源代码下载

书中所有源代码，都可以从 http://www.khp.com.cn 网站的“下载服务”专区或本书子页面下载。

深入浅出

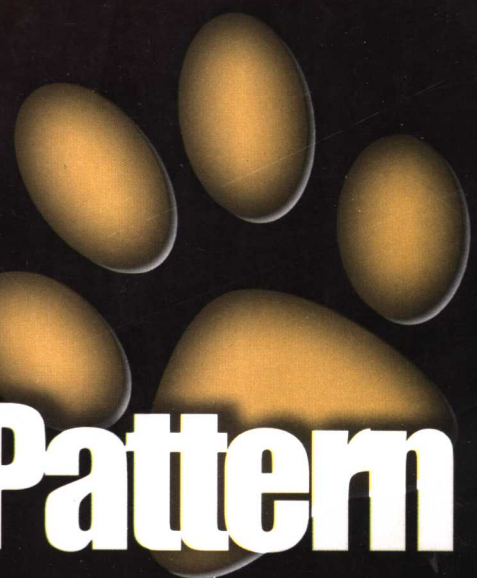
设计模式

(C#/Java 版)

架设从理论到实践应用的桥梁

- 23 种常用 GoF 设计模式
- GRASP 通用责任分配软件模式
- C# 和 Java 两种编程语言实现
- 现实生活范例、示意性图片说明
- 两个设计模式综合应用案例
- 25 道自测题及答案解析

莫勇腾 编著



Design Pattern

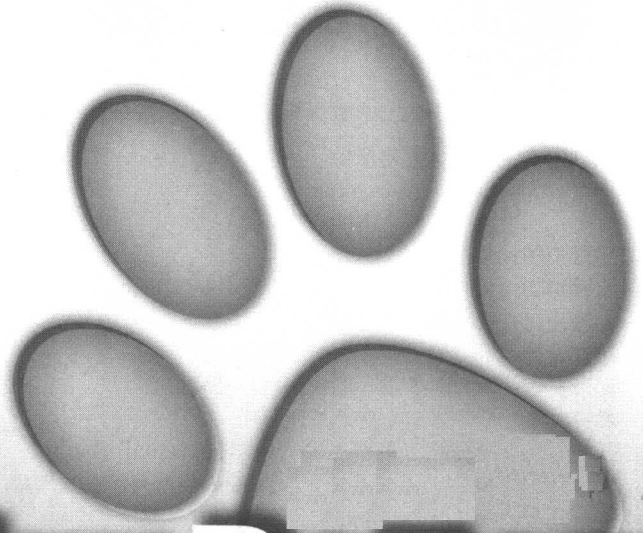
清华大学出版社

深入浅出

设计模式

(C#/Java 版)

莫勇腾 编著



Design Pattern

清华大学出版社

内 容 简 介

本书是一本通俗易懂的设计模式入门指导图书。

作者用 C#和 Java 两种语言，借助现实生活范例和图片演示，全面阐释 GRASP 及 GoF 23 种设计模式的概念及其编程应用，帮助你领悟设计模式的思想及精华，并将其融会贯通、灵活应用到自己的开发过程中。

全书用两章篇幅对设计模式和GRASP作了基本介绍，用三章的篇幅全面展开对23种设计模式的讲解：对于每一种模式，先给出定义，接着通过类比方式用一个现实世界中的例子说明模式的应用，然后分别以 C#和 Java 代码例述模式的架构实现。最后一章给出了两个设计模式综合案例，为读者实践设计模式提供了很好的学习环境。附录部分精心安排了自测题及答案，供读者练习并检验学习效果。

本书适合程序开发人员阅读，尤其适合作为大学计算机专业高年级学生和研究生的教学参考书。

图书在版编目（CIP）数据

深入浅出设计模式（C#/Java 版）/莫勇腾编著.－北京：
清华大学出版社，2006

ISBN 7-302-13564-9

Ⅰ. 深… Ⅱ. 莫… Ⅲ. 面向对象语言—程序设计
Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2006）第 088209 号

出 版 者：清华大学出版社　　**地　　址**：北京清华大学学研大厦
http://www.tup.com.cn　　**邮　　编**：100084
社 总 机：010-62770175　　**客户服务**：010-82896445
组稿编辑：夏非彼
文稿编辑：何　武
封面设计：林　陶　刘冉阳
版式设计：关　静
印 刷 者：北京科普瑞印刷有限责任公司
发 行 者：新华书店总店北京发行所
开　　本：异 16　　**印张**：22　　**字数**：480 千字
版　　次：2006 年 9 月第 1 版　　2006 年 9 月第 1 次印刷
书　　号：ISBN 7-302-13564-9/TP・8507
印　　数：0 001~4 000
定　　价：39.00 元

前　言

进入计算机时代以来，程序设计方法和编程过程不断发展演进，从结构化编程到模块化编程，然后发展到面向对象编程、组件化编程。现阶段，面向对象编程方法占据程序设计的主流，成为最常用且最重要的程序设计方法。

设计模式是资深程序员们总结出来的一种可重用的、针对面向对象软件设计的解决方案，所有结构良好的面向对象软件体系结构中都包含了许多设计模式。使用了模式的程序，其简洁和易于理解的程度远远超过了未使用模式的程序，所以设计模式日益成为编程人员追求的技术热点，越来越多的人愿意花大量的时间学习它。

本书不是一本单纯地笼统介绍设计模式的书，它从概念定义入手，以现实生活范例，结合大量代码示例介绍了面向对象设计的 GRASP（通用责任分配软件模式）原则及 GoF 的 23 种设计模式，教导读者如何将其灵活应用到自己的开发过程中。最后一章提供了两个设计模式综合应用案例，为读者进一步领悟设计模式的精妙并实践模式提供了很好的平台。在附录部分提供了一组自测习题及答案，以巩固和加深读者对模式的理解。

本着“学以致用”的编写原则，书中强调如何把模式运用得恰到好处，使读者从对设计模式的一无所知，到理解的混沌，转而进入一个明朗开阔的编程领域新天地。

本 书 特 点

（1）本书从基础开始引导读者进行正确的对象设计。很多人都学过 GoF 设计模式，但 GRASP 才是模式设计的基础。目前很多书都缺乏对 GRASP 基础原则的介绍，就直接描述 GoF 设计模式，使得读者很难理解和接受。

（2）本书让读者以最少的学习时间获得软件设计的最先进知识。由于设计模式比

较抽象，难以理解，故本书穿插大量图片，以生动现实的例子来说明模式，最大程度地降低了读者的学习曲线。正如阅读好的代码可以获取大量的编程技巧一样，阅读以好的模式编写的例子可以参悟大量的分析和设计技巧及经验。你会发觉，本书讲到的很多例子就是你以前遇到过的，将来很多时候，书中的例子还可以直接应用到你的项目中，因为它们很实用，很贴近实际需求。

（3）本书中所有设计模式的应用示例均以两种编程语言实现：Java 和 C#。读者在通过示例代码加深对模式的理解之余，还可以从这些例子中体会 Java 与 C#的区别，以及各自的长短之处，从而进一步掌握 C#和 Java 这两门主流编程语言。

目 标 读 者

本书面向那些有一定编程经验，但渴望在程序开发过程中不断提升编程质量的程序开发人员，对于即将毕业走入社会从事高质量高强度程序开发工作的计算机专业学生，本书能够带领你迅速参透设计模式的精妙，运用设计模式组织程序架构，进而写出高水平高质量的程序。阅读本书时，你掌握的编程语言可能不是 Java，也不是 C#，但模式是独立于开发语言的，理解了设计模式，掌握了设计模式的应用方法，你就能正确科学地组织分配程序中的各个类，建立架构良好的应用程序。所以，本书适合于任何编程语言的开发人员。

由于作者水平有限，书中难免有不足之处，欢迎各位读者不吝指正。读者可以发送邮件到 techmio@qq.com 与作者取得联系。

注意：本书所有示例的源代码，都可以在 http://www.khp.com.cn 网站的“下载服务”专区或本书子页面中找到，欢迎读者访问下载。

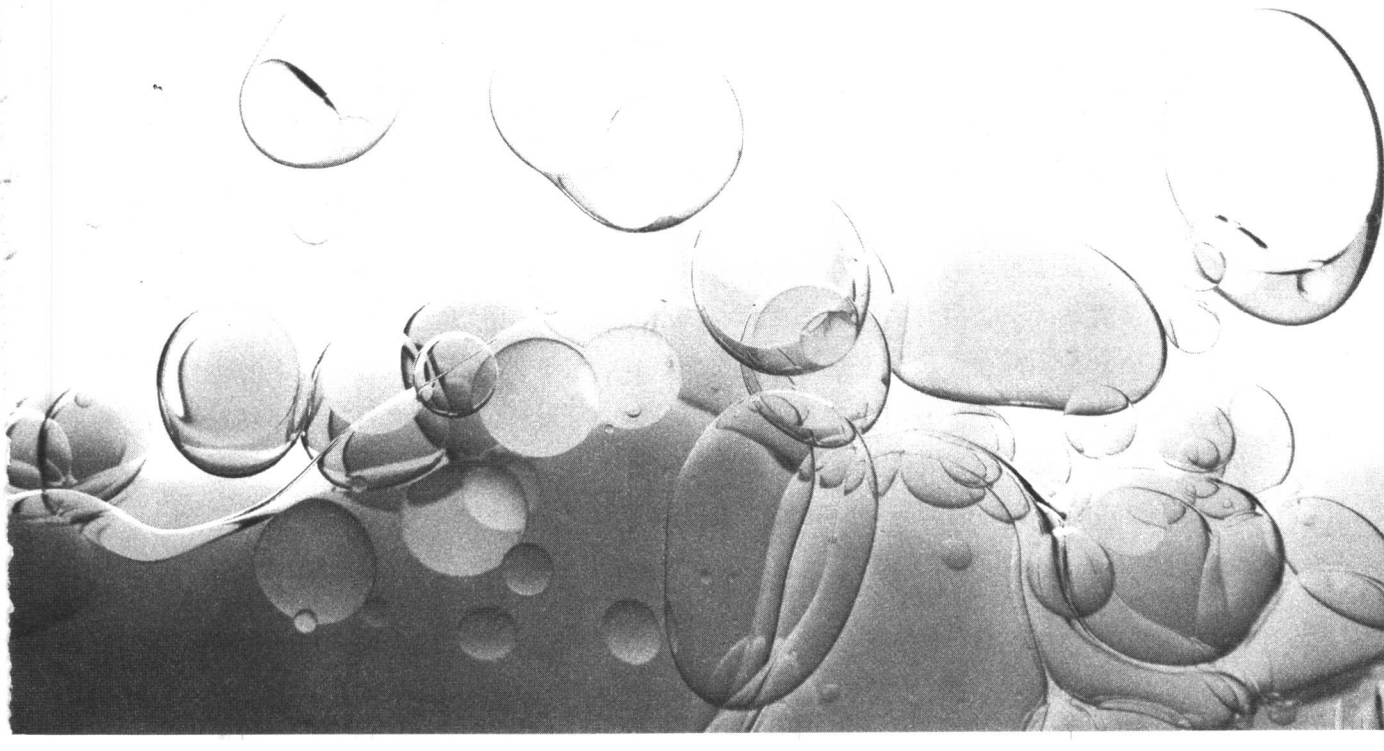
编　　者

2006 年 7 月 21 日于汕头大学

目　录

第 1 章

基 本 概 念



什么是设计模式

最近两年，模式已经成为软件项目团体里最热门的话题之一。模式的概念最早由建筑大师 Christopher Alexander 提出，他说："每个模式都描述了一个在我们的环境中不断出现的问题，然后描述了该问题的解决方案的核心。通过这种方式，你可以无数次地使用那些已有的解决方案，无需再重复相同的工作。"

模式的定义：**模式是一种问题的解决思路，它已经适用于一个实践环境，并且可以适用于其他环境。**

这个定义比较抽象，下面让我们用现实生活中的例子来说明什么是模式。比如，古人在遇到生存困难的时候，不断思考、实践、总结，最后得到了许多解决生活困难的方法。当后人遇到同样的问题时，也用同样的方法去解决，这些方法就可以称之为模式。

用牛耕田，织网捕鱼，打井取水……，所有这些都是前人创下的生活模式，我们大家自觉不自觉地把它们应用于生活的方方面面。那么程序设计呢？有没有一些前人的设计思路可供我们直接取用？

答案当然是肯定的，这就是设计模式！

设计模式通常是对于某一类软件设计问题的可重用的解决方案，将设计模式引入软件设计和开发过程，其目的就在于要充分利用已有的软件开发经验。优秀的软件设计师都非常清楚，不是所有的问题都需要从头开始解决，他们更愿意复用以前曾经使用过的解决方案。每当找到一个好的解决方案，他们会一遍又一遍地使用，熟练地使用这些已有的方案，是使他们成为专家的部分原因。设计模式的最终目标就是帮助人们利用成功软件设计师的集体经验，来设计出更加优秀的软件。

设计模式的种类很多，包括分布式编程模式、用户界面模式、数据模型模式三大类。目前流行的面向对象设计模式，仅 1995 年"gang of four"（四位作者：Erich Gamma，

Richard Helm，Ralph Johnson，John Vlissides）描述的就有二十多种，我们称之为 GoF 模式；与 GoF 模式相对应的另一种重要的设计模式是通用责任分配软件系列模式（GRASP，General Responsibility Assignment Software Patterns）。

GRASP 模式着重考虑设计类的原则及如何分配类的功能，而 GoF 模式则着重考虑设计的实现、类的交互及软件质量。可以说，GoF 模式就是符合 GRASP 模式要求的面向对象设计模式。

模式不是发明与创造，而是来自我们日常设计开发中的发现与总结，是一种实践的经验积累，它带我们走向成熟。

作为设计开发人员，我们也需要有发现模式的勇气。模式应该有以下特点：

（1）在特定的场景下有可重用性，对相同类型不同问题的环境，其解决方案都有效。

（2）可传授性，即问题出现的机会很多，解决问题的方案相同，人们相对可以接受。

（3）有表示模式的名称。

如果你发现的解决方案有以上特征，你也就发现了新的模式。

设计模式的作用

设计模式是软件开发的革命性成果，是很多软件设计人员成功设计经验的结晶，是复杂问题的简单解决方法。学习它可以让一个软件设计者提高设计能力。总地来说，设计模式主要有以下作用：

（1）重用设计，重用设计比重用代码更有意义，它会自动带来代码重用；

（2）为设计提供共同的词汇，每个模式名就是一个设计词汇，其概念使得程序员间的交流更加方便；

（3）在开发文档中采用模式词汇可以让其他人更容易理解你的想法，理解为什么你会这样做，你都做了些什么。编写开发文档也更加容易；

（4）应用设计模式可以让重构系统变得容易，可确保开发正确的代码，并降低在设计或实现中出现错误的可能性。还可以为重写其他应用程序提供很好的系统架构；

（5）正确使用设计模式，可以节省大量时间。

此外，设计模式与编程语言无关。无论采用什么高级编程语言，都可以应用设计模式。

GRASP 模式的分类

GRASP 通用责任分配软件模式没有大类，与其说它是面向对象的设计模式，不如说它是面向对象的设计要求。它细分有 9 种模式，模式与模式间不是独立的，存在相互平衡制约的关系，所以我们的设计只能努力满足 GRASP 的各种模式。本书的第 2 章会对 GRASP 的各种模式作重点介绍，这里先给出它的 9 种分类：

（1）Information Expert（信息专家）
（2）Creator（创造者）
（3）Low coupling（低耦合）
（4）High cohesion（高内聚）
（5）Controller（控制器）
（6）Polymorphism（多态）
（7）Pure Fabrication（纯虚构）
（8）Indirection（间接）
（9）Protected Variations（受保护变化）

GoF 设计模式的分类

GoF 模式在粒度和抽象层次上各不相同，分类很明显。这些分类有助于读者更快