第一图书网, tushu007.com

<<软件架构师应该知道的97件事>>

图书基本信息

书名：<<软件架构师应该知道的97件事>>

13位ISBN编号：9787121106354

10位ISBN编号：7121106353

出版时间：2010年4月

出版时间：电子工业出版社

作者：Richard Monson-Haefel 编

页数：200

译者：徐定翔,章显洲

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

前言

97个水晶切面，折射软件架构师的实践智慧　　近十数年，全球软件产业已经取得了显著的进步，软件从业人员的数量不断增加，软件项目的规模和复杂度不断攀升，软件开发组织的人员结构也因专业不断细分而日趋复杂。

因兴起时间较晚，发展的时间相对还很短，软件业从建筑业和制造业等其他成熟的工程学科中借用了许多概念和隐喻。
“软件架构”和“软件架构师”，便是借用了建筑设计中的概念。
从UML（统一建模语言，Unified Modeling Language）和软件模式（Pattern）相关的早期著作中，可以清晰地看到这种概念移用。

借用其他行业的概念会带来一些消极影响。
比如在早期阶段，一些“软件工程”研究人员和软件项目管理人员，试图以建筑业的项目管理视角和技术来管理软件项目，制造出了“瀑布式”软件开发过程，给软件业带来了长期的不良影响。
但是，经过实践验证，“软件架构”确实推动了软件开发技术的发展，为业界广为采纳并日益受到重视。

现如今，“软件架构师”已成为许多软件开发组织职位模型（Job Model）中的标准设置。
“软件架构师”也已成为众多软件开发人员梦寐以求的职业巅峰目标之一。

业界许多大师和专家总结分享了他们在软件架构设计方面的技术和经验。
在软件技术书籍市场上，已经可以找到许多与“软件架构”主题相关的书籍。

不过，现有的软件架构书籍从技术视角进行总结阐述的居多，如已有5卷本行世的《基于模式的软件架构 (Pattern-Oriented Software Architecture，POSA)》，主要阐述的是软件架构的风格、解决特定问题或特定领域的软件架构设计技术与模式等。
这些书籍当然都非常有益，甚至，可以说成为了软件架构师的必读书目。

但是，即使已经将这些软件架构设计的技术、模式烂熟于胸，可能还无法保证你能够成为优秀的软件架构师。

为何会这样？

正如《建筑十书》的作者、古罗马著名建筑师维特鲁威所说：“理想的建筑师应该既是文学家又是数学家，他还应通晓历史，热衷于哲学研究，精通音乐，懂得医药知识，具有法学造诣，深谙天文学及天文计算。
”　　优秀的软件架构师，必须多才多艺、成熟练达、经验丰富，具备极强的洞察力，能够领导和提升软件开发团队，去构建整齐有序、均衡合理、可持续发展演化的虚拟数字世界中的伟大建筑——优秀杰出的软件产品。

本书提供了分享软件架构知识的新方式，拓宽了阐述软件架构艺术的视角，总结了50多位经验丰富的软件架构师的实践经验，范围覆盖了软件架构师的职业操守、技术技能、思维模式、领导力、和客户的沟通交流、权衡利弊的平衡感等主题。

全书由97篇格言式散文构成，没有高调的说教，没有抽象的术语，而是以平实、幽默、智慧的笔触，将他们认为对成为优秀软件架构师而言至为重要的精髓和盘托出。
全书犹如一块玲珑剔透的水晶，97个切面折射出来的都是出自一线软件架构师的专业智慧。

本书第1~49篇由徐定翔翻译，第50~97篇由章显洲翻译。
我们在翻译的过程中得到了许多人的帮助。
余晟认真审校了全书的译稿，细致地指出了我们翻译不当之处；郑兆昭老师对书稿的润色让我们受益匪浅；统稿编辑白爱萍一次又一次容忍了我们修改定稿的要求。
最后要感谢家人给予我们的宽容和支持，使我们能够安心埋首于书稿中。

内容概要

优秀的软件架构师应该同时掌握业务知识和技术能力，做到这一点绝非易事，《软件架构师应该知道的97件事》想要探讨的就是这个主题。
这是一本真正的开源图书，我们邀请到50多位杰出的软件架构师参与写作。
大家无偿地分享了各自的工作经验和心得，内容从规避风险的方法到组建团队的技巧，涵盖了架构设计的方方面面。
衷心希望这97篇文章能激发您的思考，解决您工作中的困惑。

作者简介

理查德·蒙森-哈斐尔，是独立软件开发者，曾参加编写《Enterprise JavaBeans》和《Java Message Service》（均由OReilly公司出版）。
他是企业计算方面的专家，同时擅长设计和开发多点触控应用程序。

书籍目录

前言客户需求重于个人简历简化根本复杂性，消除偶发复杂性关键问题可能不是出在技术上以沟通为中心，坚持简明清晰的表达方式和开明的领导风格架构决定性能分析客户需求背后的意义起立发言故障终究会发生我们常常忽略了自己在谈判量化需求一行代码比五百行架构说明更有价值不存在放之四海皆准的解决方案提前关注性能问题架构设计要平衡兼顾多方需求草率提交任务是不负责任的行为不要在一棵树上吊死业务目标至上先确保解决方案简单可用，再考虑通用性和复用性架构师应该亲力亲为持续集成避免进度调整失误取舍的艺术打造数据库堡垒重视不确定性不要轻易放过不起眼的问题让大家学会复用架构里没有大写的“I”使用“一千英尺高”的视图先尝试后决策掌握业务领域知识程序设计是一种设计让开发人员自己做主时间改变一切设立软件架构专业为时尚早控制项目规模架构师不是演员，是管家软件架构的道德责任摩天大厦不可伸缩混合开发的时代已经来临性能至上留意架构图里的空白区域学习软件专业的行话具体情境决定一切侏儒、精灵、巫师和国王向建筑师学习避免重复欢迎来到现实世界仔细观察，别试图控制一切架构师好比两面神架构师当聚焦于边界和接口助力开发团队记录决策理由挑战假设尤其是你自己的分享知识和经验模式病不要滥用架构隐喻关注应用程序的支持和维护有舍才有得先考虑原则、公理和类比再考虑个人意见和口味从“可行走骨架”开始开发应用数据是核心确保简单问题有简单的解架构师首先是开发人员根据投资回报率（ROI）进行决策一切软件系统都是遗留系统起码要有两个可选的解决方案理解变化的影响你不能不了解硬件现在走捷径，将来付利息不要追求“完美”，“足够好”就行小心“好主意”内容为王对商业方，架构师要避免愤世嫉俗拉伸关键维度，发现设计中的不足架构师要以自己的编程能力为依托命名要恰如其分稳定的问题才能产生高质量的解决方案天道酬勤对决策负责弃聪明，求质朴精心选择有效技术，绝不轻易抛弃客户的客户才是你的客户！
事物发展总会出人意料选择彼此间可协调工作的框架着重强调项目的商业价值不仅仅只控制代码，也要控制数据偿还技术债务不要急于求解打造上手（Zuhanden）的系统找到并留住富有激情的问题解决者软件并非真实的存在学习新语言没有永不过时的解决方案用户接受度问题清汤的重要启示对最终用户而言，界面就是系统优秀软件不是构建出来的，而是培育起来的索引

章节摘录

客户需求重于个人简历　　作为工程师，我们常常要向客户推荐技术、手段，甚至方法论来解决问题。
但有时我们心里不是想寻求解决问题的最佳方案，而是希望借此丰富自己的简历。
这样做很可能得不偿失。
积累一批满意的客户，选择切合实际的技术解决他们的难题，让他们乐于推荐你，才是最好的履历。
信誉远胜过时髦的编程技巧和流行的范式。
掌握最新的技术趋势，与时俱进固然重要，但不能让客户为此买单。
作为架构师，职业操守绝不能忘。
公司托付重任给你，是期望你恪尽职守，不受利益诱惑。
如果你觉得项目不够尖端，挑战性不足，无法满足职业发展的需要，大可另栖高枝，另谋高就。
万一你别无选择，必须参与这样的项目，不要为简历所累。
忍痛割爱放弃时髦光鲜的方案确实不容易（哪怕它们并不适合当前的项目），但只有脚踏实地替客户着想，最后才能皆大欢喜。
选择正确的解决方案可以降低项目的压力，团队工作起来更开心，客户也更满意。
你会有更充裕的时间，既可以钻研现有技术，也可以利用空闲时间学习新知识，甚至重拾向往已久的业余爱好。
家人察觉你的变化后，也会感到欣慰。
把客户的长远需求摆在自己的短期利益之上，才能立于不败之地。

编辑推荐

《软件架构师应该知道的97件事》是本与众不同的技术图书。
五十多位作者中不乏像尼尔·福特（Neal Ford）、迈克尔·尼加德（Michael Nygard）、比尔·德·霍拉（Bill de h6ra）这样杰出的软件架构师，大家分享了多年积累的开发经验和工作准则．内容不限于单纯的技术范畴．还涉及如何与各方沟通、如何降低项目的复杂度、怎样强化开发团队等。
有代表性的　　主题包括：　　客户需求重于个人简历　尼廷·博万卡（NitiN Borwa Borlkar）。
关键问题可能不是出在技术上一马克·兰姆（Mark Ra Ramm）。
以沟通为中心，坚持简明清晰的表达方式和开明的领导风格一一马克·理查兹（Mark Rictlards）
。
先确保解决方案简单可用，再考虑通用性和复用性　　凯佛林·亨尼（Kevlin Henney）。
对最终用户而言，界面就是系统　维纳亚克·赫格德（Vinayak Hegde）。
提前关注性能问题一丽贝卡·帕森斯（Rebecca Parsorls）。
成为出色的软件架构师要既掌握业务知识又具备技术能力。
顶尖的软件架构师看重什么7他们如何完成项目7怎样提高自己的工作水平7推荐您阅读《软件架构师应该知道的97件事》。

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