

<<软件设计精要与模式>>

图书基本信息

书名：<<软件设计精要与模式>>

13位ISBN编号：9787121104534

10位ISBN编号：7121104539

出版时间：2010-4

出版时间：电子工业出版社

作者：张逸

页数：450

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件设计精要与模式>>

前言

我希望告别冗长的前言，仅述说第2版的变更。

写作第2版的我，疯狂地吸收了诸多大师的设计思想，这一点可以从参考文献的前后差别看到端倪。这两年以来，我又参与了几个项目的设计与开发工作，所谓“实践出真知”，在佐证大师观点的同时，自己对设计的认识更进了一步。

或许，第2版不会比第1版优秀太多，但至少会减少诸多不足。

囿于版本，我无法做出新的突破。

我期待能创作一本全新的书，全面论述我对软件设计的认识。

现在的我，还不足以写出梦想中的软件设计之道。

言归正传。

整体而言，我对第1版的所有章节都进行了一定程度的修订。

或者更正了过去的错误，或者进一步完善了原有内容。

本书的内容仍然是散漫而自由的，然而形散而神不散，大体遵循了设计的基本原则。

在第1篇“设计之要”中，我新增了“对象法则”一章，言简意赅地介绍了面向对象思想的核心要素与设计原则。

这基于我的一贯理念，即设计模式的核心本质是面向对象设计思想的运用。

只有掌握了面向对象设计思想，才能真正体会设计模式的精髓，并将其运用在实际的项目开发过程中。

“对象法则”一章可以有机地与“封装变化”一章结合起来，再加上第23章“软件体系架构”的内容，基本上勾勒出软件设计的脉络，从面向对象思想到设计模式，再到软件体系架构。

在“封装变化”一章中，我不仅完善了项目实例，还增加了关于如何“解耦具体依赖”的几种技巧。对于软件设计而言，这是非常有益的指导。

我整个儿删去了第1版的第5章“设计，由你掌握”，并将其中的部分内容转移到“封装变化”一章中。

这使得第1篇的内容更为紧凑，虽然删去了讨论极限编程的相关内容，却可以使得我们能够更加关注于设计，而不是方法学。

第2篇“.NET框架与设计模式”增加了对.NET3.X的源代码分析。

我无法做到与时俱进，因为.NET4.0即将走进.NET开发人员的程序生活。

或许在本书出版之后的不久，还会有5.0, 6.0.....我只是希望我的书不要被时代抛弃得太远。

好在设计模式本身属于经典，而经典总是能够经得起时间考验的。

本书讲述经典，自然能讨得一定好处。

更新最明显的是迭代器模式在.NET中的实现。

C#2.0引入的yieldreturn以及.NET3.0引入的Lambda表达式都为迭代器模式在.NET中成为一种惯用法贡献了一份心力。

我对此的分析，可以在一定程度上帮助读者更好地理解迭代器模式。

在第2篇中，我新增了一章“.NET中的命令模式”，通过解析.NET3.0引入的WF

(WindowsWorkflowFoundation)，展现命令模式的非凡价值。

第2篇的内容虽然与.NET平台息息相关，但对于其他平台的开发人员而言，仍有可观之处。

我在撰写本书第2版时，同样参考了Java平台的设计理念，以及Ruby中的设计模式。

<<软件设计精要与模式>>

内容概要

《软件设计精要与模式（第2版）》关注的焦点是软件设计，涵盖了大部分与设计有关的基本要素，包括面向对象思想、设计模式、重构、测试驱动开发、极限编程以及软件体系架构。其中，尤以设计模式为主，深入探讨了软件设计过程中的原则与模式，并结合大量的实例与代码演示了如何合理运用设计模式，改善程序模块的可复用性、可扩展性，实现模块间的松散耦合。全书将软件设计理论与项目实践完美地结合起来，使其告别了纯理论研究的空泛，具有现实的指导意义。

《软件设计精要与模式（第2版）》适用于所有希望提高软件设计水平的程序员、软件工程师，同时，对于软件设计师与系统架构师也具有一定的参考价值。

<<软件设计精要与模式>>

作者简介

张逸，软件架构师，敏捷教练，微软最有价值专家，InfoQ.NET丰土区常务编辑，msup资深讲师。目前担任中软国际ETC（重庆）技术总监。

主要专注于软件架构、设计模式、领域驱动设计和敏捷开发，以及基于.NET平台的企业级系统应用开发和分布式开发。

曾先后为可口可乐、摩托罗拉、HP、京东方、尼桑、摩洛哥电信、盛大、第九城市、CA、携程等企业提供设计、开发、咨询与培训服务。

他的著译作包括《软件设计精要与模式》、《WCF服务编程》。

<<软件设计精要与模式>>

书籍目录

第1篇 设计之要第1章 设计之道1.1 计划的设计与演进的设计1.2 架构的标准1.3 过度设计，还是简单设计1.4 需要设计模式吗1.5 重构是必然的1.6 UML重要吗1.7 测试驱动开发第2章 对象法则2.1 面向对象思想2.2 面向对象核心要素2.2.1 封装2.2.2 继承2.2.3 多态2.3 面向对象设计原则2.3.1 单一职责原则2.3.2 开放封闭原则2.3.3 Liskov替换原则2.3.4 依赖倒置原则2.3.5 接口隔离原则第3章 封装变化3.1 设计模式之鸽的3.2 如何封装变化3.2.1 封装对象行为的变化3.2.2 封装对象创建的变化3.2.3 封装对象结构的变化3.3 解耦具体依赖3.3.1 配置文件与反射技术3.3.2 表驱动法3.3.3 惯例优于配置3.3.4 依赖注入第4章 体验重构4.1 一个阶乘容器4.2 代码的坏味道4.3 需求发生变化4.4 运用重构利器4.5 引入设计模式第5章 测试驱动开发5.1 发现价值与生产价值5.2 运用面向对象编程思想5.3 适时小结5.4 收获价值5.5 无上之力第2篇 NET框架与设计模式第6章 NET中的工厂方法模式6.1 “嫁祸江东”之计6.2 WebRequest对象的创建6.3 改造工厂方法模式6.4 实现良好的可扩展性6.5 惯例优于配置第7章 .NET中的合成模式7.1 字处理软件7.2 NET框架中的控件7.3 “复合方式”的合成模式第8章 .NET中的装饰器模式8.1 一步一步走向装饰器模式8.2 NET框架中对流的处理第9章 NET中的迭代器模式9.1 分辨对象的职责9.2 实现迭代器模式9.3 迭代器模式大观园9.4 迭代器模式在.NET框架中的实现9.5 通过yieldreturn实现迭代器9.6 C#3.0对迭代器模式的支持第10章 NET中的策略模式10.1 税务系统10.2 利用委托实现策略模式10.3 NET框架中的Provider对象第11章 NET中的命令模式11.1 考勤管理系统11.2 命令模式在WF中的运用11.3 合成模式与命令模式的结合第3篇 媒体播放器的设计之旅第12章 面向对象与接口分离12.1 拙劣的设计12.2 面向对象设计12.3 接口分离12.4 解除具体依赖第13章 接口行为的适配13.1 新的应用场景13.2 “狼”与“羊”的关系13.3 引入适配器模式13.3.1 类的适配器模式13.3.2 对象的适配器模式13.4 适配器模式比较第14章 接口行为的扩展14.1 设计走向死胡同14.2 考察适配器模式14.3 引入访问者模式14.4 非循环访问者模式14.5 合理运用访问者模式第15章 接口行为的装饰15.1 不断变化的需求15.2 考虑代理模式15.3 引入装饰器模式第4篇 设计模式应用实践第16章 抽象工厂模式应用16.1 报表组件16.2 糟糕的设计16.3 引入抽象工厂模式16.4 工厂对象的使用.....第17章 建造者模式应用第18章 命令模式应用第19章 职责链模式应用第20章 观察者模式应用第21章 代理模式应用第22章 设计模式综合应用第5篇 NET体系架构第23章 软件体系架构第24章 数据访问层第25章 消息处理第26章 ASP.NET缓存第27章 业务逻辑层第28章 变现层参考文献

<<软件设计精要与模式>>

章节摘录

插图：第1篇 设计之要第1章 设计之道1.4 需要设计模式吗如果仅考虑实现当前的功能需求，还需要设计模式吗？

坦白地说，我并不认为设计模式与过度设计有关。

过度设计的导火索是设计模式的滥用。

很多时候，合理地利用设计模式反而能使程序结构简单化，特别是，它能够让开发过程更简单。

Christopher Alexander在描述城市和建筑模式时说，“每一个模式描述了一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心。

这样，你就能一次又一次地使用该方案而不必做重复劳动。

”设计模式正是为解决软件开发领域中的类似问题，从而提出的解决方案，它作为面向对象编程经验的总结，是软件设计技巧中最重要的方法与原则。

放弃设计模式，意味着希望成为优秀设计师的你，在将历史的车轮向前倒滚了十年之后，有幸成为了和GOF同行的伟大先驱者。

你放弃了站在前人肩膀上眺望远方的机会，而选择自己艰难吃力地踮起脚尖，最后失去的却是更远处美丽的风景。

需要设计模式吗？

答案看来是不言而喻。

关键点是需要确定模式的应用是否过度？

世界上很多天才横溢的程序员，可以在一段代码中包含6种设计模式，也可以不利用模式就能把设计做得很好。

然而，软件设计的目标并不是炫耀茴香豆的“茴”字有几种写法，我们需要的是有效的设计。

学习设计模式，不是为了炫耀、吹嘘，不是为了故作艰深，而是为了改善我们的设计，它可以为某种功能实现提供参考模型、设计方法以及应用范例。

我们不需要奉GOF的设计模式为圭臬，盲目地膜拜它，合理地运用设计模式，才是明智的抉择。

<<软件设计精要与模式>>

媒体关注与评论

本书力求讲解浅显明白。

在技术探讨上，尽可能地深入透彻；在每一字的描述上，尽可能地简单易懂。

本书适用于所有希望提高软件设计水平的程序员、软件工程师，同时，对于软件设计师与系统架构师也具有一定的参考价值。

从实践分享的视角来看，书的内容编写地非常认真。

作者从自身工作的经历，分享了自己对软件设计的理解，并以设计原则这样的方式，来分享最宏观层面上的要点。

总结、思考的分量，可见一斑。

本书有些章节很有新意，注意到了利用自身实践过的设计模式，以真实示例的方式来介绍如何灵活使用各种设计模式。

此举对读者的实际工作，颇有帮助，愿为推荐。

——西门子中国中央研究学院，首席架构师

<<软件设计精要与模式>>

编辑推荐

《软件设计精要与模式(第2版)》：“给我一个支点，我就能撬起地球”，关键不在于力量有多大，而在于如何合理地利用力量。

软件设计同样如此，思想的确立，技巧的把握，将在很大程度上制约软件架构的合理性。

《软件设计精要与模式(第2版)》力图通过大量的项目实践，并与软件设计思想完美结合，充分展示软件设计之美，以及设计“力量”的巧妙运用。

<<软件设计精要与模式>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>