

# 《简单之美》

## 图书基本信息

书名：《简单之美》

13位ISBN编号：9787111301035

10位ISBN编号：711130103X

出版时间：2010年4月

出版社：机械工业出版社

作者：倪健

页数：292

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)

# 《简单之美》

## 前言

# 《简单之美》

## 内容概要

本书不是一本关于方法论的理论性书籍（尽管已经尝试在大量的思考上进行一些理论归纳），也不是一本关于具体技术的操作手册。本书为读者呈现的是作者在软件开发实践中的思考和体验，目的在于探究实践中的问题的根源，并给出思想上的解决之道。

本书以软件开发的基本顺序为主线，以简单、想象和文化的开发思想为线索，以为软件开发实践提供指导价值为目标。本书力图做到传递经过组织的、可操作的软件开发经验，使软件开发领域的新手在知识（软件开发思想）积累过程中少走弯路，也使软件开发领域的老手可以对照自己的经验把可能的困惑降到最低。有一些有趣例子为那些有价值的思考提供支持。

本书中，作者尝试对简单、想象和文化做一些形而上的思考。这些思考会比较抽象，但却是很多话题的基础。

# 《简单之美》

## 作者简介

倪健，资深软件工程师、项目经理和架构师，拥有12年IT行业从业经验。从事过电气设计与装配、单片机开发、软件系统设计与开发、项目管理、软件架构、软件培训以及性能调优相关的工作。精通汇编语言、VB、VC、Java和C#，自主设计过一些框架。12年的工作经验帮助他建立了一个完整的认识世界的思想体系，这个体系的核心内容就是简单、想象和文化。他擅长想象，善于归纳，内心坚持，追求有意义的生活。

倪健技术博客地址：<http://nijian.javaeye.com>

## 书籍目录

场景故事人物关系图本书赞誉前言为什么写本书本书面向的读者如何阅读本书第1章 无极生太极 1.1 创造的根源 1.2 本质的把握 1.3 简单的追求第2章 关于软件开发方法论的思考 2.1 方法论的实践场景 2.2 CMM的精髓 2.2.1 过程定义 2.2.2 成熟之路 2.3 敏捷软件开发的精髓 2.3.1 人与实践 2.3.2 海岸灯塔 2.4 最好的软件开发方法 2.4.1 中庸 2.4.2 聚焦 2.5 方法论的执行 2.5.1 关于执行 2.5.2 约束与习惯第3章 关于需求的思考 3.1 需求的实践场景 3.2 需求开发 3.2.1 准确表达 3.2.2 信息传递 3.3 需求管理 3.3.1 变化之道 3.3.2 应对之道第4章 关于软件架构设计的思考 4.1 架构设计的实践场景 4.2 软件架构的本质 4.3 软件架构的设计 4.3.1 业界技术成果 4.3.2 软件框架 4.3.3 隐喻的价值 4.3.4 架构模式 4.3.5 软件架构师的素质第5章 关于软件实现的思考 5.1 软件实现的实践场景 5.2 模型的设计 5.2.1 领域模型 5.2.2 计算模型 5.3 软件开发 5.3.1 写给新手 5.3.2 掌握编程 5.4 负责制度 5.5 软件决策 5.6 质量保证第6章 关于软件测试的思考 6.1 软件测试的实践场景 6.2 软件测试的本质 6.3 破坏的意义 6.3.1 软件测试的第一个原则：聚焦 6.3.2 软件测试的第二个原则：积累 6.3.3 软件测试的第三个原则：自动化 6.3.4 软件测试的第四个原则：目标一致 6.4 一个典型的破坏：性能测试 6.5 其他第7章 关于团队的思考 7.1 团队的实践场景 7.2 湘江战役 7.3 团队建设 7.3.1 思想准备 7.3.2 如何组建团队 7.3.3 现实与理想 7.4 团队管理 7.4.1 论人 7.4.2 稳定的价值 7.4.3 团队文化第8章 关于项目管理的思考 8.1 项目管理的实践场景 8.2 项目管理的本质 8.3 关于项目经理 8.4 计划的本质 8.5 风险的问题 8.6 管理的境界第9章 关于软件维护的思考 9.1 软件维护的实践场景 9.2 软件维护的本质 9.3 组件化开发 9.4 版本依赖 9.5 妥协的方案第10章 关于组织发展的思考 10.1 组织发展的实践场景 10.2 知识积累 10.3 行业成熟 10.4 知识资产

### 章节摘录

插图：而软件开发的本质是什么呢？软件开发是一项具有艺术性的科学工作。一方面，因为软件是逻辑学和数学的应用，所以软件开发是科学性的工作；另一方面，因为软件的实现有无限种结构形式，而且每一种结构形式都富含人类的创造和想象，所以软件开发是艺术性的工作。有些人还可能会从心理学和管理学角度来阐释软件开发的本质。我不赞同这个观点，但这不是重点。重点是什么呢？重点是我们需要形成一个自己的、系统而且完整的观念。这种观念，一定会与你所有的认知融合在一起。它来自经验和实践，反复验证之后会成为你思想的一部分。很显然，要排斥一种观念是容易的。可是，没有自己的思想体系，就更加容易陷入一种迷惘的状态。如果你尝试建立有体系的观念，你会用自己的观念解释一切，你也会不断地在失败中修正观念，那么你最终会在自己观念的带领下走向成功。举个例子，在现实世界中，苹果会从树上掉下来，你不会怀疑这一点，这就是你的世界观的一部分。可是，在微观世界里，你的观念会遭到重大打击。你需要调整世界观，用更复杂的方法来解释世界。你不断调整，最终得到真谛。这种有体系的观念，我称作思想体系。思想体系的建立是很有用的。我们不妨以学习为例。你知道吗？最有效的学习方法就是用自己的思想体系来接纳外部知识。这也可以说明一个问题，你看，在计算机书店，有很多大师级的作品，他们的知识向所有人开放，可是很多人从中得到的收获总是非常有限的，为什么？因为思想体系还没有建立，很多人还无法对书中的知识进行有意识地提炼和抽象，所以总是会有很多疑惑和不解。其实，只要经过足够长的时间，每一位软件开发人员都会不自觉地形成自己的软件开发思想，这和过日子是一样的。不过，有很大一部分软件开发人员在经历短暂的思想积累以后，却偏离了软件开发的方向。很多软件开发人员，热衷于转向另一个更加不成熟的领域——软件开发管理。

## 媒体关注与评论

正如本书所说，这是一本为那些在实际工作中陷入迷惘或存在困惑的软件开发、管理者和企业经营者准备的书。当你面对软件开发、管理和架构中的迷惘和失败时，你也许会在追溯问题根源的过程中用到本书中的知识。——Java开发者社区 在IT技术领域，中国缺少技术大师，但中国不缺少热爱技术、热爱思考的人，本书的作者便是其中的一个典型代表。他用自己的思考和经验带领读者朋友一起去探讨软件开发的本质——大道至简。如果你也是一位热爱技术和思考的人，本书决不容错过。——架构师社区 这是一本充满智慧的书，它让我们看到了一位在软件开发实践中善于观察、总结和思考的智者形象；这是一本富含哲理的书，它将软件开发的一切问题的本质都归于简单、想象和文化，耐人寻味；这是一本饱含经验的书，它系统地论述了作者在软件开发的整个过程中所总结的大量有价值的观点，引人深思；同时它也是一本充满趣味、非常耐读的书，书中的场景故事既让我们读起来心情舒畅，又能体会到其中蕴含的软件开发哲学。建议所有从事IT工作的朋友都收藏这本书，常读常新。——中文Java技术网 本站站在一个较高的角度对软件开发的各个环节和层面（需求分析、架构、实现、测试、维护、团队和项目管理等）都进行了详尽的阐述，尝试着去探讨软件开发的本质和方法论。作者将自己多年的从业经验融入本书中，希望使得无论是新手还是老手，都可以从本书中获得有价值的经验和技巧。——PHPChina 本书也许会让你产生共鸣，因为很多问题也是你在实际开发中曾经遇到并认真思考过的；本书也许会让你有些疑惑，因为有些观点只有当你的经验足够丰富时才会悉心领会；本书也许会让你拍案叫绝，因为有些问题你百思不得其解，而它却一语“道破”天机。强烈建议反复品读本书，它值得你拥有。——Ajax中国 我们始终相信，最简单的也就是最有效的，一个好的idea往往都能通过一些简单的技术实现，从最早的IM的成功，到SNS的火爆；从AJAX旧瓶装新酒，到jQuery改变JavaScript的书写习惯，莫不是如此。事实证明，简单才是王道！本书用简单的原则、富于想象的精神引领我们悟透软件开发的本质和奥义。——jQuery中文社区 大道至简，软件行业也是如此。书中一位饱含经验和善于总结的智者引领我们去思考与领悟软件开发的本质，发人深省。这是一本值得开发者反复阅读的书，历久弥新。——AIRRIA 艾睿网 品位本书仿佛是在聆听一位智者将他多年积累下来的宝贵经验娓娓道来，也仿佛是在欣赏一个精彩的外企白领故事。本书蕴含大量软件开发和管理的哲理，而且极具趣味性。——一起Ext 当你手持武器带领着你的兄弟们披荆斩棘、一次又一次穿越了障碍的时候，你们有没有回头远望并深思过？你们所做过的一切，所使用的一切，所创造的一切，都深思过么？来，拢个火堆，围着坐下来聊聊、想想。当太阳再次升起的时候，你们是不是还会保持原有的队形，是不是依旧穿着厚重的盔甲，肩扛战锤，像以前一样前进？这本书深入讨论了如何巧妙地“丢盔弃甲”，让你穿着休闲装一样能站在胜利之巅！——51CTO.COM读书频道 与它的名字一样，本书的内容也在简单、朴实中给读者一种至上的美感。本书为国内的开发者奉上了一顿丰盛的精神文化大餐，赶紧享受吧！——CSDN读书频道

## 《简单之美》

### 编辑推荐

《简单之美:软件开发实践者的思考》：这是一次软件开发者的心灵沟通之旅，10大社区一致鼎力推荐。作者在软件领域工作了12年，也思考了12年。从懵懂无知的开发新手。到成竹在胸的领域行家，其间的人生历练和思想结晶，都化成这本书中的文字，呈现在你的面前。你可以在这本书中看到自己过去或未来的影子，也可以听到自己欲言未吐的声音。还可以感觉到自己面对峰回路转的喜悦。《简单之美:软件开发实践者的思考》的内容涵盖软件开发的整个流程，从方法论到架构思想，从需求分析到软件维护，从个人世界到团队管理.....所有的内容都是从思想出发，以实践为证，直奔问题的根源，挖掘简单的本质。帮助读者体验软件开发中的简单之美是作者撰写这本书的价值追求。当你掩卷沉思的时候，请记得有一颗心灵在对你微笑。多年以来，不管是从事一线的软件开发工作，还是从事管理工作，作者一直在思考这样一个问题：业界有这么多知识财富，可是在实践中真正能够被吸收和应用的却很少，这些知识财富的价值是毋庸置疑的，软件开发人员的热情和渴求也是有目共睹的，可问题究竟出在哪里呢？作者最后得出的结论是：这个问题要归结于思想和文化。无论方法论也好，管理理论也好，都是技术层面的，它们来自于大师们的总结和提炼，本身是非常好的，但是它们有复杂的上下文，例如在敏捷开发中没有提到或者强调的——注重个体文化、专业化思想，以及多元化价值认同等，这些都属于思想和文化的范畴，它们是方法论和管理理论的运行环境。换句话说，如果割裂这些上下文，机械地运用那些技术层面的东西，效果就总是不好。所谓机械运用，就是追求形式化的执行，而没有站在以人为本的角度展开思考。以人为本的核心就是对思想和文化的关注。成熟的思想和文化，可以使软件开发中的每一个细节都变得浑然天成；相反，不注重思想和文化的建设，就只能依赖无法预料的外部约束，例如，沉迷于方法论的技术细节而无法自拔。事实上，在软件开发领域，没有解决问题的银弹，没有提升效率的短期计划，也没有获得成功的操作指南，一切都依赖于人。



## 精彩短评

- 1、制度方法流程固然重要，人是更重要的一环，相辅相成。战略对了，战术也要能执行到位。
- 2、就没见过几本国人写的、能看的技术书籍
- 3、风格稍微有点奇怪的一本书。包含了故事，大道理（并非贬义），意识形态，切身体验和一些一笔带过的思路技巧。作者对于项目管理的理解很深入。对于风险管理的理解很深入。
- 4、没有经验，只能大概看看有个印象了
- 5、里面的故事值得一读
- 6、故事写得好，可以去写小说了；的确仅仅是思考，并未给出具体的解决方案；“项目管理的本质是服务”这个观点比较新颖。
- 7、当初抱着想了解点实际软件开发过程从而看能不能找到点想法的动机借了这本书，后面就没仔细看了，走马观花了一番，感觉一般。虽然在中途就不想看了，可是由于自己有看书就要看完不然总感觉不对劲的习惯，于是草草结案了，也许这是个坏习惯。因为似乎没有把时间花在更重要的事情上。
- 8、这本书讲述了一个故事，这个故事涉及到软件工程的方方面面，作者把自己工作中的感悟和思想非常巧妙的融入到故事的各个情节场景中。通过这个故事反观当前个人所面临的迷茫、困惑、软件工作中的一些场景，给我带来很多启发。对书中部分章节能与作者表述的思想产生共鸣，更多的是学习作者对软件工程中各场景的深刻思考，在我翻豆瓣的时候很巧合的把它买下来，很幸运在我这个阶段能读到这本书。
- 9、小说写得蛮有趣的，期待续集
- 10、杀书头！
- 11、我一般给星标的标准是看一本书值不值得读，收获多少。简单之美这本书可能可以算在我今年读书列表中对我触动最深的前三名内了。我最不喜欢的章节是软件测试那章，内容有点浮；其余章节部分的观点我也持保留意见。但是这些都不能否认这本书是作者IT生涯的诚恳思考和反省，这些都是非常珍贵的东西。我觉的，如果是一个带有一点理想主义的IT业者，这本书都是值得去好好读一读的。P.S.2013年的最后一天的中午读完，有种赶上晚班车的愉快感.....
- 12、这么难得的一本书，窃喜，哈哈。  
对了，怎么关注的人不多，可惜了。
- 13、比预想的要糟糕啊
- 14、一般
- 15、一个项目寄托多少哀思
- 16、视角独特，有感而发，对很多软件从业人员眼中司空见惯的现象和问题进行了分析。特别是第六章关于测试的章节，张弛有度，节奏感十足！
- 17、启迪思维、引人深思
- 18、很值得一读。相信具有几年软件开发经验的人员去读会更有共鸣吧。
- 19、第五章内容，才读了两节。第一节通过孔如之（Ralph）和林峰在TFC项目争论最后引出了“标准化在项目中的重要性”的结论，同时也能看到林峰成长，但林峰现在还是很迷茫，对自己的定位还不是很清晰。一方面有弱爆了的成实捧他是个技术大咖，另一方面在Ralph面前瞬间被秒成技术渣，总觉得自己不能完全施展开。至于王蓉被Ralph试探问有木有男朋友，难道还有点感情戏？在男作者写的小说里，女性角色总是觉得缺了点什么。第二节主要讲模型设计，就是对领域模型、计算模型做了详尽的解释描述，最后引出了云计算中的PaaS应用场景。这两节涉及软件生产过程各个方面。和本书其他章一样，都是在反思中感悟，在批判中立论。第二节干货较多，总之，领域模型和计算模型设计对于软件项目来说非常重要。
- 20、软件开发的系统性思维
- 21、这本书针对软件开发流程，从最开始的需求分析到最后的资产积累做了系统的讲解，每章都有亮点，观点新颖独到有理有据令人信服，可以看出作者有丰富的软件开发管理经验。尤其是需求分析和项目管理更是令人赞赏。奇怪的是评价两极分化严重，但我很认可，以至于惊讶这是一个中国人写的书。
- 22、讨厌java出来的思维
- 23、哈哈 作者谈女人与两个男人的肉体关系 莫非有经历的？

## 《简单之美》

- 24、看了网上有的2个章节试读的pdf。项目管理那节写得挺有新意，“管理即服务”这个提法有那么点意思。回头去图书馆借来再扫一下。
- 25、观点客观、结合实践、毫不偏颇，都是一个多年的软件从业者的真心体会，虽没有很多具体的方法指导，但看完很受启发，每一个深受困扰中国式开发模式困扰的管理者都应该读读。
- 26、一只老鸟的行业感悟，因为相似的背景，很多地方有共鸣。
- 27、值得推荐，做一个善于思考的IT从业者！
- 28、1.这是一次软件开发者的的心灵沟通之旅 2.10大社区一致鼎力推荐 3.探索软件开发与管理的精髓与本质：简单、想象和文化 4.从思想出发，以实践为证，直奔问题的根源，挖掘简单的本质 5.与方法论无关，与具体技术无关，与软件开发的思想和哲学有关

1、本书应该适用于那些在实际工作中陷入迷惘或存在困惑的软件开发、管理者和企业经营者。推荐！（来自互动）

2、本书的内容有些另类，绝不似书名所呈现的中规中矩，但确实体现了一种美，是一种简单到极致的优雅，似乎又繁复如星空般的深邃，包容如峭立千仞之高的山壁。这是一本可以称之为轻松加愉快的思想随笔，又是一篇如杜拉拉升职记般的职场小说，它还贯穿了整个软件开发过程，揭露了从方法论、需求、架构设计、编码实现，到测试与维护以及团队管理的诸多要诀。这正是本书的另类之处。我在阅读本书时，情不自禁地被放在书中每章篇首的实践场景所吸引，甚至忽略了本书的重要内容，直接根据提示转到下一个章节的实践场景，一气呵成，直到将这些实践场景阅读完毕。作者挥洒自如的文笔，入木三分的人物刻画，以及细腻含蓄的情感描写，将我彻底吸引住了。在大结局中，孔如之与儿子在阳光中巴黎圣母院前的对话，让人意犹未尽，似乎满怀希望，却又历尽沧桑，真是让我产生“情何以堪”的感慨。这是本书感性的一面了。只是看完这9篇由实践场景片段组成的小说，就已经值回票价了。而从技术书籍的角度来看，本书的意义显然并不在于此，作者完整地勾勒出软件开发的全貌，诸多感悟与体会都可以成为软件开发人员的重要借鉴。作为本书理性的一面，这些内容需要反复阅读和分析琢磨，才能引起你的共鸣，许多模糊在心头的概念，在作者简明扼要的叙述下，或许就会产生“拨开云层见月明”的感悟。以本书第4章为例。作者给出了一个简单的实例描述了框架构建的过程。首先从背景描述出发，展现了对保险业务中对保单进行处理的需求功能。这段背景描述将复杂的保险需求阐述得非常清楚而富有条理，体现了作者撰写文档的高超能力。接下来是作者对这一背景描述的抽象。这段抽象有理有据，较好地体现了从需求捕获到分析的过程演变，利用抽象搭建了基本的领域模型。紧接着是对约束的思考，这是架构师必须完成的工作。根据对需求的抽象和关于约束的思考，就能够做出合理的架构决策。作者在本书中反复强调的“使用自然语言和讲故事的方式”，通过实例得到了具体的展现。事实上，在Joel on software一书中，Joel Spolsky也提出了同样的观点，认为通过讲故事的方式描述用例场景，可以更好地促进理解与交流。本书作者扩大了这种方式的应用范围，引入到架构设计过程中。对这一做法，我深表赞同。事实上，我在架构过程中，也常常采用类似手法，通过在文档撰写设计的故事场景，帮助我梳理设计思路，有时候，甚至在文档中自问自答，在这样的编写过程中我慢慢找到了解决方案。本书对领域模型的讨论也有着个人独到的见解。例如他对静态模型和动态模型分类，又例如他提出了使用贫血模型的好处。在Martin Fowler提出贫血模型之后，业界曾经掀起过对贫血模型与充血模型的争论。然而，争论到了最后，也没有一个确切的结果。从经典的OO原则来看，它要求将对象的数据和行为组织在一起，这正是批判贫血对象的主要论据。我比较倾向于这个观点，认为对象没有行为，就是“死”的，缺乏自治的能力。但在实际开发过程中，我也常常体会到贫血模型的好处，尤其是在模型重用与解耦方面，贫血模型都有其显著的优势。本书作者认为，贫血模型的“第一个好处是，有利于信息交换。第二个好处是，清晰了对象的职责。第三个好处是，实体对象（贫血对象）的实现更加灵活。第四个好处是，可以确保实体对象（贫血对象）只能在内存中用于计算。”这些好处都说到了点子上。虽然，我对于贫血对象的使用仍然抱有谨慎态度，但本书对此的阐述依旧给我提供了不错的参考。在项目管理方面，书中强调了“负责制度”的实施。这首先关系到责任定位的问题。项目延迟或失败，究竟是项目管理的问题，还是架构设计的问题？是编码实现的问题，还是测试维护的问题？作者认为，负责制度的缺失可能会影响项目的质量。书中提到：“在软件开发过程中，人是最重要的因素，而责任、权利和利益是保证这个因素发挥作用的关键。”“建立负责制度的目的，不是为了惩罚某人，也不是为了永久取消某人的职业发展权利，它只是通过责任人利益损失的形式，来表明这样一个事实：没有金刚钻，别揽瓷器活。”事实上，负责制度的关键不在于制度的确立，而在于执行。如果没有创建公平、公开、公正的执行环境，这种制度只会给软件开发带来负面影响。这也是作者仅仅提出问题，却没有给出好的答案原因所在。相对而言，我个人更倾向于Scrum“回顾会议”，在基于迭代与渐进式开发的基础上，这种方式更能够有效解决项目开发中存在的问题。倘若是新手阅读本书，由于缺乏足够的工作阅历与开发经验，很难理解作者写作的意图。但我们绝对不能因为这种认识上的障碍，而将本书拒之千里之外。事实上，越早阅读本书，越能够开拓读者的眼界，提前感受业界的真实与谎言，反而能够帮助新手更快地确立自己的职业生涯规划。对于混迹行业多年的老鸟而言，阅读本书，一定能找到那些似曾相识的画面。作者对技术的深入探讨，也一定能给予我们启发，即使观点不同，也可以求同存异。所谓“嘤其鸣矣，求其

## 《简单之美》

友声”，这是我在阅读本书时收获的如遇旧友般的快乐！本文转自著名网站InfoQ

：<http://www.infoq.com/cn/news/2011/04/beauty-of-simple>评论人：张逸国内著名的软件架构师、敏捷导师、微软最有价值专家、MUSP培训专家张逸先生，著有《软件设计精要与模式》和翻译了《WCF服务编程》一书，在业界德高望重。关于本书的更多信息：<http://tinyurl.com/68xwxfd>

3、很多软件开发人员，在实践中经常处于一种混乱状态——出现问题的时候，没有人知道怎么做才算最好，或者大家缺乏足够的信任和共识，又或者不断地讨论却得不到满意的解决方案。这本书可以帮助我们摆脱混乱状态。（来自卓越）

4、通过方法或手段来提高软件开发人员的主动性是错误的。从理想主义者的角度来看，软件开发人员和企业之间的关系是平等的，他们以一种契约的形式，相互服务于对方。一切都建立在“内驱力”之上！（来自当当）

5、我看非常不错，不像有些人说的那么差劲。我比较认同第8.2节中的两个观点：“项目管理工作的基本思路不是控制，而是创造环境和顺势引导。”“把项目管理和软件开发工作清晰地分开，是一种最佳的实践。”这种总结应该和我一样来自教训吧...麻烦个别tx把书看完再发表意见哈，不要误导我们。（来自卓越）

6、用现在比较流行的话来说，本书中对软件开发和管理提出的一些观点都非常犀利，反映了作者的深入思考和深刻见解，很多朋友都非常赞同。尤为值得一提的是，作者为本书虚构的一个大型故事案例（分散在各章，寓意非常深刻）吸引了很多的读者朋友，可读性和趣味性比《杜拉拉升职记》还要好。关于本书，有个简单的讨论帖，感兴趣的朋友可以看看：<http://www.javaeye.com/topic/659958>

7、褒：作者编故事的能力还是不错的，看完目录、前言，我就先把每章开始的故事连起来读了一遍，不得不说，很有《圈子圈套》的意味，好好发挥下，超越之绝对不是问题，那么多憋闷、抑郁的coder，需要很多可以yy的故事的，至少，本人读完是很happy的，如同人家描述武侠故事，可以很好的满足中国大男人的某些心态的~方法论、人的重要性及团队建设、项目管理相关论述，还是有些受益的。-----我也有分割线了~-----贬：目录大结构还算清晰简单，但是每一章的正文，总是读的艰难，想为其画个思维导图，总是不得要领，全书立意有点太过庞大，书中不乏精辟之语，但总感到不成系统，你觉得说的都对，崇尚简单美的一本书，却让人有繁杂之感，遗憾。这让我想起了高中时写作文的场景，或纠结于情绪的表达、或充斥着东拉西拽的名言和精辟小故事，本书似有此感。更让人汗颜的是，本书66~67页中间撕掉了两页，多塞了两张也不撕的干净点！

## 章节试读

### 1、《简单之美》的笔记-关于项目管理的思考

特别喜欢作者第8章对项目管理的一番论述，甚合我意！

# 《简单之美》

## 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:[www.tushu000.com](http://www.tushu000.com)