

感受设计演变过程中所蕴含的**大智慧**。

体会**乐与怒**的程序人生中值得回味的一幕幕。

大话

设计模式

菜鸟



设计模式的趣味解读，面向对象的深入剖析。
在诙谐与温馨中做一次面向对象编程思维的体操。

程杰 著

清华大学出版社



原创经典

TP311.5/224

2007

大话设计模式

程 杰 著

清华大学出版社
北 京

内 容 简 介

本书通篇都是以情景对话的形式,用多个小故事或编程示例来组织讲解 GoF(设计模式的经典名著——Design Patterns: Elements of Reusable Object-Oriented Software, 中译本名为《设计模式——可复用面向对象软件的基础》的四位作者 Erich Gamma、Richard Helm、Ralph Johnson, 以及 John Vlissides, 这四人常被称为 Gang of Four, 即四人组, 简称 GoF) 总结的 23 个设计模式。本书共分为 29 章。其中, 第 1、3、4、5 章着重讲解了面向对象的意义、好处以及几个重要的设计原则; 第 2 章, 以及第 6 到第 28 章详细讲解了 23 个设计模式; 第 29 章是对设计模式的全面总结。附录部分是通过一个例子的演变为初学者介绍了面向对象的基本概念。本书的特色是通过小菜与大鸟的趣味问答, 在讲解程序的不断重构和演变过程中, 把设计模式的学习门槛降低, 让初学者可以更加容易地理解——为什么这样设计才是好的? 是怎样想到这样设计的? 以达到不但授之以“鱼”, 还授之以“渔”的目的。引导读者体会设计演变过程中蕴藏的大智慧。

本书适合编程初学者或希望在面向对象编程上有所提高的开发人员阅读。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13501256678 13801310933

图书在版编目 (CIP) 数据

大话设计模式 /程杰著. —北京: 清华大学出版社, 2007.12

ISBN 978-7-302-16206-3

I. 大… II. 程… III. 软件设计 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2007) 第 152713 号

责任编辑: 陈 冰

责任校对: 张 剑

责任印制: 王秀菊

出版发行: 清华大学出版社 地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编: 100084

c-service@tup.tsinghua.edu.cn

社 总 机: 010-62770175 邮购热线: 010-62786544

投稿咨询: 010-62772015 客户服务: 010-62776969

印 刷 者: 清华大学印刷厂

装 订 者: 三河市金元印装有限公司

经 销: 全国新华书店

开 本: 203×260 印 张: 24.75 字 数: 683 千字

版 次: 2007 年 12 月第 1 版 印 次: 2008 年 3 月第 2 次印刷

印 数: 5001~8000

定 价: 45.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 026653-01

本书是一本程序集？NO！

本书是一本故事集？NO！

本书是一本通过故事讲述程序如何设计的方法集。了解优秀软件设计的演变过程比学习优秀设计本身更有价值，因为设计的演变过程中蕴藏着大智慧。

感受设计演变过程中所蕴含的大智慧，
体会乐与怒的程序人生中值得回味的一幕幕。

*设计模式的趣味解读，面向对象的深入剖析。
在诙谐与温馨中做一次面向对象编程思维的体操。*

序

这本书最初起源于作者程杰在其博客中所写的连载文章——《小菜编程成长记》。随着文章的一篇篇发布，这些文章新颖的表现形式和独特的风格受到了众多读者的关注和喜爱，很多人在博客中留下了评语。有些虽然只有短短的一句话，但也可以看出是对作者由衷的感谢。作为本书的策划编辑，最初我也是在博客园中浏览博文时阅读到这些文章的，我的直觉和网友们热情洋溢的评语告诉我，这些文章有作为一部书出版的价值，于是我就联系了程杰。几个月后，我拿到了这部书的初稿。

初审后，我发现书稿中存在问题。比如，当时书稿中还没有对 UML 类图进行讲解的内容，这会导致初学者学习后面的内容时感到理解困难，于是我请作者在第 1 章中增加了 UML 类图这一节，这是简洁却精彩的一节；另外，当时作者为了便于表达某些举例的含义，有相当数量的代码都是用中文编写的，虽然中文代码看似易懂，但却会令绝大多数早已熟悉了英文代码的程序员们感到困惑和难以阅读，所以我请作者把代码改回为程序员们所熟悉的英文代码，并同时添加了更详细的中文注释。经过几番认真和辛苦的修改与调整，现在，这本书在你的手中了。

对于这本书，我想说的是，其中的很多篇章非常的精彩，会令你禁不住叫好，但也有一些篇章会显得有些拖沓，或者是有些牵强，然而，随着你读过那些精彩的段落，读过那些不那么精彩的段落，最终，你会读到书的最后一页（很多书不能使你做到这一点），当你读完全书时，你会发现，你的心情很愉快，很平静，即使是那些当时看起来不那么精彩的段落，现在也都成为了这温馨故事的一部分。你会记得书中那个好学、天真、而又执著的小菜，也会记得那个善于启发，经验老道的大鸟。

下面这些是来自作者博客的网友评论，看完这些热情洋溢的评论，就和作者一起，进入设计模式的大话境界吧。

本书策划编辑 陈冰

2007 年 10 月 18 日

网友评论

daigua: 看到这篇精彩的成长记，我连饭都不想吃了，什么事都不想做，就想把它看完。写得太好了！是啊，现在很多教材都太枯燥了，不好理解。其实书的意义就在于让人学到知识，而不在于用什么方式，为什么一定要那么教条呢，只要能让人比较容易地学到书里的知识就是一本好书。谢谢你啊，给了我很大的信心。我现在很有信心把编程进行到底，哈哈。

光头小松鼠: 绝对经典！一篇小故事，把程序的灵活性、可扩展性、可维护、可复用等说得怎一个妙字了得！

沉默天蝎: 感激，让我这个菜鸟顿悟。这样的写法太好了，如果老大你出书，我肯定购买！

碳碳: 这种学习的方式真的很神奇，尽管每个人都能想到，但不是每个人都能做到。或许可以把系列文章归档出书，说不定会收到追捧，呵呵。

Bryant: 真的是太棒了！我原来看过一些有关设计模式的书，都觉得太抽象，根本就不能理解，也不知道啥时候能用上。看过你写的这些文章，才知道应该怎样在实际中运用这些模式，而且文笔非常的幽默，享受！Thx ^^ 支持！有个建议，最好慢慢地把所有的设计模式都聊聊！

Bryant: 不错，楼主说的非常幽默，通俗，把我们一步一步带入面向对象的世界 thx ^^

Bryant: 太棒了，我正是这样初学设计模式的小菜，需要这样的文章，谢谢楼主！

菜鸟飞: 楼主，加油，支持你。在这里献上崇高的敬意，不管你有没有感受到我挚热的目光。请你相信，有这样一些人一直在默默地关注着你，期待着你。

wdx2008: 非常好!!! 幽默，搞笑，易懂，真神人也，鬼神不可测！支持楼主!!

空明流转: 呵呵，楼主说得蛮好。国外的文章好就好在有例子，“废话”多，所以比较好理解。至于行文风格嘛，这个倒是因人而异的。我个人就偏向于论文式的行文风格，逻辑严密，层层递进，阐述也很清晰。就有点像有序数组，二分法就能轻松查找到自己想要的东西，但国内的那种论文式的文章，呵呵，我看是卖弄的成分居多，实作的成分偏少，所以才那么难读的吧。

Char: 现在的大学就缺少这种既通俗易懂，又有内容的东西。

Apple: 不错，学习了。希望博主能再接再厉多写点，看了很多书都没有看你的文章明白得快。

SnowDoggie: 呵呵，挺好的。其实要想找个绝对没有漏洞的例子是很辛苦的，关键在于文章本身能说明问题，能体现作者的意图就足够了。昨天和朋友一起爬山的时候还讨论了你的文章风格，其实最有用的还是你这种寓教于乐，步步深入的风格，阳春白雪的经典虽然是经典，大众却不见得喜欢。

Jerry: 不错的文章，简单明了，又不乏趣味，好的文章就得顶下。

izhizhe2000: 很好，整个系列写完之后可以出书了，保证受大学生的广泛欢迎！

mekong: 很是欣赏这样幽默风趣又不失睿智深刻的文字。

Wuyisky: 呵呵，楼主不仅程序写得好，而且还有文学天赋。佩服！

Jack: 真正的高手是用最生动的语言，最简单的例子，这才是真正的“深入浅出”。赞!!! 老兄，加油，继续哟。

BoyLee: 人才，爱死你了。做了一年外包，没技术含量。正打算从头学习这些东西，这样的方式我最喜欢了。

Leoxu: 很不错，对正在找工作的我有很大的帮助。以后会多来光顾。

Ame: 写得承上启下，始终有一主干线贯穿，作者的文字功底很强啊！

Artech: 我很喜欢你的写作风格！以一种调侃的方式讲明一个深奥的问题。我一直在尝试如何以一种让每个人都懂得的语言来向大家分享我所理解的.NET。你给了我一个启发。

8: 醍醐灌顶！感谢，领悟了不少东西!!!

Yufengly: 真是太容易理解了，而且看后印象深刻，继续努力！期待下文……支持作者！

Sopper: 支持，例子举得很形象，写得很棒，以后会常来关注。

d: 会技术的高人有很多，但能把技术讲得如此通俗易懂的高人并不多，你是一个，谢谢~~~

white.wu: 非常喜欢您这种授人以“渔”的文章。

Answer: 强啊，本菜鸟受益很大，谢谢。

Hanlei: 强，很受益啊，感谢楼主，写出这么好的文章来。

金色海洋 (jyk): 继续呀，我们期待中……，写得很好，一看就懂。

DSharp: 看博客园这么久了，终于看到一篇有中国特色的好文。

本书出版后的读者评论

当当网的本书读者评论（当当网是只有买了书的读者才能发表评论的）

zenmelpengyou: 这本书是我从识字以来看过的最好的一本书了，诙谐幽默而又能学到知识~~~~~

aivdesign: 正在看。真是受益匪浅，作者不但用很通俗易懂的方式介绍了设计模式，同时还渗透了求职、面试、办公室文化等很多生活场景，让设计模式的概念融入其中，真是想忘也不容易忘啊，呵呵！

wxhwzz: 书很不错~诙谐幽默！应该是一种很好的创新，绝对是设计模式入门级的好书，赞一个！当当发货的速度也很快，两天就收到了！很感动~~

hwj_no1: 确实是本好书，在入门阶段和晋级阶段，都可以看看。并且能看到不同的启发，确实适合我这样的初级选手，语言和风格都很清新，在初级的不同阶段阅读，有横看成岭侧成峰的感觉。

asky99: 适合设计模式入门的初学者研读，技术要点融汇在诙谐幽默的小说当中，是我读过的最轻松的技术类书籍，支持作者。博客园的朋友。

蓝天 DotNet: 刚看了几天，感觉很不错的一本书。不象其它设计模式书籍，都是纯理论的，刚开始看就叫人难以捉摸。

Sofut: 这个是我在当当买过的最好的一本书！

xpwang_leo: 此书乃不可多得之佳作，通俗易懂，读完很容易吸收，其实以前也读过很多设计模式的书籍，发现再重来读这本书，有种醍醐灌顶的感觉，以前生涩难懂的地方，现在豁然开朗，如果想学设计模式，读它吧。

niyo: 不错，浅显易懂，看小说一样。

bellxie: 以前每次看模式的书，总是看到一半就昏昏欲睡，这本书一口气全部看完了。内容深入浅出，通俗易懂。总算看懂了几个模式。不错的好书。

苗苗老师: 看惯了 Wrox 的书，这次换换思维方式。看中国人写的书就是容易接受，毕竟是用母语思维然后用母语写的，不经过翻译。看书的过程不像看外国人写的书，总是走走停停，需要回味一下。看这本书是行云流水，一气呵成。

siso: 单单看后面的基础知识，就比许多 C#教程还好。

adamzhang: 这本书我每天花一个半小时左右来读，差不多两三周就看完了。非常通俗易读，而且讲得很好，能联系现实生活，很有趣味性。

卓越亚马逊的本书读者评论

ceduce193: 由于以前就一直关注作者的《小菜编程成长记》，发现 china-pub 上有，就在那儿买了。刚从 china-pub 拿到书，大致翻了一下，感觉很不错。正如作者在前言所讲的，“精彩的代码是如何想出来的，要比看到精彩的代码更加令人期待。”每一个设计模式都是从小故事入手，然后一步一步在小菜犯错大鸟启发中引出设计模式。书中有一段讲解这样写，“为了回归的大局，增加一种制度又何尝不可，一

个国家，两种制度，这在政治上，是伟大的发明。在软件设计模式中，这种不能修改，但可以扩展的思想也是最重要的一种设计原则，它就是开放-封闭原则”，这个例子举得太精彩了。反而觉得作者公开的两章不能完全代表书的全貌，强烈推荐购买。

stoat: 此书不错，比晦涩的教材好理解多了。通过示例代码的不断完善引出了各个设计模式的特征和使用方法，还有风趣的人物对话。我买的时候 38 块多，才过几天就降了 5 快!! 郁闷啊! 卓越的物价变得太快了吧!

互动出版网的本书读者评论

hermitbab: 我觉得不错，至少原先看的那些设计模式都很枯燥，看这个书就好像看小说一样，看好了记忆也深刻。

tiananz2006: 在书店看了两章，整书的思路就象 Head First 的思路，当然不如 Head First 所举的范例那么有震撼力，但仍然是本入门的好书! 购买此书的理由主要是: 1· Head First 在网上都有下载了，可以对比着读; 2· 所谈范例更针对于学生学习编程; 3· 价格便宜，支持国内原创。

songlonglong: 感觉很不错，四人团的书太晦涩，如果和这本书结合看，效果很好。

第二书店的本书读者评论

c25894670: 我对于设计模式只是一知半解，一直感觉 GoF 大师的作品以自己的水平实在无法理解。这本书实在是非常适合我这种不喜欢自己动脑子理解大师语言的人，用通俗的例子让我理解设计模式的精髓，我感觉爽得要死……推荐想入门设计模式的人看这本书，入门的不二法门……

lizhizhe2000: 在博客上看过作者的一些连载，写得很风趣，很适合初学者看看，对那些工作了两三年的人也有实际价值。

124.114.129.*: 支持这样的书籍，大道至简至易。

125.35.4.*: 刚收到书，看了几页，感觉这本书应该是程序员学习的第一本书，也许在讲设计模式的书中这不是最好的。但肯定是能让你看了不困能从头到尾坚持看完的，而且每一种设计模式的用法只用一句话就简明概括出来了，比起以前买的看了二、三十页就头脑发胀混混欲睡的牛人的著作，这本书真的可以称得上是《设计模式——菜鸟天书》，强烈推荐。到家才 37.2，便宜透了。

58.246.92.*: 这本书完全是作者的工作经验总结，很值得看。

biser: 入门的好书啊，不光菜鸟，老鸟也是很值得一读的，经典得很啊。

CSDN 读书频道的本书读者评论

218.106.251.*: “编程是门技术，更是一门艺术。”写得真好，看完了第一章，非常喜欢作者这种步步深入，循循善诱的写作方式。设计模式的书籍我也看过不少，不过以这么通俗的方式来讲解，特别是还有故事情节，幽默对白，的确给人耳目一新之感。让人忘记了是在学习设计模式，仿佛是在读小说一样，而读完之后，作者要讲的问题也就自然明白了。我猜测，这本书一定会引起不同凡响的轰动。

feng545218: 这是我的专业老师强烈向我们推荐的一本适合我们的书，看后真的很好。既轻松幽默，又能记忆深刻，还形象化的理解了问题。

218.82.87.*: 作者太 TM 有才了, 怎么想到的, 把 23 个模式变成 23 个 MM, 哈, 恶搞了超级女生、潜规则、黄金甲, 有意思。不过我最喜欢的还是第一章活字印刷那一段, 哈, “喝酒唱歌, 人生真爽。”面向对象经作者这么一说, 的确通俗无比, 比那种只会教条式的讲解面向对象的要强太多了。

abware: 强啊, 作者对 OO 和设计模式的理解有一定深度, 不然不会写得这么深入浅出。

dangnilaoqu: 不错呀。谁说理科就不能出文学家呀?

218.82.87.*: 刚刚看完, 正想发表一些感想, 发现楼下的朋友说风格抄袭 Head First, 我不赞同这个观点。比如里面写到了三国曹操吟诗, 写到了近期国内的 IT 或娱乐事件。Head First 是好书, 这一本没看完我还不知道是不是好书。但显然两本书风格是不一样的, 我认为这是一本非常中国化的设计模式的技术书, 如果国人都都不支持国人作品, 特别是已经不错的作品, 那国人的技术又如何可以振兴?

Maciloveyou: 写的很有特点, 风格很像《水煮三国》, 挺有意思的。

JunsGo: 如果高校的教材也这么有趣那该多好啊! ☺

58.246.92.*: 用很简单的生活道理来告诉你深奥的编程思路。

作者博客的本书读者评论

肖雪平: 昨晚有幸读到了你的《戏说面向对象编程》(也许就是上面提到的小菜编程成长记), 非常欣赏。说实在话, 一个人只是单纯的会写点文字, 发发自己的牢骚; 或者单纯的懂技术, 开发能力强, 我个人认为, 算不得什么。难得的是, 两者兼备。而你, 做到了。能让外行一眼就爱上面向对象, 爱上设计模式, 并且一读就明白。何其难啊! 这正是我一直努力的目标。因为我的职业就是软件开发老师。读过的晦涩、艰深、却百无一用的所谓教授专家所写的书, 不敢说过千, 但至少过百吧。看看你的, 禁不住由衷的感慨: 深入浅出的真谛, 莫过于此! 您的尝试非常有意义, 希望您再接再厉, 写出更多更好的文章, 为我们国家的软件行业培养更多更实用的人才。

James Liang: 很有意思啊, 之前就觉得你对这方面的东西有很深的研究和思维, 现在居然还赋予激情幽默的故事, 将它们说的很轻松和诙谐, 不错的。所有的模式是为业务而服务的。期待这本书有更好的表现。嘿嘿……

横刀天笑: 恭喜啊! 很赞同你的有些观点: 成为诗人后可能不需要刻意地按照某种模式去创作, 但成为诗人前他们一定是认真地研究过成百上千的唐诗宋词、古今名句。我一直认为学习设计模式也要走从量变到质变这条路, 不要把设计模式当圣经。

mekong: 衷心恭喜您! 真是应了是金子总会发光的! 期待着! 您能这样写书真是我等之福! 您教的工厂模式是我最先学用的模式, 逐渐领略了其优势而开始关注设计模式, 所以陆续浏览过 N 本设计模式的书, 总因晦涩难懂坚持不了 50 页、80 页就弃之而沿用了笨拙但顺手的方法。现在终于看到由您写的这本包含这多设计模式的书, 期待早早拜读……

newrain: 这一系列文章真的不错, 很不错的文章, 非常喜欢这样闲侃式的把思想传授给我们。

肖卓耘: 刚去 dangdang 了……说没货了……等一有货就买……(买回去当小说看) 说实话, 写得太好了……能和《明朝那些事》相媲美……

小侯: 很少有人能把设计模式介绍得清楚易懂, 你做到了, 而且很好, 很试合我们中国人的口味, 呵呵, 可能说得有点大, 不过很适合我, 请继续努力, 我会经常来光顾的。

棠棠 dotNet: 绝对经典! 看到博客园有这样的文章, 真是爽。长期关注楼主的文章。

scotoma: 博主的文采和文章我一直都很喜欢, 看到末尾博主所提到的每天的坚持, 我会努力做到的, 您的书我已经买了, 像博主所说的, 大学刚毕业很难找到工作, 我感觉主要的原因是每个人的四年

中的每一天的 24 小时是怎么花的，只要你努力的用好这些时间，我想四年会学到很多，还怕找不到工作吗？其实如果每个人都能够利用好时间，什么事情都可以做得相对完美的。期待博主的好文章。

ily: 楼主写的东西很好，可能对有些大虾来说是小儿科，但我相信对大多数开发人员来说不仅是一种互相学习，也是一种如何学会更高效更合理开发的思路，继续支持你！

tom385: 真的不错，一口气看完了，相当通俗易懂，学到很多东西！谢谢你，期待更多好文章！

Ruby113028: 其实像这类的书还是很有必要的。很多技术方面的书都当成字典一样的查询，由于实在是太枯燥了，往往读了没多少时间就放在一旁休息了。一直在寻找有没有作者能把技术书写活了，不要像帮助文档那样的枯燥。唉，如果都像帮助文档那样，还不如看帮助呢。希望能多几本大话设计模式这样的书。没有必要一定要划归到技术类，在休闲的时候看看其实也蛮有趣的，还能在休闲的时候学到点东西。

张卫林: 前段时间专程去书店买了，现在看了几章，非常不错。一本值得从头看到尾的书，也是一本能让人从头看到尾的书。

yarco: 另外，再次感谢作者，实际上我能很明确的看懂 UML 图例的这个聚合现象，完全是因为您在本书里放的大雁和雁群的关系，我几乎在一霎那间就懂了 UML:))

本书策划编辑对本书的评论

在出版这本书前，
我就知道这是一本好书，
我想象过它畅销的情景，
这，
是一回事。

在出版这本书后，
我真正看到这本书得到了读者的一致好评，
成为了一本畅销书，
这，
则完全是另外一回事。

给出几个在我写下上面这段文字时的数字吧：

当当网畅销排行榜第 4 名

卓越亚马逊畅销榜第 5 名

第二书店畅销榜第 1 名

CSDN 2007 年度十大精品图书第 1 名

我是本书的策划编辑陈冰，给读者带去好书是我的责任。如果你有任何意见或建议想要告诉我，可以直接到我的博客 (<http://w3cbook.blog.sohu.com/>) 上留言。

今天是 2008 年 1 月 31 日，还有几天就要过大年了。在这里我代表我自己、本书作者程杰和清华大学出版社全体同仁给读者们拜年，希望每位读者的每一天都有过大年前快乐的心情。

前 言

- 本书是一本程序集？NO。
- 本书是一本故事集？NO。
- 本书是一本通过故事讲述程序如何设计的方法集。
- 本书是给连 Hello World 都没写过的非程序员看的书吗？NO。
- 本书是给玩过穿孔纸带（0/1）、写过汇编、BASIC、C、C++、Delphi、Java、C#等语言，开发过覆盖全球、使用人数过亿、数百万行代码等大型系统的骨灰级程序员看的书吗？NO。
- 本书希望能给渴望了解 OO 世界的初学者、困惑于僵硬、脆弱、无法复用的代码编程体验者、一直打着 OO 编程的旗号，做着过程式开发的基于对象的编程实践者一些好的建议和提示。

本书起因

写本书源于我一次做培训的经历，学生大多是计算机专业的学生或有过一定经验的在职开发者。他们都知道类、方法、构造方法、甚至抽象类、接口等概念，并用 Visual Studio 写过不少的 Windows 或 Web 程序，可是当我提问为什么要用面向对象，它的好处在哪里时，却没有人能完整地讲得出来，多数人的反应是，概念知道的，就是表达不清楚。

针对于此，我就举了中国古代的四大发明中活字印刷的例子（见第 1 章），通过一个虚构的三国曹操做诗的情景，把面向对象的几大好处讲解了一下，学生普遍都感觉通俗易懂，觉得这样的教学比直接告诉面向对象有什么好处要更加容易理解和记忆。

这就使得我不断地思考这样一个问题，学一门技术是否需要趣味性、通俗性的引导。

我在思考中发现，看小说时，一般情况下我都可以完整地读完它，而阅读技术方面的图书，却很少有真正的每章每页的仔细阅读。尽管这两者是有很大区别，技术书中可能有不少知识是已经学会或暂时用不上的内容，但也不得不承认，小说之所以可以坚持读完是因为对它感兴趣，作者的文字吸引你。而有些技术书的枯燥乏味使得阅读产生了困难，通常读个前几章就留待以后再说了。

技术课的教学同样如此，除非学生是抱着极大的学习动机来参与其中，否则照本宣科的教学、枯燥乏味的讲解，学生一定会被庞杂的概念和复杂的逻辑搅晕了头脑，致使效果大打折扣。也正因为此，往往造成部分学生，学了四年的计算机编程，却可能连面向对象有什么好处都还说不清。

为什么不可以让技术书带点趣味性呢，哪怕这些趣味性与所讲的技术并不十分贴切，只要不是影响技术核心的本质，不产生重大的错误，让读者能轻松阅读它，并且有了一定的了解和感悟，这要比一本书写得高深无比，却被长期束之高阁要好得多。

也正是这个原因，本人开始了关于设计模式的趣味性写作的尝试。

本书读者

显然本书不是给无任何编程经验的人看的，对于想入这一行的朋友来说，找一门编程语言，从头开

始或许才是正道。而本书也不太适合有了多年面向对象开发经验，对常用的设计模式了如指掌的人看的。毕竟这里更多的是一些基础性的东西。

我时常拿程序员的成长与足球运动员的成长做对比。

GoF 的《设计模式》好比是世界顶级足球射门集锦，《重构》、《敏捷软件开发》、《设计模式解析》好比是一场场最精彩的足球比赛。我为之疯狂，为之着迷。可是我并不只是想做一个球迷（软件使用者），而是更希望自己能成为一个足球运动员（软件设计编程者），能够亲自上场比赛，并且最终能成为球星（软件架构师）。我仔细地阅读这些被誉为经典的著作，认真地实践其中代码，但是我总是半途而废、坚持不下去，我痛恨自己意志力的薄弱、憎恶自己无端地放弃，难道我真的就是那么的笨？

痛定思痛，反思悔过。我终于发现，贝利、马拉多纳不管老、胖是用来敬仰的，贝克汉姆、罗纳尔迪尼奥不管美、丑是用来欣赏的，但他们的球技……嗨，客气地说，是不容易学会的，客观地说，是不可能学得会的。为什么会这样？原来，我学习中缺了一个很重要的环节，我们在看到了精彩的球赛，欣赏球星高超球技的同时，却忽略了球星的成长过程。他们尽管有一定天分，但却也是从最底层通过努力一点一点慢慢显露出来的，我们需要的不仅仅是世界杯上的那定乾坤的一脚，更需要这一脚之前是如何练出那种神奇的方法，对于程序员来讲，精彩的代码是如何想出来的，要比看到精彩的代码更加令人期待。

本书显然不是培养足球明星（软件架构师）的俱乐部，而是训练足球基本功的学校，培训的是初学足球的小球员（面向对象的程序员），本书希望的是读者阅读后可以打好面向对象的基础，从而更加容易并深入的去理解和感受 GoF 的《设计模式》以及其他大师作品的魅力。

本书定位

本书是在学习众多大师智慧结晶的图书作品、分享了网上多位朋友的实践经验的基础上，加之自己的编程感受写出来的。正如牛顿有句名言：“如果说我比别人看得更远些，那是因为我站在了巨人的肩上。”但显然，本书并没有创造或发现什么模式，因此谈不上站在巨人肩膀上看得更远。所以作者更希望本书能成为一些准备攀登面向对象编程高峰的朋友的登山引路人、提携者，在您登山途中迷路时给予指引，在您峭壁攀岩摔跤时给予保护。

本书特色

本书有两个特色，第一特色是重视过程。看了太多的计算机编程类的图书，大多数书籍都是集中在讲授优秀的解决方案或者一个完美的程序样例，但对这些解决方案和程序的演变过程却重视不够，好书之所以好，就是因为作者可以站在学习者的角度去讲解问题所在，让学习门槛降低。《重构与模式》中有一句经典之语：“如果想成为一名更优秀的软件设计师，了解优秀软件设计的演变过程比学习优秀设计本身更有价值，因为设计的演变过程中蕴藏着大智慧。”本人就希望能通过小菜与大鸟的对话，在不断地提问与回答过程中，在程序的不断重构演变中，把设计模式的学习门槛降低，让初学者可以更加容易地理解，为什么这样设计才是好，是如何想到这样设计的。

本书的第二个特色就是贴近生活。尽管编程是严谨的，不容大话和戏说。但生活却是多姿多彩的，而设计模式也不是完全孤立于现实世界而凭空想出来的理论。事实上所有的模式都可以在生活中找到对应。因此，通过主人公小菜和大鸟的对话，将求职、面试、工作、交友、投资、兼职、办公室文化、生

活百味等等非常接近程序员生活原貌的场景写到了书中，用一个个小故事来引出模式，会让读者相对轻松地进入学习设计模式的状态。当然，此举的最大目的还是为了深入浅出，而非纯粹噱头。

本书内容

本书通篇都是以情景对话的形式，用一个又一个小故事或编程示例来组织的。共分为四个部分。第一部分是面向对象的意义和好处以及几个重要的设计原则，通过小菜面试的失败引出；第二部分是详细讲解 23 个设计模式；第三部分是对设计模式的总结，利用小菜梦到的超级模式大赛的场景，把所有的面向对象和模式概念都拟人化来趣味性的总结设计模式之间的异同和关键点。第四部分是附录，主要是针对面向对象不熟悉读者的一个补充，通过一个例子的演变介绍了类、封装、继承、多态、接口、事件等概念。

本书人物及背景

小菜：原名蔡遥，22 岁，上海人，上海某大学计算机专业大学四年级学生，成绩一般，考研刚结束，即将毕业，正求职找工作。

大鸟：原名李大辽，29 岁，小菜的表哥，云南昆明人，毕业后长期从事软件开发和管理工作，近期到上海发展，借住小菜家在宝山的空套房内。小菜以向大鸟学习为由，也从市区父母家搬到宝山与大鸟同住。

本书研读方法

本书建议按顺序阅读，如果您感觉由于面向对象知识的匮乏，例如对继承、多态、接口、抽象类的理解不足，造成阅读上的困难，不妨先阅读附录一的“培训实习生——面向对象基础”部分，然后再从第 1 章开始阅读。如果您已经对不少设计模式熟悉，也不妨挑选不熟悉的模式章节阅读。

本书源代码下载地址：

<http://www.cnblogs.com/Files/cj723/BigTalkDesignPattenSourceCode.rar>

也可到清华大学出版社网站（www.tup.com.cn）查找并下载本书源代码。

尽管本书中的代码都提供下载，但不经过读者的自己手动输入过程，其实阅读的效果是大打折扣的。强烈建议读者根据样例自己写程序，只有在运行出错，达不到预期效果时再查看本书提供的源程序，这样或许才是最好的学习方法。有问题可及时与我联系。我的电子邮箱是 chengjielong@163.com，博客是 <http://cj723.cnblogs.com/>。

本书中的很多精华都来自许多大师作品，建议读者通过笔记形式记录，这将有助于您的记忆和理解设计模式，增强最终的读书效果。

本书中出现的“[]”是表示句子摘自某书。例如，“策略模式（Strategy）：它定义了算法家族，分别封装起来，让它们之间可以互相替换，此模式让算法的变化不会影响到使用算法的客户[DP]。”其中“[DP]”表示此名摘自《设计模式：可复用面向对象软件的基础》，详细摘要说明请参看附录二。

本书中 29 章中的虚拟人物姓名都是软件编程中的专业术语，因此凡是专业术语被指向人物姓名的都用斜体字表示，以和实际术语区分。例如，“第一位是我们 OOTV 创始人，*面向对象先生*”，这里的斜体字 *面向对象* 指人名。

关于本书学习的疑问解答

➤ 看本书需要什么基础？

主要是 C# 或其他编程语言的基础知识，如变量、分支判断、循环、函数等编程基础，关于面向对象基础可参看本书的附录一。

➤ 设计模式是否有必要全部学一遍？

答案是，Yes！别被那些说什么设计模式大多用不上，根本不用全学的舆论所左右。尽管现在设计模式远远不止 23 种，对所有都有研究是不太容易的，但就像作者本人一样，在学习 GoF 总结的 23 个设计模式过程中，你会被那些编程大师们进行伟大的技术思想洗礼，不断增加自己对面向对象的深入理解，从而更好的把这种思想发扬光大。这就如同高中时学立体几何感觉没用，但当你装修好房子购买家俱时才知道，有空间感，懂得空间计算是如何的重要，你完全可能遇到买了一个大号的冰箱却放不进厨房，或买了开关门的衣橱（移门不占空间）却因床在旁边堵住了门而打不开的尴尬。

重要的不是你将来会不会用到这些模式，而是通过这些模式让你找到“封装变化”、“对象间松散耦合”、“针对接口编程”的感觉，从而设计出易维护、易扩展、易复用、灵活性好的程序。成为诗人后可能不需要刻意地按照某种模式去创作，但成为诗人前他们一定是认真地研究过成百上千的唐诗宋词、古今名句。

如果说，数学是思维的体操，那设计模式，就是面向对象编程思维的体操。

➤ 我学了设计模式后时常会过度设计，如何办？

作者建议，暂时现象，继续努力。

设计模式有四境界：

1. 没学前是一点不懂，根本想不到用设计模式，设计的代码很糟糕；
2. 学了几个模式后，很开心，于是到处想着要用自己学过的模式，于是时常造成误用模式而不自知；
3. 学完全部模式时，感觉诸多模式极其相似，无法分清模式之间的差异，有困惑，但深知误用之害，应用之时有所犹豫；
4. 灵活应用模式，甚至不应用具体的某种模式也能设计出非常优秀的代码，以达到无剑胜有剑的境界。

从作者本人的观点来说，不会用设计模式的人要远远超过过度使用设计模式的人，从这个角度讲，因为怕过度设计而不用设计模式显然是因噎废食。当你认识到自己有过度使用模式的时候，那就证明你已意识到问题的存在，只有通过不断的钻研和努力，你才能突破“不识庐山真面目，只缘身在此山中”的瓶颈，达到“会当凌绝顶，一览众山小”的境界。

编程语言的差异

本书讲的是面向对象设计模式，是用 .NET 中的 C# 语言编写，但本书并不是主要讲解 C# 语言或 .NET

框架的图书，因此本书同样适合 Java、VB.NET、C++等其他一些面向对象语言的读者阅读来学习设计模式。

就 Java 而言，主要差异来自 C#对于子类继承父类或实现接口用的都是“:”，而 Java 中两者是有区别的。

当 Cat 继承抽象类 Animal 时，Java 语法是

```
public class Cat extends Animal
```

当 Superman 实现接口 IFly 时，Java 语法是

```
public class Superman implements IFly
```

然后 Java 中所有的方法都是虚拟的，因此不用指定 new 或是 override 修饰符。还有一些其他差异，但基本都不影响本书的阅读。

对于 VB.NET 的程序员，如果阅读困难，不妨去网上查找关于转换 C#与 VB.Net 语言的工具，比如 <http://www.kamalpatel.net/ConvertCSharp2VB.aspx>，将下载本书的源代码转换后再进行阅读。

C++的程序员，可能在语言上会有些差异，不过本书应该不会因为语言造成对面向对象思想的误读。

不是一个人在战斗

首先要感谢我的妻子李秀芳对我写作本书期间的全力支持，没有她的理解和鼓励，就不可能有本书的出版。而我们的宝宝也将在 2008 年初出生，希望等宝宝懂事后能知道，在宝宝的母亲怀胎过程中，宝宝的父亲也在为书的诞生而努力。也希望本书成为赠送给他或者她的最好的礼物。

父母的养育才有作者本人的今天，本书的出版，寻根溯源，也是父母用心教育的结果。养育之恩，没齿难忘。

本书起源于本人在“博客园”网站的博客 <http://cj723.cnblogs.com/> 中的一个连载文章《小菜编程成长记》。没想到连载引起了不小的反应，网友们普遍认为本人的这种技术写作方式新颖、有趣、喜欢看。正是因为众多网友的支持，本人有了要把 GoF 的 23 种设计模式全部成文的冲动。非常感谢这些在博客回复中鼓励我的朋友。

这里需要特别提及洪立人先生，他是本人在写书期间共同为理想奋斗的战友，写作也得到了他的大力支持和帮助，我写作的不少妙句也来自我们俩共同合作的网站 <http://www.miaojunet.net>。在此对他表示衷心的感谢。

写作过程中，本人参考了许多国内外大师的设计模式的著作。尤其是《设计模式》（作者：简称 GoF 的 Erich Gamm, Richard Helm, Ralph Johnson, John Vlissides）、《设计模式解析》（作者：Alan Shalloway, James R. Trott）、《敏捷软件开发：原则、模式与实践》（作者：Robert C.Martin）、《重构——改善既有代码的设计》（作者：Martin Fowler）、《重构与模式》（作者：Joshua Kerievsky）、《Java 与模式》（作者：阎宏等等，没有他们的贡献，就没有本书的出版。也希望本书能成为更好阅读他们这些大师作品的前期读物。

写作过程中，本人还参考了 <http://www.dofactory.com/> 关于 23 个设计模式的讲解，并引用了他们的结构图和基本代码。在博客园中的许多朋友，比如张逸、吕震宇、李会军、idior、Allen Lee 的博文，MSDN SmartCast 中李建忠的讲座，CSDN 博客中的大卫、ai92 的博文，网站 J 道 www.jdon.com 的版主

banq 的文章都给本人的写作提供了非常大的指引和帮助，在此表示感谢。另外博客园的双鱼座先生还对本人的部分代码提出了整改意见，也表示衷心的感谢。详细参考资料与网站链接，见附录二。

事实上，由于本人长期有看书记读书笔记的习惯，所以书中引用笔记的内容，也极有可能是来自某本书或者某个朋友的博客、某个网站的文章。而本人已经无法一一说出其引用的地址，但这些作者的智慧同样对本书的写作带来了帮助，在此只能说声谢谢。

最后，对本书的责任编辑陈冰先生及清华大学出版社的相关工作人员，表示由衷的感谢。本书的出版离不开陈先生的指导和其他工作人员的辛勤工作。

程 杰

2007年7月