

PHP 应用技巧 与实例

数字时代工作室 编著



PHP 应用技巧与实例

数字时代工作室 编著

人 民 邮 电 出 版 社

图书在版编目(CIP)数据

PHP 应用技巧与实例/数字时代工作室编著. —北京: 人民邮电出版社, 2001.7
ISBN 7-115-09412-8

I. P... II. 数... III. PHP 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 036959 号

内 容 提 要

本书主要讲述 PHP 4.0 的安装、配置、语法以及使用 PHP 4.0 创建网络的高级应用技巧。将开发工具与实例结合起来, 偏重于应用层次。本书试从专业开发人员的角度介绍使用 PHP 4.0 开发网站的方式, 以使读者获得较高的起点, 能够学以致用, 迅速掌握比较实用的网站开发技巧。

本书所面对的对象主要是网络爱好者和广大网站开发人员, 适用面很广。对相关专业的
大专院校师生、网页设计培训班学员也有一定的参考价值。

PHP 应用技巧与实例

-
- ◆ 编 著 数字时代工作室
责任编辑 张瑞喜 姚予疆
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ pptph.com.cn
网址 <http://www.pptph.com.cn>
读者热线: 010-67129212 010-67129211(传真)
北京汉魂图文设计有限公司制作
北京密云春雷印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787×1092 1/16
印张: 21.5
字数: 515 千字 2001 年 7 月第 1 版
印数: 1-5 000 册 2001 年 7 月北京第 1 次印刷
ISBN 7-115-09412-8/TP·2299
-

定价: 38.00 元(附光盘)

本书如有印装质量问题, 请与本社联系 电话: (010)67129223

前 言

PHP 4.0 自 2000 年 5 月 20 日正式推出以来,市场上还较少针对 PHP 4.0 的完整介绍。PHP 4.0 相比 PHP 以前版本,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,因此更为有效、可靠。其在大多数情况下运行比 PHP 3.0 快,脚本描述更强大并且更复杂。PHP 4.0 在 3.0 版的基础上增加或增强了 30 多种有用的特征,主要如下: COM/DCOM 支持、加密支持、FTP 支持、ISAPI 支持、扩充 API 模块、自动资源释放、类型检查、“here”打印等。

另外,即使是现在市面上罕有的几本介绍 PHP 3.0 的图书,绝大部分也只是对 PHP 语言的泛泛之谈,对于经常使用的部分函数调用也讲解得不是很详细,并且给出的范例典型性不强,不是很适合于制作具有一定专业水准的网站。

因此,这本有关 PHP 4.0 的书,一方面,力求能用简短的篇幅,讲述网页的一些制作技术,这一部分适合于初学者;另一方面将详细介绍 PHP 4.0 的用法,包括环境的安装、PHP 语法介绍、常用函数讲解、范例讲解;最后将大致描述一个具有一定商业性,专业级网站所需的各个部分,如用户管理、安全认证、电子商务系统的设计过程和制作要点。做到既适合初学者,更侧重于专业人员。

特色:既讲 UNIX 系统下的安装与设置,也讲 Windows 下的配置问题。既介绍 HTML、PHP 的基本语法,更侧重高水准和实用性。书中将讲述几个常用的 PHP 应用实例,如计数器、留言版、多用户论坛、用户数据库管理、安全认证、电子商务等。

参与本书编写工作的人员主要有许辉耀、张岩、倪楠、陈晓亚和于纲等。由于作者本身的水平有限,再加上创作时间仓促,本书难免存在疏漏和错误之处,衷心希望各界专家和读者朋友不吝赐教。

数字时代工作室

目 录

第 1 章	PHP 简介	1
1.1	PHP 的起源和发展	2
1.2	PHP 的优点	3
1.3	PHP 的功能	5
1.4	PHP 与其他动态网页语言的比较	6
第 2 章	PHP 开发环境设置	9
2.1	工作环境说明	10
2.2	在 Windows 下的安装	10
2.2.1	PHP 的安装	11
2.2.2	Apache 服务器的安装	12
2.2.3	在 Windows 9x/Me 中运行 Apache	16
2.2.4	在 Windows NT/2000 下运行 Apache	17
2.2.5	ToolKit For Apache 的安装	18
2.2.6	MySQL 服务器的安装	21
2.2.7	在 Windows NT/2000 下运行 MySQL	24
2.3	在 Linux 下的安装	25
2.3.1	Linux 的一些基本概念和操作	25
2.3.2	编译前的准备工作	26
2.3.3	安装 MySQL 服务器	27
2.3.4	测试 MySQL	28
2.3.5	Apache 及 PHP 的安装	30
2.3.6	Apache 及 PHP 的配置	31
2.4	Zend Optimizer 的安装	33
2.4.1	系统需求	33
2.4.2	安装过程	34
2.4.3	运行检测	34
2.5	PHP 编辑器介绍	35
第 3 章	PHP 轻松入门	37
3.1	PHP 程序分析	38
3.1.1	复活节彩蛋	38
3.1.2	书写规则	39
3.1.3	从 HTML 中脱离	39
3.2	输出语句	41



3.2.1	字段宽度	42
3.2.2	对齐方式	42
3.2.3	普通字符	43
3.3	程序注释	44
第 4 章	数据类型	45
4.1	数据类型	46
4.2	常量	46
4.2.1	整数型常量	46
4.2.2	浮点数常量	47
4.2.3	字符串常量	47
4.2.4	符号常量	49
4.3	变量类型	50
4.3.1	整型变量	50
4.3.2	浮点型变量	50
4.3.3	字符串变量	51
4.3.4	数组变量	51
4.3.5	多维数组	52
4.3.6	对象变量	53
4.3.7	变量替换	53
4.3.8	变化的变量	54
4.4	变量类型转化	54
4.4.1	自动转化	54
4.4.2	变量类型的确定	55
4.4.3	类型强制转化	55
4.4.4	字符串类型的转化	55
4.5	运算符号	56
4.5.1	运算符的优先级	56
4.5.2	算术运算	57
4.5.3	字符串合并运算	58
4.5.4	赋值运算	59
4.5.5	位运算	59
4.5.6	逻辑运算	60
4.5.7	其他运算符	61
第 5 章	程序控制	63
5.1	表达式	64
5.2	复合语句	65
5.3	if 语句	66
5.4	switch 语句	68



5.5	for 语句	69
5.6	foreach 语句	70
5.7	while 语句	71
5.8	其他流程控制语句	72
5.8.1	break 语句	72
5.8.2	continue 语句	73
5.9	程序外调用语句	74
5.9.1	require()语句	74
5.9.2	include()语句	75
第 6 章	PHP 中的函数	79
6.1	函数的定义和调用	80
6.1.1	函数的定义	80
6.1.2	函数的调用	80
6.1.3	函数的返回值	81
6.2	参数传递	82
6.2.1	形式参数	82
6.2.2	变量参数	83
6.2.3	数组参数	83
6.2.4	默认参数	84
6.3	变量的活动范围	85
6.3.1	全域变量	85
6.3.2	静态变量	87
6.4	嵌套函数	88
6.5	递归函数	89
第 7 章	面向对象编程	91
7.1	面向对象的基本概念	92
7.1.1	对象	92
7.1.2	类	92
7.1.3	封装	93
7.1.4	继承	93
7.1.5	多态性	93
7.2	PHP 与对象	94
7.2.1	类的定义和使用	94
7.2.2	构造函数	95
7.2.3	继承	98
7.2.4	抽象类的定义和实现	99
7.2.5	重载的实现	101
7.2.6	多态性的实现	102



第 8 章 HTML 表单	105
8.1 <FORM>标识对	106
8.1.1 METHOD 属性	106
8.1.2 ACTION 属性	106
8.1.3 ENCTYPE 属性	106
8.2 <INPUT>标识对	107
8.2.1 TYPE 属性	107
8.2.2 NAME 属性	107
8.2.3 VALUE 属性	108
8.2.4 SIZE 属性	108
8.2.5 MAXLENGTH 属性	108
8.2.6 CHECKED 属性	108
8.2.7 SRC 属性	108
8.3 文本输入框	108
8.4 隐藏框	109
8.5 单选框	109
8.6 复选框	110
8.7 多行文本框	110
8.8 列表选择	111
8.9 图像按钮	111
第 9 章 应用实例	113
9.1 页面欢迎词	114
9.2 计数器	115
9.3 文件上传	116
9.4 反馈表单	122
9.5 在线人数的统计	123
9.6 站内搜索引擎	124
第 10 章 PHP 中的图形处理	127
10.1 使用说明	128
10.2 图形处理函数	128
10.3 创建动态图像的基本步骤	134
10.4 图形应用实例	136
10.4.1 图形计数器	136
10.4.2 图形日历的创建	138
10.4.3 投票统计	141
10.4.4 生成缩略图	143
10.4.5 取得图形信息	145
第 11 章 模式匹配	147





11.1 正则表达式	148
11.1.1 转义字符	148
11.1.2 方括号表达式	149
11.2 POSIX 风格的匹配函数	150
11.2.1 匹配函数 <code>Ereg()</code>	150
11.2.2 替换函数 <code>Ereg_replace()</code>	151
11.2.3 分割函数 <code>split()</code>	151
11.2.4 合并函数 <code>join()</code>	152
11.3 基于 PERL 风格的模式匹配	152
11.3.1 模式定界符	152
11.3.2 模式选项	153
11.3.3 PERL 定义的其他转义字符	154
11.4 PERL 风格的匹配函数	155
11.4.1 匹配函数 <code>preg_match()</code>	155
11.4.2 匹配函数 <code>preg_match_all()</code>	155
11.4.3 替换函数 <code>preg_replace()</code>	157
11.4.4 分割函数 <code>preg_split()</code>	157
第 12 章 MySQL 入门	159
12.1 MySQL 的数据类型	160
12.1.1 存储空间位数	163
12.1.2 数字类型	165
12.1.3 字符串类型	165
12.1.4 关于 NULL	167
12.2 MySQL 的命名规则	167
12.2.1 大小写敏感	169
12.2.2 MySQL 中的保留字	169
12.3 MySQL 中的自定义变量	171
12.4 MySQL 中的注释语法	171
12.5 MySQL 中的 SQL 语句	172
12.5.1 数据库创建语句	172
12.5.2 数据库选择语句	172
12.5.3 数据库删除语句	173
12.5.4 数据表创建语句	173
12.5.5 数据表更改语句	175
12.5.6 数据表删除语句	176
12.5.7 插入语句	176
12.5.8 查询语句	177
12.5.9 更新语句	178



12.5.10	删除语句	179
12.5.11	状态显示语句	179
12.5.12	状态描述语句	180
12.5.13	加锁/解锁语句	180
12.5.14	选项设置语句	181
第 13 章	PHP 与 MySQL	183
13.1	phpMyAdmin 介绍	184
13.1.1	phpMyAdmin 的安装和配置	184
13.1.2	phpMyAdmin 的保护	184
13.1.3	phpMyAdmin 的应用	187
13.2	PHP 中的 MySQL 操作函数	188
13.2.1	服务器连接函数	188
13.2.2	服务器关闭函数	189
13.2.3	数据库选择函数	189
13.2.4	数据库查询函数	190
13.2.5	数据记录操作函数	191
13.2.6	数据库级操作函数	193
13.2.7	数据库信息函数	193
13.3	PHP 数据库操作的一般步骤	196
第 14 章	Cookie 的使用	197
14.1	Cookie 概况	198
14.2	Cookie 的工作原理	199
14.3	在 PHP 中使用 Cookie	199
14.3.1	Cookie 的设置	200
14.3.2	Cookie 的处理	201
14.3.3	Cookie 的删除	201
14.3.4	Cookie 的限制	201
14.4	Cookie 的应用	201
第 15 章	Session 技术	203
15.1	Session 概述	204
15.2	Session 的工作原理	204
15.3	使用 PHPLib 实现 Session 功能	205
15.3.1	环境说明	205
15.3.2	安装过程	205
15.3.3	PHPLib 的工作原理	206
15.4	PHP 4.0 中的 Session	208
15.4.1	PHP.ini 中的 Session 配置选项	208
15.4.2	PHP 4.0 中 Session 相关函数	209



15.5	Session 的扩展	211
15.5.1	Session 补充函数	211
15.5.2	Session 扩展函数	213
15.5.3	MySQL Session 处理程序	213
15.5.4	MySQL Session 程序清单	215
15.5.5	DBM Session 处理程序	217
15.5.6	DBM Session 程序清单	218
15.5.7	测试程序	219
第 16 章	WWW 鉴权	223
16.1	HTTP 认证规范	224
16.2	HTACCESS 认证	225
16.2.1	Apache 指令介绍	225
16.2.2	允许有限个用户访问限制目录	226
16.2.3	使用组允许多个用户访问限制目录	228
16.3	MySQL 数据库认证	229
第 17 章	PHP 编译配置	231
第 18 章	PHP 配置选项说明	239
18.1	PHP 配置选项	240
18.2	和 Session 相关的配置值	246
第 19 章	Apache 配置文件介绍	249
19.1	Apache 配置文件	250
19.2	PHP 函数索引	259
第 20 章	用 PHP 进行网站规划	287
20.1	计划一个网站	288
20.1.1	欢迎页面的设计	288
20.1.2	联系信息页面	288
20.2	使用 PHP 进行网站规划	289
20.2.1	通用变量设置	289
20.2.2	通用页面设计	290
20.2.3	欢迎页面的设计	291
20.2.4	联系信息页面	292
20.3	规划和设计论坛	292
20.3.1	论坛安装说明	292
20.3.2	程序一览	293
20.3.3	数据表定义文件(table.txt)	293
20.3.4	样式表定义(bbs.css)	294
20.3.5	配置文件(config.inc.php)	295
20.3.6	通用页面程序(Page.inc.php)	296

20.3.7	用户注册页面(regstep1.php)	297
20.3.8	注册提交页面(regstep2.php)	300
20.3.9	用户登录页面(bbslogin.php)	302
20.3.10	登录认证页面(bbsauth.php)	304
20.3.11	论坛列表页面(bbslist.php)	304
20.3.12	列表显示函数(bbsshow.inc.php)	309
20.3.13	发表文章页面(bbsadd.php)	310
20.3.14	发表文章保存页面(bbssave.php)	314
20.3.15	发表文章保存程序(bbssave.inc.php)	316
20.3.16	文章内容显示页面(bbscontent.php)	318
20.3.17	文章回复页面(bbsreply.php)	320
20.3.18	文章回复保存页面(bbsreplysave.php)	326



第 1 章

PHP 简介

本章要点：

- PHP 的起源和发展
- PHP 的优点
- PHP 的功能
- PHP 与其他动态网页语言的比较

PHP 是一种服务器端 HTML 嵌入式的脚本语言, 已经成为流行的网页脚本语言之一, 它在保证最大可操作性的前提下, 提供了比 CGI 更快的执行速度。多平台特性使其可以无缝地运行在 UNIX 和 Windows 平台上。更为突出的是它对数据库的操作能力, 强大的兼容性使得 PHP 可以操作几乎所有的数据库, 并且在数据库操作的简便性上得到了绝大多数人的认可。

本章将介绍 PHP 的起源和发展、优点、功能及与其他动态网页语言的比较等方面, 通过一些关于 PHP 的基本情况, 使读者在使用 PHP 之前能够对 PHP 有一个整体的认识。

1.1 PHP 的起源和发展

PHP 是 Rasmus Lerdorf 在 1994 年秋天构思出来的。最早的未发行版本是用来与观看他的在线简历的人保持联系的, 当初起名为“个人主页工具”(Personal Home Page Tools), 这也是 PHP 缩写的由来。随着 PHP 影响的扩大, 人们又为它起了个很专业的解释: 超文本预处理器(PHP: Hypertext Preprocessor)。第一个交付用户使用的版本是在 1995 年初发行的, 即 PHPv1.0。当时它仅包括一个只能懂得很少几条宏指令的非常简单的分析引擎和一组用于主页信息反馈的工具(一个留言簿、一个计数器和一些其他的東西)。1995 年年中, Rasmus 重写了整个解析器, 并取名为 PHP/FI 2。FI 来源于他所写的另一个可以接受 HTML 表单数据的程序包, 组合了个人主页工具(Personal Home Page Tools)脚本和表单集成工具(Form Interpreter), 并且加入了对 mSQL 数据库的支持, 从此奠定 PHP 在动态网页开发上的影响力。此后 PHP/FI 便以一种令人惊奇的速度传播开来, 人们开始大量使用它编写程序。

我们无法确切地统计它的使用范围。但是根据估计, 到 1996 年底, 世界上已经有至少 15 000 个站点在使用 PHP/FI 2。到 1997 年底, 这个数字已经增长到超过 50 000 个站点, 而且 PHP 发展经历了一个重大的转折, 这就是 PHP 的开发从 Rasmus 个人的爱好升级到一群程序员们有组织的工作。再后来, 这个解析器被 Zeev Suraski 和 Andi Gutmans 重写, 通过这次全面的重写, 大量 PHP/FI 的功能被移植到 PHP3 中, 并且成为了 PHP3 的基本雏形。

在 1998 年, 已经有大量的商业化产品, 例如 C2 的 StrongHold Web server 和 RedHat Linux 捆绑了 PHP/FI 解析器或 PHP3 解析器。PHP 跟 Apache 服务器紧密结合的特性, 加上它不断地在进行更新并被加入新的功能, 而且几乎支持所有数据库以及高速的执行效率, 使得 PHP 的影响迅速扩大。根据当时 NetCraft 估计的保守数据, PHP 已经被世界上 150 000 个站点所采用。通过进一步分析, 我们知道其使用数量远比使用 Netscape 公司的企业级 Web 服务器的站点多。由于 PHP 的源代码完全是公开的, 所以不断地有新的函数库加入、不停地在更新, 使得它无论在 UNIX 或 Windows 的平台上都可以有更多的新功能, 并且使它在程序设计方面获得更好的支持。

在 2000 年 5 月, PHP 4.0 正式版发布了。到目前为止, 最新的版本为 PHP 4.0 RC3。PHP 4.0 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比 PHP 3.0 快, 其脚本描述也更强大并且更复杂。PHP 4.0.0 这些优异的性能是 PHP 脚本引擎重新设计产生的结果。PHP 4.0 脚本引擎即第四代 Zend 核心引擎使用了更有效的“编译——执行”范式, 而不是 PHP 3.0 所采用的“执行——当解析时”模型。无论用户接受与否, PHP 都将在 Web CGI



的领域上，掀起颠覆性的革命。无论是专业的网页设计人员，还是有兴趣在自己的个人主页上提供网页动态效果的众多网友，PHP 语言将是网页制作的必修课程之一。

1.2 PHP 的优点

学习和使用 PHP 有许多好处。PHP 虽然是个开放源码项目，没有什么商业支持，但是 PHP 和商业软件几乎没什么区别，PHP 在网上有众多的支持者并且不时有新的扩展模块出现，PHP 邮件列表的更新速度是相当快的。如果你有什么问题，只要在网上发个帖子，很快就能得到满意的答复。下面让我们总结一下 PHP 的优点：

➤ 学习过程更简单

PHP 的学习与 Java 和 Perl 有很大的不同，用户不必再像学习 Java 或 Perl 那样要花费数天时间苦学语法才可以写出一个像样的程序。只要了解一些基本的语法和语言特色，你就可以开始 PHP 编码之旅了。之后在编码过程中如果遇到什么麻烦，还可以再去翻阅相关文档。PHP 的语法与 C、Perl、ASP 或者 JSP 均有一定的类似之处。对于那些对上述之一的语言较熟悉的人来说，学习 PHP 将不再乏味。同样，如果你对 PHP 了解较多，会为学习其他语言打下良好的基础。

大多数用户只需要 30 分钟就可以将 PHP 的核心语言特点全部掌握，你可能已经非常了解 HTML，甚至已经知道怎样用编辑设计软件或者手工来制作好看的 Web 站点。由于 PHP 代码能够无障碍地添加进站点，所以在设计和维护站点的同时，可以很轻松地加入 PHP，使得你的站点更加具有动态特性。

➤ 开放源码并可免费使用

PHP 遵守 GNU 条约，任何人均可按条约免费使用并进行源码改写，使用者还可通过 PHP 的站点、邮件列表等方式获得支持。通过各种媒介，用户可以免费复制、编译、打印和分发所有的 PHP 源码、文档，可以无限制地发布自己编写的程序。用户的任何一个用 PHP 编写的程序都属于用户自己，可以按照自己的意愿进行处理，而不需要付任何版税。另外，网络上已专门开设了 PHP 的支持站点、代码交换站点，相当多的支持者们也开发出了许多强大的基库，让人们随意调用。

➤ 运行速度快

PHP 采用 HTML 内置标记技术，类似当前流行的 ASP，解释程序本身作为 Web 服务器的一个模块运行，极大地提高了运行时的解析速度，从页面表单提交的数据自动成为程序中同表单名的变量，而无需手工赋值。经测试表明，在 Web 站点访问量非常大时，PHP 的解析速度相当于传统 CGI 程序的 4 倍，因此非常适合应用在大中型站点。

➤ 可以嵌入 HTML

当使用者使用 C 或 Pascal 等程序设计语言编程时，所有的代码必须编译成一个可执行的



文件，编译成功之后才执行可执行文件进行结果验证。PHP 并不需要编译(至少不必要编译成可执行文件)。使用者可以把自己的代码混合到 HTML 中。结果的验证过程也很简单，只要通过浏览器来浏览该 PHP 页面就可以根据浏览器的显示结果来判断程序的正误。比如有下面的代码：

```
<HTML>
  <HEAD><TITLE>Test</TITLE></HEAD>
<BODY>
  <H1><? echo "hello world";?></H1>
</BODY>
</HTML>
```

当 Web 浏览器浏览 PHP 页面的时候，Web 服务器的 PHP 部分将在被调用时进行解释。Web 服务器在请求的 Web 页中寻找<? ... ?>标记，并按要求执行这些 PHP 代码。由 PHP 代码生成的输出将替代<? ... ?>标记。当上述 PHP 代码运行后，所有的 PHP 代码都消失了，仅仅留下了 HTML 语句。Web 页面将变成如下所示的内容：

```
<HTML>
  <HEAD><TITLE>Test</TITLE></HEAD>
<BODY>
  <H1>hello world</H1>
</BODY>
</HTML>
```

➤ 数据库功能强大

PHP 最强大和最重要的特征是它的数据库集成层，PHP 可以编译成具有与许多数据库相连接的函数。这使得利用它来完成一个含有数据库功能的网页将不再复杂。目前 PHP 支持的数据库主要有：

Adabas D、DBA、dBase、dbm、filePro、Informix、InterBase、mSQL、Microsoft SQL Server、MySQL、Solid、Sybase、ODBC、Oracle 8、Oracle、PostgreSQL 等。

PHP 还完全支持 ODBC 接口，这样的话，凡是支持 ODBC 接口的数据库，PHP 都可提供有力的支持，而且这些数据库的操作都是 PHP 内部包括的，无需其他附件介入，因此在实际应用中，可得到比任何后台技术都要快的数据库访问性能。在网上，PHP 与 MySQL 是绝佳的组合，使用这两个产品你可以轻而易举地实现你所需要的强大功能。你还可以自己编写外围的函数取间接存取数据库。通过这样的途径，当更换使用的数据库时，你可以轻松地更改编码以适应这样的变化。PHPLib 就是最常用的可以提供一般事务需要的一系列基库。

➤ 可扩展性极强

PHP 已经进入了一个高速发展的时期。对于一个非程序员来说为 PHP 扩展附加功能可能会比较难，但是对于一个 PHP 程序员来说并不困难。PHP 不但内置了对文件上传、密码认证、CookieS 操作、邮件收发、动态 GIF 生成等功能的支持，还极有远见地提供了对 GZIP 文件、





PDF、XML 的直接支持。用户还可以编写自己的扩展模块，或从网上下载别人编写的其他模块、库，给将来的扩展提供了极大的空间。

➤ 面向对象编程

PHP 提供了类和对象。基于 Web 的编程工作非常需要面向对象编程能力。PHP 支持构造器、提取类等。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，可以根据需要实现嵌套。PHP 4.0 还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。PHP 4.0 实现了对象重载，引用技术等新功能。对象重载语法允许第三方的基于面向对象的类库使用 PHP 4.0 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。

➤ 可伸缩性

传统网页的交互作用是通过 CGI 来实现的。CGI 程序的伸缩性不很理想，因为它为每一个正在运行的 CGI 程序开一个独立进程。解决方法就是将经常用来编写 CGI 程序的语言的解释器编译进 Web 服务器(比如 mod_perl、JSP)。PHP 就可以以这种方式安装，虽然很少有人愿意这样以 CGI 方式安装它。内嵌的 PHP 可以具有更高的可伸缩性。

除此之外，PHP 具有相当强的可移植性。用 PHP 写出来的 Web 后端 CGI 程序，可以很轻易地移植到不同的操作平台上。例如，以 Linux 为平台的网站，当系统负荷过高时，可以快速地将整个系统移到 SUN 工作站上而不用重新编译。另外，PHP 的开发者们为了更适合 Web 编程，开发了许多外围的流行基库。更好的是，你可以选择 PHP 安装版本的功能组建，PHP 可以做到你想让它做到的一切而且无所不能！

1.3 PHP 的功能

PHP 发展到现在，已经相当完善了，PHP 的函数库很丰富，对于源代码开发提供了多样的支持。主要有以下几点：

- ✧ 支持多种数据库
- ✧ HTTP 认证
- ✧ 文件上传支持
- ✧ HTTP cookie 支持
- ✧ 正则表达式格式
- ✧ 出错操作
- ✧ 英文拼写检查
- ✧ BC 高精度度计算
- ✧ 历法
- ✧ PDF 文档格式
- ✧ Hyperwave 服务器
- ✧ 图形处理

