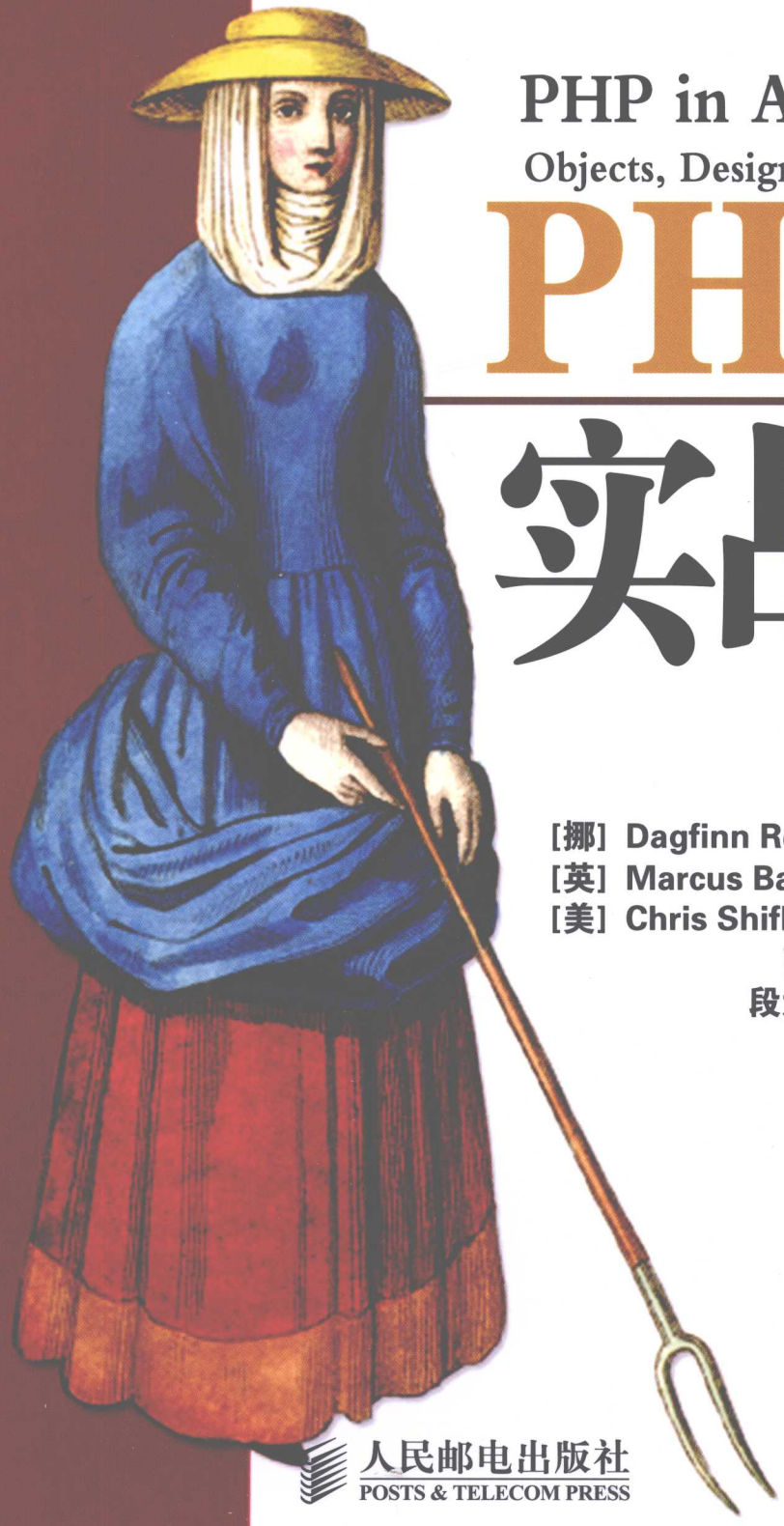
TURING 图灵程序设计丛书 Web开发系列

MANNING



PHP in Action

Objects, Design, Agility

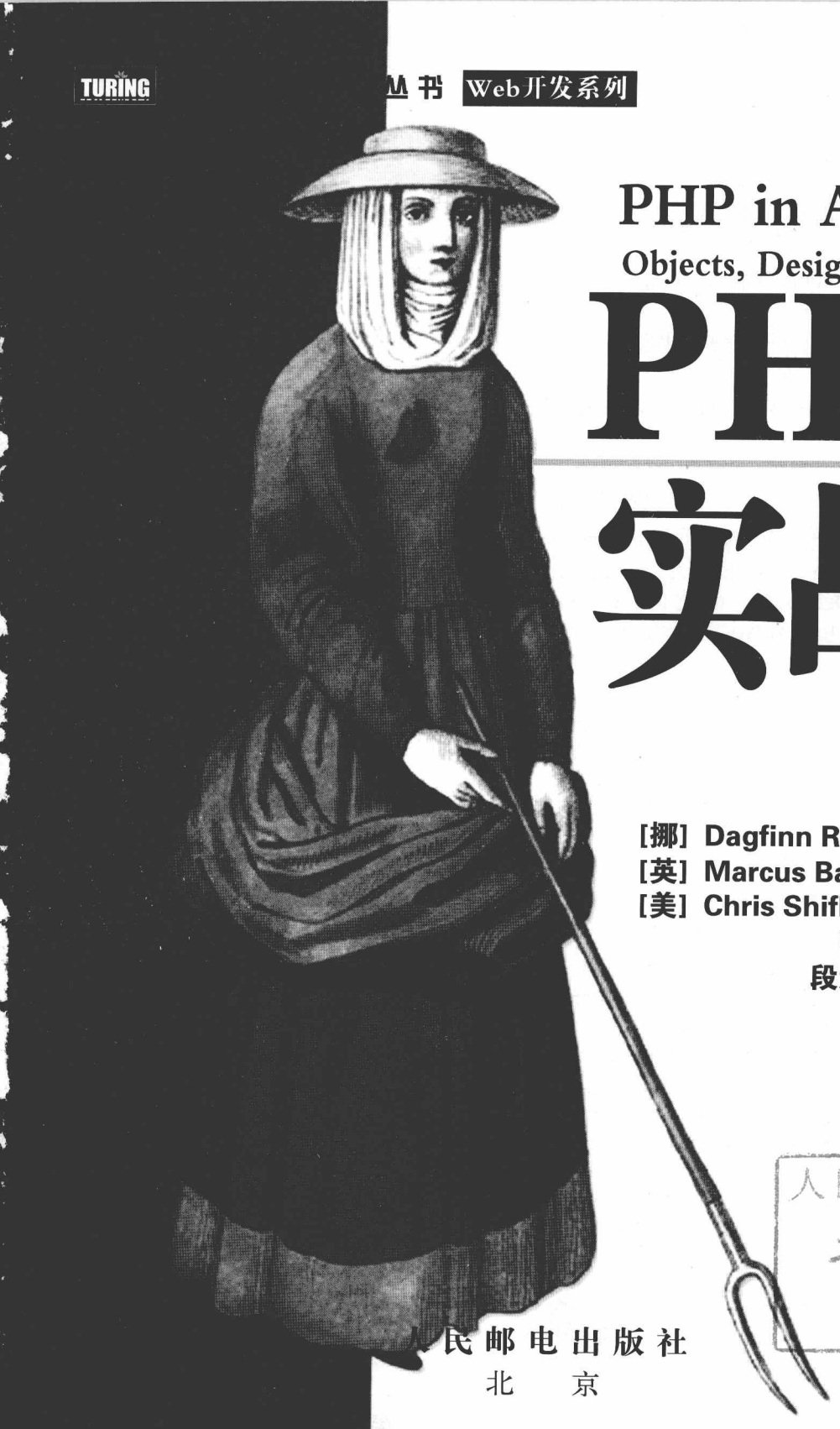
PHP 实战

[挪] Dagfinn Reiersøl
[英] Marcus Baker 著
[美] Chris Shiflett

张颖 等译
段大为 审校

人民邮电出版社
POSTS & TELECOM PRESS

TURING 丛书 Web开发系列



PHP in Action

Objects, Design, Agility

PHP 实战

[挪] Dagfinn Reiersøl
[英] Marcus Baker 著
[美] Chris Shiflett

张颖 等译
段大为 审校

人民邮电出版社
北京

图书在版编目（CIP）数据

PHP实战 /（挪）雷勒索（Reiersøl, D.），（英）贝克（Baker, M.），（美）史夫利特（Shiflett, C.）著；张颖等译. —北京：人民邮电出版社，2010.1
（图灵程序设计丛书）
书名原文：PHP in Action: Objects, Design, Agility
ISBN 978-7-115-21745-5

Ⅰ. ①P… Ⅱ. ①雷… ②贝… ③史… ④张… Ⅲ. ①PHP语言－程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字（2009）第206532号

内 容 提 要

随着PHP 5的发布和Zend框架项目的应用，PHP和敏捷思想、设计模式以及单元测试的联姻成为主流话题。本书首先介绍了面向对象设计的原则、模式和技巧，然后讲述了如何在PHP上利用其语法和特性，处理和实施这些原则、模式和技巧，并应用于Web编程中遇到的难题上。

本书适合所有使用PHP开发Web应用程序的人员阅读。

图灵程序设计丛书

PHP实战

◆ 著　[挪] Dagfinn Reiersøl　[英] Marcus Baker　[美] Chris Shiflett
译　张　颖　等
审　校　段大为
责任编辑　傅志红
执行编辑　陈兴璐

◆ 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@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京隆昌伟业印刷有限公司印刷

◆ 开本：800×1000　1/16
印张：27.75
字数：656千字　2010年1月第1版
印数：1－3 000册　2010年1月北京第1次印刷

著作权合同登记号　图字：01-2008-3296号

ISBN 978-7-115-21745-5

定价：69.00元

读者服务热线：(010)51095186　印装质量热线：(010)67129223

反盗版热线：(010)67171154

版 权 声 明

前　言

这本书的背后有我个人的一段故事。几年前，我意识到自己职业生涯中的一切努力和我的预期并不十分相符。虽然还没有严重到中年危机的地步，但这种意识促使我开始以新的方式来思考。

那时我正从事PHP的Web编程。在我所供职的公司里，我处于一种孤独的境地。于是我决定仔细审视一下自己的工作。我扪心自问："如何才能让自己有更卓越的表现？"一个想法就是，每天结束的时候都对当天的工作做出总结。我做的最成功的是什么？怎样才能取得更多成功？什么是不太成功的？如何才能减少不成功的事？

木桶的短板就是调试。很明显，它占用了我绝大多数的时间，如果有什么可以使调试更有效或者根本可以不要调试，那么我的工作会更高效。我开始尝试更早地捕捉bug。我试过防御型编程，但不是很成功。不久我偶然发现了敏捷过程和测试驱动开发、极限编程和重构。这些方法粗看上去像我和我的同事们这些年来一直的做法，但确实要好不少。我首先在自己的工作中运用了这套方法。当时，PHP社区中鲜有人做此尝试。我是比较早这样做的，我使用了2002年3月的PHPUnit最初的测试版实践测试先行。

有这本书的构想，是在我接手了一位程序员同行一些拙劣的PHP代码之后。我意识到代码可以改进、重构，并且可以系统地阐述它。我想这对于有些人一定是有用的。并且，PHP领域尚没有关于敏捷过程和测试驱动开发的图书。

一件事情意外地加速了这个写书的项目：我被解雇了。（几个月后，我成了那间解雇我的公司的董事会成员，这是另一个故事。）我花了将近三年的时间完成本书。要让审阅者对它十分满意不是件易事，很多部分我都重写了好几次。Marcus Baker和Chris Shiflett在将近结尾处参与了进来。同时，PHP和敏捷思想、设计模式和单元测试的联姻成为本书的主流话题。这个过程中最重要的官方事件是PHP 5的发布和Zend框架项目的启动。

整个过程当中我学到很多东西，最重要的是，如果想自己写一本书，那就一定要多读书。我相信深层理解的重要性，不只是知道很多的细节，而是对每个细节都有深入的理解。我也相信，这来源于扎实的基础和从多方面理解问题的能力。

这一切都引领我不断重新审视那些最基本的问题。我不断地问一些看起来有些傻的问题。在网络论坛中，即使是在讨论我非常熟悉的问题时，我也经常被误认为是初学者。我相信，我自己的理解越深，我就能为别人把这个问题解释得更好。我希望这种探索对你们来说也是有帮助的。

Dagfinn Reiersøl

致　谢

本书的写作得到了我的一些朋友和对手的帮助。

我先解释一下所谓的对手，我用那个词只是表明一种看法。他们不是坏人，也不是故意跟我过不去（希望如此）。但是有几位的确让我的日子有点不好过，让我不得不去做一些我本不打算做的事情，而结果却提高了本书的质量。对此我表示感激，但在此我不提他们的姓名。

朋友方面，感谢我的妻子Regine和我的女儿Maria，感谢你们的爱、支持和鞭策。感谢我的儿子Jacob（已经6岁了）那懵懵懂懂的热情和令人忍俊不禁的见解，这在书中的一些地方得到了体现。

更实际地讲，最重要的贡献来自于合著者：我的好朋友Marcus Baker，我还从未见过他；以及Chris Shiflett，他在百忙之中抽出时间为本书撰写了有关安全的内容。

和其他Manning的作者一样，我对整个出版团队和他们对质量的严格要求印象十分深刻。他们知道并且也做到了如何将一本书的可读性和趣味性提升到更高的高度。可能我有些自夸，但我对这个结果确实非常满意，每一次我重读书中的章节，都会觉得是一种享受。

审阅的过程令人疲劳但是却非常重要。出版人Marjan Bace有一种特殊的能力和决心来处理哪怕是最难以令人兴奋的反馈，即使反馈意见非常不具体，他都能从中提炼出一些有用的东西。

感谢这些在百忙之中抽时间阅读了不同阶段原稿的审校者们：Richard Lynch, Andrew Grothe, Kieran Mathieson, Jochem Maas, Max Belushkin, Dan McCullough, Frank Jania, Jay Blanchard, Philip Hallstrom, Robin Vickery, David Hanson, Robbert van Andel, Jeremy Ashcraft, Anthony Topper, Wahid Sadik, Nick Heudecker及Robert D. McGovern。特别是Mark Monster，他在本书出版之前又再一次通读了全书，确保了技术上的准确性。

另一位间接的贡献者是我长期的同事和朋友Per Einar Arnstad。从我们充满创意的讨论和互动中得来的思想，是我研究软件的基石，他的创业精神也激励我勇于承担这项风险，使本书的出版成为可能。

还感谢另一位同事Tarjei Huse，在阅读了原稿之后，他给予了我几乎是最全面的、充满智慧的反馈。

最后，特别感谢妻子Kathrine Breistøl，她答应我，如果我的财务状况糟糕到了一定程度，就会全面清理厨房里所有瓶子去卖钱。当然，我从来也没要求她兑现过。

关 于 本 书

本书从某种意义上来说有“一女二嫁”之嫌。它首先介绍了面向对象设计的最新原则、模式和技巧，然后又将这位“新人”许配给两位“新郎官”，第一位是PHP编程语言，另一位是PHP程序员每天的日常工作。

更具体地讲，本书就是介绍如何利用PHP的语法和特性，设计和实现这些原则、模式和技巧的。本书同时也将这些原则、模式和技巧应用于Web编程中遇到的各种具体和常见的难题上。

读者对象

本书适合使用PHP开发应用程序的程序员，还有希望学习最新面向对象实践、原则和技巧并将其应用于Web编程的日常问题的读者。

这不是一本关于PHP的入门书籍，它假设你有最基本的PHP知识，或者有其他编程语言的经验，而且对Web编程基本概念及可能遇到问题有所了解。

内容结构

本书分为四个部分。第一部分和第二部分介绍开始提到的原则、模式和技巧，并且展示如何在PHP中实现它们。第一部分引入并展开讨论面向对象编程和设计的主题。第二部分将介绍单元测试和重构。第三部分和第四部分将前两部分的素材应用到日常Web编程中可能遇到的问题上。第三部分关于Web界面，而第四部分解决数据库和数据存储的问题。

第一部分：工具和概念

第一部分中的几章循序渐进，从PHP面向对象编程的具体细节讲到面向对象应用程序设计这个更抽象的主题。

第1章介绍并讨论了PHP和敏捷实践的利弊。

第2章和第3章解释PHP面向对象编程的机制和语法。尽管对象和类是难以分割的东西，但是第2章主要关注对象的特性，而第3章则重点关注类。

第4章讨论了对象和类的概念好在哪里，它们如何与现实世界产生联系，以及如何区分面向对象设计的优劣。

第5章讨论最基本的类的关系——继承、关联和组合，以及程序设计中接口的作用。

第6章开始我们正式探索面向对象设计，将面向对象理论作为设计的指导思想。

第7章介绍设计模式——再现常见设计问题的解决方案，并且描述一些最常见的问题。

第8章介绍设计原则和模式在处理日期和时间时的另一种应用。

第二部分：测试和重构

第二部分的重点是测试和重构（即改善现有代码的设计），将从质量保证和学习过程这两个方面进行探讨。

第9章介绍了单元测试和测试驱动开发，使用了数据库事务处理（transaction）类作为实例。

第10章在单元测试领域进行了深入的挖掘，展示了如何正确配置测试，并且使用模拟对象和其他替代品来简化测试。这一章以第9章为基础，通过事务处理类创建了一个联系人管理器。

第11章是关于重构的，并且重点关注Web应用程序。这一章既解决传统意义上面向对象重构的问题，也针对如何将设计不良的过程化代码转化为更好管理的状态。

第12章将注意力从单元测试转向Web测试，结束了对测试这个主题的讨论。再一次以联系人管理器为例，展示了如何确认用户界面是否符合用户期望，以及如何自顶向下地设计出整个Web应用程序。

第三部分：构建Web界面

第三部分专注于Web编程的重要方面：Web界面。

第13章解释了从程序代码中分离HTML的原则，并且描述了如何通过使用模板引擎和具体的实现技巧。

第14章解决的问题是将很多分散的组件组装成网页，同时介绍了如何实现复合视图设计模式。

第15章介绍了用户交互和MVC（模型-视图-控制器）模式。

第16章将会教给你如何把MVC模式运用于Web开发上面，包括实现页面控制器和前端控制器。

第17章深入讲解了服务器端和客户端输入验证以及如何实现验证同步。

第18章展示了如何利用PEAR中的HTML_QuickForm包开发表单处理。

第四部分：数据库和基础结构

第四部分解决的是从面向对象的角度来处理数据库和数据存储的问题。

第19章分两个不同的部分。一部分讲述如何在面向对象程序里适当地处理数据库连接以及如何处理数据库连接所需要的设置。另一部分讲述数据库抽象，即如何使代码独立于具体的数据库管理系统。

第20章面对的问题，源自我们必须使用一门完全独立的编程语言（SQL）来查询数据库。这一章讲述了如何封装、隐藏以及生成SQL代码。

第21章将前两章的内容集中在一起，形成一个比较完整的面向对象数据存取的设计模式。

附录

附录A给出一些关于测试和测试工具的具体信息，这些不太适合在测试章节中介绍，所以在这里给出，同时也包括了SimpleTest和PHPUnit API核心部分的参考资料。

附录B介绍了PHP的安全问题。

使用本书

本书各章相对独立，可以从任何一章开始阅读，而不必一定要读过之前的章节。除非你对面向对象编程和设计已经得心应手，否则的话，应首先读完第一部分，这有助于你对第三部分和第

四部分有更容易、更深层、更透彻的理解。但在随后的章节里，所有例子的工作原理都会被详细地解释。这些例子使第一部分阐明的概念更加易懂，但是一般并不依赖这些概念。

但是，某些部分里的一些章彼此有较强的联系。例如，如果没有理解第9章和第10章里面介绍的单元测试的基本概念，那么就很难读懂第11章中关于重构的例子。

示例代码

本书正文和代码清单中的源代码都用代码体。很多代码清单都带有注解以阐明重要概念。某些情况下，代码清单中会有编号，对应于正文中的注解。

关于书名

*In Action*系列丛书通过结合介绍、概览和how-to示例来帮助人们学习和记忆。根据对认知科学的研究，人们记住的东西正是他们在自我激励的探索中所发现的东西。

尽管在Manning出版社工作的没有认知科学家，但我们很肯定一点，即要想让学过的东西记得牢固，就必须经历探索、行动和重复所学这样几个阶段。人们只有积极地探索过，才能理解和记住新东西，也即掌握新东西。人们是在实践中学习。*In Action*指南的一个要点是它是示例驱动的。它鼓励读者试着去实现、去实践新代码和探索新想法。

本书这样命名还有一个更现实的原因：我们的读者很忙。他们读书是为了完成工作或解决问题。他们需要能够易于进退的图书，并且恰好在他们正需要的时候学到最需要的东西。他们需要能帮他们进行实践的书。本系列的这套丛书正是为这样的读者设计的。

关于封面插图

本书封面上的法国农妇在法语里叫做“paysanne”。这个插图取自1805年版Sylvain Maréchal关于不同地区着装风格的四卷纲要。该书于1788年在巴黎首印，正是法国大革命爆发的前一年。书中的每幅画都是手绘的。

Maréchal收集的插图多种多样，生动地展现了200年前世界各城镇和州/省的独特风情。当时，距离数十英里的两个地区的人，通过不同的着装风格即可区分开来。这些图画生动地描述了那个时期以及除了充满活力的现代以外所有其他历史时期，人们生活的疏离感和距离感。

着装风格改变了，而当时地区性的多样性也渐渐消融了。如今，很难分辨出人们是哪个洲的居民。也许乐观地看，我们经历这些文化和视觉的多样性是为了拥有更加多样化的个人生活，或者是为了拥有更加多样和有趣的知识和技术生活。

我们这些在Manning出版社工作的人盛赞这样的创造性、主动性，当然还有这些图书封面带给计算机事业的乐趣，这些图书封面基于两个世纪前丰富的地区多样性，其中的图片把我们带入到那时的生活中！

目　录

第一部分　工具和概念

第三部分 构建 Web 界面

Part 1 第一部分

工具和概念

当你要完成某项工作时，总是自然而然会在开始时寻找需要的工具。在面向对象编程的世界中，工具和概念之间的区别非常模糊。有描述和实现概念关系的工具，也有作为设计过程工具的概念。

本书第一部分将介绍这些工具和概念，其中多数属于面向对象编程和应用程序设计的范畴。在应对第三部分和第四部分的Web编程挑战时，将应用这些工具和概念。我们将介绍PHP中对象和类的语法，为何以及如何使用它们，还有如何使用设计模式和面向对象原则。

本部分内容

第 1 章 PHP与现代软件开发

本章内容

1.1 PHP的作用

1.2 语言、原则和模式

1.3 小结

一幅漫画中，有个穿医生制服的人正在打电话："是的，琼斯先生，针灸会暂时起到一定作用。任何庸医的治疗都可能会暂时起作用。但是只有科学的医疗实践才能让一个人永生。"[①]

这段荒谬而自大的话语显然不能让病人信服。可是，如果我们忽略那些牵强的细节，会发现通过这个虚构的医生的故事，至少说明了一个重要问题——谨记长期目标的重要性。

医疗的长期好处与本书的主题相去甚远，但从软件的长期发展来看，则是另一回事了。现代软件工程并不想让软件永远可用，但生产力长期发展却是新技术、原则和方法论发展的关键问题之一。这就是为什么面向对象编程是今天的实际（de facto）标准：这种方式可以让软件从推出第一版之后的后续各版本更易于维护和开发。其他专业术语，如设计模式（design pattern）和敏捷开发（agile development），也都与此相关。

PHP 5是一次新的尝试，力图通过PHP（"PHP: Hypertext Processor"的递归缩写）更轻松地运用这些概念和方法。

本书将以此为着眼点，揭示这次尝试带来的新变化。我们将围绕三个互相关联的目标展开讨论。

- 工具包的探索和最佳利用。用现代方法和工具将开发技巧提升到一个新水平。
- 提供全面介绍。从用户界面到数据库交互，在Web编程的各个方面应用该工具包。
- 保持其简单性。遵循爱因斯坦的建议——凡事应该尽可能简单，而又不过于简单。

无论出于什么原因使用PHP（有可能像我一样是出于偶然才使用的），了解PHP的长处总会有所帮助，而且更有助于了解如何克服它的局限。出于此目的，本章开始将讨论PHP自身的一些利弊，然后介绍现代的面向对象方法和敏捷方法，了解它们与PHP的关联。

1.1 PHP 的作用

PHP一直是主要用于Web编程的一种语言，现在仍是这样。而且，随着PHP 5（或者PHP 6，也许你在阅读本书时已经发布这个版本了）的发布，它带来了最新的特性，成为一种与现代面向

① 这段情景来自于我的记忆——几年前在同事的办公室看过这幅漫画，后来就再没见过。