如何做出响应。

请求PHP脚本的方式却完全不同。在文档发送给客户软件以前，该文档先由PHP处理。而且PHP引擎执行它在该文档中所发现的任何PHP代码。图1.2说明了一个请求PHP脚本的客户端请求，且这个PHP脚本返回一个已处理过的HTML文档。

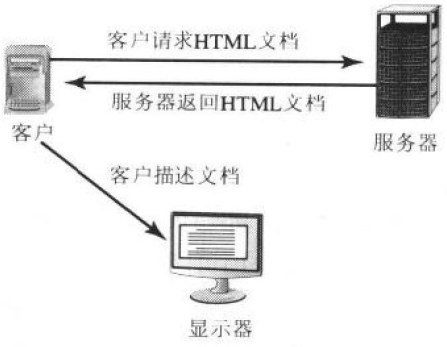

图1.1 HTML文档请求

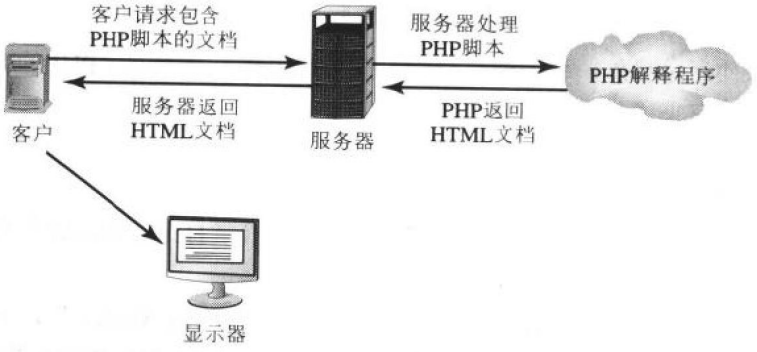

图1.2 PHP脚本请求

这两个过程之间显示了PHP与HTML之间的差别：PHP完全在服务器端执行，并且是一种崭新的编程语言；HTML只用来出版超文本，并在客户端处理。

了解PHP

每当使用一种具体的编程语言设计应用程序时，关键是要了解该编程环境的全部能力和局限性。万维网开发也是这样，没有任何差别。本节将讨论如何使用PHP给静态HTML赋予生命的活力。这一初级探讨将为大家学习本书后面的所有内容打下坚实的基础。在开始探讨万维网开发与可以使用的环境以前，我们先来编写自己的第一个PHP脚本。我们还将给大家介绍变量，以及PHP和动态内容的一些基础知识。

注意： 有关安装与配置PHP的详细信息，请参见附录A。

第一个PHP脚本

在PHP脚本开始执行以前，必须先命令服务器去执行它。为此，必须用一组特殊的标注把脚本块括起来，这组标注的作用是让服务器知道PHP代码是什么。当服务器遇到一个PHP开始标注时，该PHP开始标注与结束标注之间的任何东西将作为PHP来执行。PHP开始标注如下所示：

```
<?php
```

而结束标注则是：

```
?>
```

编程人员也可以使用<? ... ?>来替代，但<?php开始标注是首选方法，因为这允许PHP嵌套在支持XML的代码内。一般情况下，上述两种方法的任何一种都是允许的，而且使用哪种方法主要取决于个人的偏好。

另一种方法也会打开和结束PHP脚本：即启用php.ini文件中的相应选项，要想启用这些选项，必须修改php.ini文件。配置asp_tags选项可以使用ASP格式的脚本块（利用<% ... %>）来打开与结束一个脚本块。在php.ini文件中，找到带有asp_tags的那一行，并把它从asp_tags = Off改成asp_tags = On。是否使用这些ASP标注格式也主要取决于个人的偏好和编程风格。

注意：<?php ... ?>方法是推荐方法，而且也是本书中将要使用的方法。

如果大家使用过另外一种编程语言，尤其是使用过一种类似于PHP的脚本描述语言，接下来的几节对大家来说将是非常基本的复习材料。

“Hello, World!”程序

只要是介绍程序设计语言的图书，书中就会有一个“Hello, World!”程序。一般说来，它是学习任何一种程序设计语言的第一个程序。

我们将编写一个最小的PHP脚本，该脚本将生成“Hello, World!”，并把它发送给浏览器。这个脚本将举例说明开始与结束标注和一个生成输出结果的PHP结构。

创建一个标题为hello_world.php的文档，把清单1.1中的代码输入到该文档中，把该文档保存在万维网服务器的文档根中，然后在浏览器中导航到该文档。

- 在默认的Red Hat配置上，文档根可以在/var/www/html中找到。
- 如果大家像附录A所描述的那样安装了PHP和Apache，文档根将是/usr/local/apache/htdocs。然后，包含该文件的完整路径将是/usr/local/apache/htdocs/hello_world.php。

提示：为了方便起见，本书中的所有演示脚本都放在了选配光盘上。大家可以用万维网浏览器来检查这些文件，以便查看它们的结果；大家也可以像我们建立代码一样，用文本处理程序来打开它们。

清单1.1 未加翻译的PHP网页（hello_world.php）

```
<html>
<head>
     <title>Hello, World!</title>
</head>
<body>
     <?php print('Hello, World!\n'); ?>
</body>
</html>
```

清单1.1中的脚本从表面上看极其简单，而且简单得会让人产生误解。执行这段代码时发生了相当多的事情。请注意那些常见的HTML实体——html、head、title和body标注。但

此为试读，需要完整PDF请访问：www.ertongbook.com