

原书第8版

Mc
Graw
Hill
Education

软件工程

实践者的研究方法

[美] 罗杰 S. 普莱斯曼 (Roger S. Pressman) 著
布鲁斯 R. 马克西姆 (Bruce R. Maxim) 著
郑人杰 马素霞 等译

Software Engineering
A Practitioner's Approach Eighth Edition

Eighth Edition

Software Engineering A PRACTITIONER'S APPROACH

Roger S.
PRESSMAN

Bruce R.
MAXIM



机械工业出版社
China Machine Press

计 算 机 科 学 丛 书

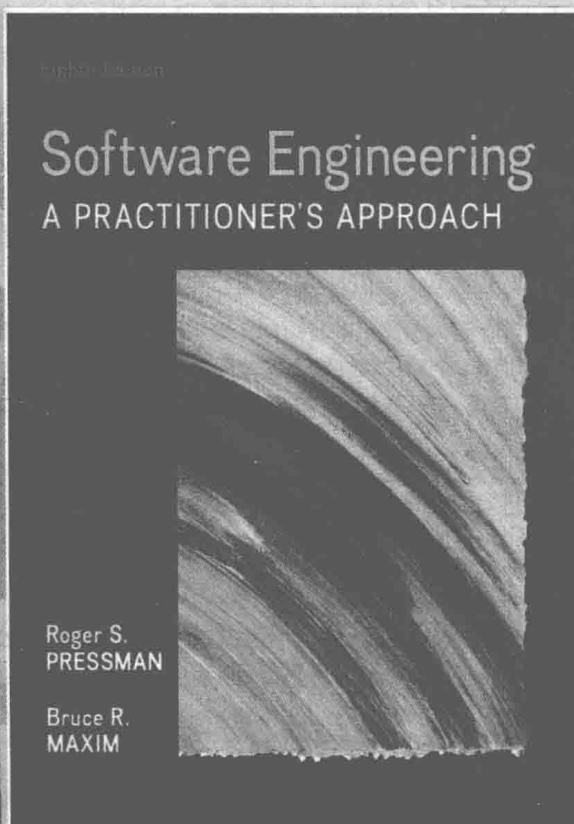
原书第8版

软件工程

实践者的研究方法

[美] 罗杰 S. 普莱斯曼 (Roger S. Pressman) 著
布鲁斯 R. 马克西姆 (Bruce R. Maxim) 著
郑人杰 马素霞 等译

Software Engineering
A Practitioner's Approach Eighth Edition



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

软件工程: 实践者的研究方法 (原书第 8 版) / (美) 罗杰 S. 普莱斯曼 (Roger S. Pressman), (美) 布鲁斯 R. 马克西姆 (Bruce R. Maxim) 著; 郑人杰等译. —北京: 机械工业出版社, 2016.9

(计算机科学丛书)

书名原文: Software Engineering: A Practitioner's Approach, Eighth Edition

ISBN 978-7-111-54897-3

I. 软… II. ①罗… ②布… ③郑… III. 软件工程 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2016) 第 226945 号

本书版权登记号: 图字: 01-2014-4760

Roger S. Pressman, Bruce R. Maxim: Software Engineering: A Practitioner's Approach, Eighth Edition (978-0-07-802212-8).

Copyright © 2015 by McGraw-Hill Education.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including without limitation photocopying, recording, taping, or any database, information or retrieval system, without the prior written permission of the publisher.

This authorized Chinese translation edition is jointly published by McGraw-Hill Education and China Machine Press. This edition is authorized for sale in the People's Republic of China only, excluding Hong Kong, Macao SAR and Taiwan.

Copyright © 2016 by McGraw-Hill Education and China Machine Press.

版权所有。未经出版人事先书面许可, 对本出版物的任何部分不得以任何方式或途径复制或传播, 包括但不限于复印、录制、录音, 或通过任何数据库、信息或可检索的系统。

本授权中文简体字翻译版由麦格劳-希尔(亚洲)教育出版公司和机械工业出版社合作出版。此版本经授权仅限在中华人民共和国境内(不包括香港、澳门特别行政区及台湾地区)销售。

版权 © 2016 由麦格劳-希尔(亚洲)教育出版公司与机械工业出版社所有。

本书封面贴有 McGraw-Hill Education 公司防伪标签, 无标签者不得销售。

本书自第 1 版出版至今, 30 多年来在软件工程界产生了巨大而深远的影响。第 8 版不仅加入了移动 App 项目等与时俱进的内容, 而且调整了篇章结构, 更利于教师针对不同课程进行选择。同时, 第 8 版继承了之前版本的风格与优势, 全面且系统地讲解软件过程、建模、质量管理、项目管理等基础知识, 涵盖相关概念、原则、方法和工具, 并且提供丰富的辅助阅读资源和网络资源, 指导有兴趣的读者进行更深入的学习和研究。

本书适合作为软件工程相关专业高年级本科生和研究生教材, 也可供软件专业技术人员和管理人员阅读参考。

出版发行: 机械工业出版社(北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 曲 熠

责任校对: 殷 虹

印 刷: 北京市荣盛彩色印刷有限公司

版 次: 2016 年 11 月第 1 版第 1 次印刷

开 本: 185mm × 260mm 1/16

印 张: 43

书 号: ISBN 978-7-111-54897-3

定 价: 99.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光/邹晓东

文艺复兴以来，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的优势，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅肇划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自1998年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与Pearson, McGraw-Hill, Elsevier, MIT, John Wiley & Sons, Cengage等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出Andrew S. Tanenbaum, Bjarne Stroustrup, Brian W. Kernighan, Dennis Ritchie, Jim Gray, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Abraham Silberschatz, William Stallings, Donald E. Knuth, John L. Hennessy, Larry L. Peterson等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力相助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专门为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方式如下：

华章网站：www.hzbook.com

电子邮件：hzsj@hzbook.com

联系电话：(010) 88379604

联系地址：北京市西城区百万庄南街1号

邮政编码：100037



华章科技图书出版中心

译者序

Software Engineering: A Practitioner's Approach, Eighth Edition

本书是国际知名软件工程专家罗杰 S. 普莱斯曼 (Roger S. Pressman) 的最新著作。自 35 年前第 1 版问世以来, 这本书在软件工程界产生了巨大而深远的影响。其权威性是无可置疑的, 在培养软件工程专业人才方面所起的作用也是显而易见的。在这一版中, 新加入的布鲁斯 (Bruce) 作为第二作者参与了本书的编写工作。

我自 20 世纪 80 年代中期开始从事高校软件工程方面的教学与科研工作, 多年来, 这本书的各个版本一直是我的重要参考书, 它给了我许多启发和帮助, 我也曾多次向许多业界好友和学生推荐此书。

如今基于计算机的系统已经广泛而深入地渗透到经济、国防和人们日常生活的各个领域, 特别是在互联网的推动下, 不仅许多行业得以改进和更新, 而且产生了一批新的行业, 展现了全新的业态。我们必须意识到, 在计算机不断向社会的深度和广度层面发展的过程中, 软件始终处在系统的核心地位, 起着中枢和灵魂的作用, 而且这种作用正日益突出。因此, 如何为现代化系统配备合格和优良的软件也就更加受到人们的广泛关注。

本书系统地论证了软件工程领域的基本知识和最新研究成果, 包括新的概念、原则、技术、方法和工具。同时书中还为读者提供了进一步学习和研究的线索, 包括许多可供利用的网上资料和信息。与第 7 版相比, 本版继承了一些优点, 同时也做了一些改动、扩充和更新。

本书特点

1. 全书内容分为五个部分, 共 39 章, 还包括三个附录。五个部分的内容分别为软件过程、建模、质量管理、管理软件项目以及软件工程高级课题。
2. 本书继承了前一版的特色, 突出了软件质量管理的内容, 同时也加强了软件过程部分。此外, 增加的内容还包括: 软件工程项目对人员的要求; 近年来软件产业发展中出现的新课题——移动 App 和软件系统安全性。
3. 仍然在各章的开头给出“要点浏览”(包括概念、人员、重要性、步骤、工作产品和质量保证措施)以及“关键概念”(全章内容的关键词)。
4. 仍然在各章的末尾给出“小结”“习题与思考题”以及“扩展阅读与信息资源”, 这些都非常适合有兴趣、有需要的读者沿着所提供的线索开展进一步的学习和研究。
5. 仍然保留了本书历次版本在各章中为读者提供的多种形式的辅助阅读信息, 可以说这是本书的一个突出特点。这些信息从形式上分为两类: 一类是采用通栏形式的说明框, 包括要点浏览、信息栏、软件工具和 SafeHome 对话框等; 另一类是居于页面右侧的说明框, 包括关键概念、引述、建议、关键点、提问和网络资源。这些说明框非常有益于读者理解和进一步探索相关内容。

读者对象

本书仍然面向三类读者, 即高校学生 (特别是研究生)、教师和专业软件技术人员。总

体而言,本书适合作为高校计算机或信息技术相关专业的教学用书,特别适合为软件工程课程提供教学服务。

对于采用本书作为教材的教师,在此提供几点建议:

1. 由于学时所限,不可能将本书的全部内容纳入教学,因此从中选取适合的部分是必然的。可以对软件管理部分做一些压缩,但即使如此,我认为也不可把有关管理的内容完全删除。
2. 近年来软件工程领域出现了许多新技术和新方法,作者已将其及时引入书中。但请教师注意,对于初学者来说,牢固地掌握基本概念、基本技能和方法仍然十分重要。

译者说明

参与本书翻译工作的译者以华北电力大学和清华大学的教师为主,也有少数软件企业和中国软件行业协会的研究人员。他们是:马素霞(第1~4章、22~25章、28章及附录1~3)、宋兰(第7~11章及29章)、石敏(第12、13章)、周长玉(第14、15章)、吴爽(第16~18章)、韩新启(第19~21章)、王海青(第30章及35~39章)、王素琴(第31~34章)。此外,刘瑾完成了第5、6章的翻译工作。我负责第26、27章以及前言和作者简介部分。在翻译过程中,我们得到了华北电力大学控制与计算机工程学院洪海、熊里、赵敏、李树超、高晶晶、吕骁同学的帮助,在此对他们的辛勤劳动表示感谢。我对全部译稿、马素霞教授对大部分译稿做了仔细审核与修改,并更正了原书中个别的错漏之处。

本书英文版有900多页,翻译工作量巨大,而译者均有繁重的本职工作,时间并不宽松,因此译文中难免有不当之处,敬请读者见谅并不吝指正。

总之,这是一本非常优秀的软件工程读物,本人十分高兴地向国内读者推荐。我们相信,认真阅读它,定会使你获益匪浅。

郑人杰

2016年7月

如果有这样一款计算机软件，它能满足用户的需求，能在相当长的时间内无故障地运行，修改起来轻松便捷，使用起来更是得心应手，那么，这款软件必定是成功的，它切实改善了我们的生活。但是，如果有这样一款软件，它令用户失望，错误频出，修改起来困难重重，使用起来更是举步维艰，那么，这必定是一款失败的软件，它使我们的生活一团糟。谁都希望开发出优秀的软件，为我们的生活带来便利，而不是把自己陷入失败的深渊。要想使软件获得成功，在设计和构建软件时就需要有规范，需要采用工程化的方法。

自本书第1版问世以来的近35年中，软件工程已经从少数倡导者提出的一些朦胧概念发展成为一门正规的工程学科，已被公认为是一个值得深入研究、认真学习和热烈讨论的课题。在整个行业中，软件工程师已经代替程序员成为人们优先选择的工作岗位，软件过程模型、软件工程方法和软件工具都已在全行业的所有环节成功采用。

尽管管理人员和一线专业人员都承认需要有更为规范的软件方法，但他们却始终在争论应该采用什么样的规范。有许多个人和公司至今仍在杂乱无章地开发着自己的软件，甚至即使他们正在开发的系统要服务于当今最为先进的技术，状况也仍是如此。许多专业人员和学生并不了解现代方法，这导致他们所开发的软件质量很差，因而造成了严重的后果。此外，有关软件工程方法真实本质的争论一直持续进行着。软件工程的地位问题已成为一门对比研究课题。人们对软件工程的態度已经有所改善，研究工作已取得了进展，不过要成为一门完全成熟的学科，我们还有大量的工作要做。

我们希望本书能够成为引导读者进入正在走向成熟的软件工程学科的入门读物，和以前的7个版本一样，第8版对学生和专业人员同样具有很强的吸引力。它既是软件专业人员的工作指南，也是高年级本科生和一年级研究生的综合性参考书。

第8版中包含了许多新的内容，它绝不只是前一版的简单更新。这一版不仅对内容做了适当的修改，而且调整了全书的结构，以改进教学顺序；同时更加强调一些新的和重要的软件工程过程和软件工程实践知识。此外，本书进一步加强了“支持系统”，为学生、教师和专业人员提供了更为丰富的知识资源。读者可访问专门为本书建立的网站（www.mhhe.com/pressman）查阅这些信息。

篇章结构

本书共39章，分为五个部分。这种划分有利于那些无法在一个学期内讲完全书内容的教师灵活安排教学。

第一部分“软件过程”给出了有关软件过程的各种不同观点，讨论了所有重要的过程模型，还涉及惯用过程和敏捷过程在指导思想上的分歧。第二部分“建模”给出了分析方法和设计方法，重点讲解面向对象方法和UML建模，同时也介绍了基于模式的设计以及基于WebApp和移动App的设计。第三部分“质量管理”介绍了有关质量管理的概念、规程、技术和方法，使得软件团队能够很好地评估软件质量，评审软件工程项目产品，实施软件质量保证规程，并正确地运用有效的测试策略和战术。此外，这一部分还讨论了形式化建模和验

证方法。第四部分“管理软件项目”介绍了与计划、管理和控制软件开发项目的人员有关的问题。第五部分“软件工程高级课题”讨论了软件过程改进和软件工程的发展趋势。

第8版沿用了前面几个版本的做法，在各章中都提供了大量的辅助阅读信息，包括一个虚拟软件团队在工作中遇到困难时展开的对话，还包括对各章相关知识给出的补充方法和工具。

第8版中五个部分的划分有利于教师根据学时和教学要求安排课堂内容。在一个学期内可以安排一个部分的内容，也可以安排多个部分的内容。软件工程概论课程可以从五个部分中选择若干章作为教材。侧重分析和设计的软件工程课程可以从第一部分和第二部分中选取素材。面向测试的软件工程课程则可以从第一部分和第三部分中选取素材，还应加上第二部分中的一些内容。侧重管理的课程应突出第一部分和第四部分的内容。我们用上述方式组织第8版的内容，意在给教师提供多种教学安排的选择。但无论如何选择这些内容，都可以从“支持系统”中获得补充资源。

相关资源[⊖]

学生资源

本书为学生提供的各种学习资料包括：在线学习中心提供的各章学习指南，实践测验，题解以及多种在线资源（软件工程检查单、一套正在演化的微型工具、综合案例研究和工件产品模板等）。此外，1000多种网络参考文献可供学生更深入地探究软件工程问题，还有500多篇可下载的参考文献，这些都为读者提供了关于高级软件工程课题的更为详尽的信息。

教师资源

本书为教师提供的各种教学资料包括：在线（也可下载）教师指南，由700多个PPT组成的教辅资源和试题库。当然，学生资源（如微型工具、网络参考文献及可下载参考文献）和专业人员资源也可供教师使用。

在本书的教师指南中，我们为各种类型的软件工程课程提出了建议，介绍了与课程配合开展的软件项目、部分问题的题解和许多有用的教学辅助工具。

专业人员资源

本书为产业界专业人员（也包括在校学生）提供的各种资料包括：软件工程文档和其他工作产品的大纲和模板，一套有用的软件工程检查单，软件工程工具目录，综合性网络资料以及用于软件工程过程具体任务划分的“通用过程模型”。

由于有了在线支持系统的配合，使得本书既有内容上的深度，又有一定的灵活性，这些优势是传统的教科书所无法比拟的。

布鲁斯·马克西姆（Bruce Maxim）为这一版贡献了新的力量，他不仅具备丰富的软件工程知识，还带来了新的内容和洞见，这些对于读者来说将是十分宝贵的。

⊖ 关于本书资源，请访问 www.mhhe.com/pressman 查看，有需要的读者可向麦格劳·希尔教育出版公司北京代表处申请，电话：8008101936/010-62790299-108，电子邮件：instructorchina@mcgraw-hill.com。——编辑注

致谢

我们要特别感谢渥太华大学的 Tim Lethbridge，他帮助我们开发了 UML 和 OCL 的案例，以及配合本书内容的其他案例研究。Colby 学院的 Dale Skrien 开发了附录 1 的 UML 教辅资源。他们的帮助和意见都是十分宝贵的。此外也感谢高级软件工程师 Austin Krauss，他提供了关于视频游戏产业软件开发的宝贵意见。同时，要对为第 8 版评审做出贡献的几位教授表示感谢，他们是佛罗里达大学的 Manuel E. Bermudez、堪萨斯州立大学的 Scott DeLoach、密歇根州立大学的 Alex Liu 和犹他州立大学的 Dean Mathias。正是他们的详尽而真诚的评审意见帮助了我们，使得本书更加完善。

特别感谢

十分高兴有机会与罗杰合作，参与本书第 8 版的撰写工作。在此期间我的儿子 Benjamin 推出了他的第一款移动 App，我的女儿 Katherine 开始了她的室内设计生涯。我十分高兴地看到他们已经长大成人。同时非常感谢妻子 Norma，她热情地支持我，使我能够将所有空闲时间都投入本书的写作。

布鲁斯 R. 马克西姆 (Bruce R. Maxim)

随着本书各版本的不断推出，我的两个儿子 Mathew 和 Michael 也逐渐从小男孩成长为男子汉。他们在生活中的成熟、品格和成功鼓舞着我，没有什么比这更让我自豪了。他们现在也已经有了自己的孩子——Maya 和 Lily，这两个女孩已经是移动计算时代新智能设备方面的奇才。最后要感谢妻子 Barbara，她宽容我花费如此多的时间在办公室工作，并且还鼓励我继续写作本书的下一个版本。

罗杰 S. 普莱斯曼 (Roger S. Pressman)

罗杰 S. 普莱斯曼 (Roger S. Pressman)

普莱斯曼是软件工程领域国际知名的顾问和作家。40多年来，他作为工程师、经理人、教授、作家、咨询师和企业家始终奋战在这一领域。

普莱斯曼博士现任一家咨询公司 (R. S. Pressman & Associates, Inc.) 的总裁，该公司致力于协助企业建立有效的软件工程实践。这些年来，他已经开发了一套用于改进软件工程实践的技术和工具。他还是一家创业公司 (Teslaccessories, LLC) 的创始人，这家制造公司专门为特斯拉 Model S 系列电动车生产定制产品。

普莱斯曼博士是 9 本书的作者，其中包括两本小说。他还写了许多技术和管理方面的文章。他曾任《IEEE Software》和《The Cutter IT Journal》等行业杂志的编委，以及《IEEE Software》杂志“Manager”专栏的编辑。

普莱斯曼博士还是著名的演讲家，曾在许多重要的行业会议上做主题演讲，在国际软件工程会议和一些行业会议上做辅导讲座，并且一直是 ACM (美国计算机协会)、IEEE (美国电气与电子工程师协会) 以及 Tau Beta Pi、Phi Kappa Phi、Eta Kappa Nu 和 Pi Tau Sigma 等组织的成员。

布鲁斯 R. 马克西姆 (Bruce R. Maxim)

马克西姆博士 30 多年来曾任软件工程师、项目经理、教授、作家和咨询师。他的研究兴趣涉及软件工程、人机交互、游戏设计、社交媒体、人工智能以及计算机科学教育等领域。

马克西姆博士现任密歇根大学迪尔伯恩分校计算机与信息科学系副教授，他曾为该校工程与计算机科学学院建立游戏实验室。他曾经发表多篇有关计算机算法动画、游戏开发以及工程教育方面的论文。他还是畅销的计算机科学导论课本的作者之一。在密歇根大学工作期间，马克西姆博士曾监管了几百个产业界软件开发项目。

马克西姆博士的专业经验包括在医学院管理研究信息系统，为某医学校区指导计算教学，并承担统计程序员的工作。他还曾担任某游戏开发公司的首席技术官。

马克西姆博士是若干著名教学奖以及某著名社团组织服务奖的获得者。他还是 Sigma Xi、Upsilon Pi Epsilon、Pi Mu Epsilon、ACM、IEEE、美国工程教育协会、女工程师协会以及国际游戏开发者联盟等社会组织的成员。

目 录

Software Engineering: A Practitioner's Approach, Eighth Edition

出版者的话
译者序
前言
作者简介

第 1 章 软件的本质	1
1.1 软件的本质	3
1.1.1 定义软件	3
1.1.2 软件应用领域	4
1.1.3 遗留软件	5
1.2 软件的变更本质	6
1.2.1 WebApp	6
1.2.2 移动 App	7
1.2.3 云计算	7
1.2.4 产品线软件	8
1.3 小结	8
习题与思考题	8
扩展阅读与信息资源	9
第 2 章 软件工程	10
2.1 定义软件工程学科	11
2.2 软件过程	11
2.2.1 过程框架	12
2.2.2 普适性活动	12
2.2.3 过程的适应性调整	13
2.3 软件工程实践	13
2.3.1 实践的精髓	14
2.3.2 通用原则	14
2.4 软件开发神话	16
2.5 这一切是如何开始的	18
2.6 小结	19
习题与思考题	19
扩展阅读与信息资源	19

第一部分 软件过程

第 3 章 软件过程结构	22
3.1 通用过程模型	23
3.2 定义框架活动	24
3.3 明确任务集	24
3.4 过程模式	25
3.5 过程评估与改进	27
3.6 小结	28
习题与思考题	28
扩展阅读与信息资源	28
第 4 章 过程模型	29
4.1 惯用过程模型	30
4.1.1 瀑布模型	30
4.1.2 增量过程模型	32
4.1.3 演化过程模型	32
4.1.4 并发模型	36
4.1.5 演化过程的最终评述	37
4.2 专用过程模型	38
4.2.1 基于构件的开发	38
4.2.2 形式化方法模型	39
4.2.3 面向方面的软件开发	39
4.3 统一过程	40
4.3.1 统一过程的简史	41
4.3.2 统一过程的阶段	41
4.4 个人过程模型和团队过程模型	42
4.4.1 个人软件过程	42
4.4.2 团队软件过程	43
4.5 过程技术	44
4.6 产品和过程	45
4.7 小结	46
习题与思考题	46
扩展阅读与信息资源	47

第5章 敏捷开发	48	7.2 核心原则	77
5.1 什么是敏捷	49	7.2.1 指导过程的原则	78
5.2 敏捷及变更成本	50	7.2.2 指导实践的原则	78
5.3 什么是敏捷过程	50	7.3 指导每个框架活动的原则	80
5.3.1 敏捷原则	51	7.3.1 沟通原则	80
5.3.2 敏捷开发战略	52	7.3.2 策划原则	81
5.4 极限编程	52	7.3.3 建模原则	83
5.4.1 极限编程过程	52	7.3.4 构建原则	87
5.4.2 工业极限编程	54	7.3.5 部署原则	89
5.5 其他敏捷过程模型	56	7.4 工作实践	90
5.5.1 Scrum	56	7.5 小结	91
5.5.2 动态系统开发方法	57	习题与思考题	92
5.5.3 敏捷建模	58	扩展阅读与信息资源	92
5.5.4 敏捷统一过程	59	第8章 理解需求	94
5.6 敏捷过程工具集	60	8.1 需求工程	95
5.7 小结	61	8.2 建立根基	100
习题与思考题	61	8.2.1 确认利益相关者	100
扩展阅读与信息资源	62	8.2.2 识别多重观点	100
第6章 软件工程的人员方面	64	8.2.3 协同合作	101
6.1 软件工程师的特质	64	8.2.4 首次提问	101
6.2 软件工程心理学	65	8.2.5 非功能需求	102
6.3 软件团队	66	8.2.6 可追溯性	102
6.4 团队结构	67	8.3 获取需求	103
6.5 敏捷团队	68	8.3.1 协作收集需求	103
6.5.1 通用敏捷团队	68	8.3.2 质量功能部署	105
6.5.2 XP 团队	69	8.3.3 使用场景	106
6.6 社交媒体的影响	70	8.3.4 获取工作产品	106
6.7 软件工程中云的应用	71	8.3.5 敏捷需求获取	107
6.8 协作工具	71	8.3.6 面向服务的方法	107
6.9 全球化团队	72	8.4 开发用例	107
6.10 小结	73	8.5 构建分析模型	111
习题与思考题	73	8.5.1 分析模型的元素	112
扩展阅读与信息资源	74	8.5.2 分析模式	114
第二部分 建模		8.5.3 敏捷需求工程	114
第7章 指导实践的原则	76	8.5.4 自适应系统的需求	114
7.1 软件工程知识	77	8.6 协商需求	115
7.2 核心原则	77	8.7 需求监控	116
7.2.1 指导过程的原则	78	8.8 确认需求	117
7.2.2 指导实践的原则	78		
7.3 指导每个框架活动的原则	80		
7.3.1 沟通原则	80		
7.3.2 策划原则	81		
7.3.3 建模原则	83		
7.3.4 构建原则	87		
7.3.5 部署原则	89		
7.4 工作实践	90		
7.5 小结	91		
习题与思考题	92		
扩展阅读与信息资源	92		

8.9 避免常见错误	117	11.4 需求建模的模式	152
8.10 小结	118	11.4.1 发现分析模式	152
习题与思考题	118	11.4.2 需求模式举例：执行器 - 传感器	153
扩展阅读与信息资源	119	11.5 Web / 移动 App 的需求建模	156
第 9 章 需求建模：基于场景的 方法	121	11.5.1 多少分析才够用	156
9.1 需求分析	122	11.5.2 需求建模的输入	157
9.1.1 总体目标和原理	122	11.5.3 需求建模的输出	157
9.1.2 分析的经验原则	123	11.5.4 内容模型	158
9.1.3 域分析	123	11.5.5 WebApp 和移动 App 的 交互模型	159
9.1.4 需求建模的方法	125	11.5.6 功能模型	159
9.2 基于场景建模	126	11.5.7 WebApp 的配置模型	160
9.2.1 创建初始用例	126	11.5.8 导航建模	161
9.2.2 细化初始用例	128	11.6 小结	161
9.2.3 编写正式用例	129	习题与思考题	162
9.3 补充用例的 UML 模型	131	扩展阅读与信息资源	162
9.3.1 开发活动图	131	第 12 章 设计概念	163
9.3.2 泳道图	132	12.1 软件工程中的设计	164
9.4 小结	133	12.2 设计过程	166
习题与思考题	133	12.2.1 软件质量指导原则和 属性	166
扩展阅读与信息资源	133	12.2.2 软件设计的演化	168
第 10 章 需求建模：基于类的方法	135	12.3 设计概念	169
10.1 识别分析类	135	12.3.1 抽象	169
10.2 描述属性	138	12.3.2 体系结构	169
10.3 定义操作	138	12.3.3 模式	170
10.4 类 - 职责 - 协作者建模	140	12.3.4 关注点分离	170
10.5 关联和依赖	145	12.3.5 模块化	171
10.6 分析包	145	12.3.6 信息隐蔽	171
10.7 小结	146	12.3.7 功能独立	172
习题与思考题	146	12.3.8 求精	172
扩展阅读与信息资源	146	12.3.9 方面	173
第 11 章 需求建模：行为、模式和 Web / 移动 App	148	12.3.10 重构	173
11.1 生成行为模型	148	12.3.11 面向对象的设计概念	174
11.2 识别用例事件	149	12.3.12 设计类	174
11.3 状态表达	149	12.3.13 依赖倒置	176
		12.3.14 测试设计	177

12.4 设计模型	177	13.11 敏捷性与体系结构	203
12.4.1 数据设计元素	178	13.12 小结	204
12.4.2 体系结构设计元素	178	习题与思考题	205
12.4.3 接口设计元素	179	扩展阅读与信息资源	205
12.4.4 构件级设计元素	180		
12.4.5 部署级设计元素	181	第 14 章 构件级设计	207
12.5 小结	181	14.1 什么是构件	208
习题与思考题	182	14.1.1 面向对象的观点	208
扩展阅读与信息资源	183	14.1.2 传统的观点	209
		14.1.3 过程相关的观点	211
第 13 章 体系结构设计	184	14.2 设计基于类的构件	212
13.1 软件体系结构	185	14.2.1 基本设计原则	212
13.1.1 什么是体系结构	185	14.2.2 构件级设计指导方针	214
13.1.2 体系结构为什么重要	186	14.2.3 内聚性	215
13.1.3 体系结构描述	186	14.2.4 耦合性	216
13.1.4 体系结构决策	187	14.3 实施构件级设计	217
13.2 体系结构类型	188	14.4 WebApp 的构件级设计	222
13.3 体系结构风格	188	14.4.1 构件级内容设计	222
13.3.1 体系结构风格的简单		14.4.2 构件级功能设计	222
分类	189	14.5 移动 App 的构件级设计	222
13.3.2 体系结构模式	191	14.6 设计传统构件	223
13.3.3 组织和求精	192	14.7 基于构件的开发	223
13.4 体系结构考虑要素	192	14.7.1 领域工程	223
13.5 体系结构决策	194	14.7.2 构件的合格性检验、适应	
13.6 体系结构设计	194	性修改与组合	224
13.6.1 系统环境的表示	195	14.7.3 体系结构不匹配	225
13.6.2 定义原型	195	14.7.4 复用的分析与设计	226
13.6.3 将体系结构细化为构件	196	14.7.5 构件的分类与检索	226
13.6.4 描述系统实例	197	14.8 小结	227
13.6.5 WebApp 的体系结构		习题与思考题	228
设计	198	扩展阅读与信息资源	228
13.6.6 移动 App 的体系结构			
设计	198	第 15 章 用户界面设计	230
13.7 评估候选的体系结构设计	199	15.1 黄金规则	231
13.7.1 体系结构描述语言	200	15.1.1 把控制权交给用户	231
13.7.2 体系结构评审	201	15.1.2 减轻用户的记忆负担	232
13.8 经验学习	201	15.1.3 保持界面一致	233
13.9 基于模式的体系结构评审	202	15.2 用户界面的分析和设计	234
13.10 体系结构一致性检查	203	15.2.1 用户界面分析和设计	
		模型	234

15.2.2 过程	235	16.6.2 设计粒度	266
15.3 界面分析	236	16.7 移动 App 模式	267
15.3.1 用户分析	236	16.8 小结	268
15.3.2 任务分析和建模	237	习题与思考题	268
15.3.3 显示内容分析	240	扩展阅读与信息资源	269
15.3.4 工作环境分析	240		
15.4 界面设计步骤	241	第 17 章 WebApp 设计	271
15.4.1 应用界面设计步骤	241	17.1 WebApp 设计质量	272
15.4.2 用户界面设计模式	243	17.2 设计目标	273
15.4.3 设计问题	243	17.3 WebApp 设计金字塔	274
15.5 WebApp 和移动 App 的界面设计	245	17.4 WebApp 界面设计	274
15.5.1 界面设计原则与指导方针	245	17.5 美学设计	275
15.5.2 WebApp 和移动 App 的界面设计 workflow	248	17.5.1 布局问题	276
15.6 设计评估	249	17.5.2 平面设计问题	276
15.7 小结	250	17.6 内容设计	277
习题与思考题	250	17.6.1 内容对象	277
扩展阅读与信息资源	251	17.6.2 内容设计问题	278
		17.7 体系结构设计	278
第 16 章 基于模式的设计	253	17.7.1 内容体系结构	279
16.1 设计模式	254	17.7.2 WebApp 体系结构	280
16.1.1 模式的种类	255	17.8 导航设计	281
16.1.2 框架	256	17.8.1 导航语义	281
16.1.3 描述模式	257	17.8.2 导航语法	283
16.1.4 模式语言和模式库	258	17.9 构件级设计	283
16.2 基于模式的软件设计	258	17.10 小结	283
16.2.1 不同环境下基于模式的设计	258	习题与思考题	284
16.2.2 用模式思考	259	扩展阅读与信息资源	284
16.2.3 设计任务	260		
16.2.4 建立模式组织表	260	第 18 章 移动 App 设计	286
16.2.5 常见设计错误	261	18.1 挑战	287
16.3 体系结构模式	262	18.1.1 开发因素	287
16.4 构件级设计模式	263	18.1.2 技术因素	288
16.5 用户界面设计模式	264	18.2 开发移动 App	289
16.6 WebApp 设计模式	266	18.2.1 移动 App 质量	290
16.6.1 设计焦点	266	18.2.2 用户界面设计	291
		18.2.3 环境感知 App	292
		18.2.4 经验教训	293
		18.3 移动 App 设计的最佳实践	294

18.4 移动开发环境	295	20.4 评审的正式程度	321
18.5 云	297	20.5 非正式评审	322
18.6 传统软件工程的适用性	298	20.6 正式技术评审	323
18.7 小结	298	20.6.1 评审会议	324
习题与思考题	299	20.6.2 评审报告和记录保存	324
扩展阅读与信息资源	299	20.6.3 评审指导原则	325
		20.6.4 样本驱动评审	326
		20.7 产品完成后评估	327
		20.8 小结	327
		习题与思考题	327
		扩展阅读与信息资源	328
第三部分 质量管理			
第 19 章 质量概念	302	第 21 章 软件质量保证	329
19.1 什么是质量	303	21.1 背景问题	330
19.2 软件质量	304	21.2 软件质量保证的要素	330
19.2.1 Garvin 的质量维度	304	21.3 软件质量保证的过程和产品	
19.2.2 McCall 的质量因素	305	特性	332
19.2.3 ISO 9126 质量因素	306	21.4 软件质量保证的任务、目标和	
19.2.4 定向质量因素	306	度量	332
19.2.5 过渡到量化观点	307	21.4.1 软件质量保证的任务	332
19.3 软件质量困境	308	21.4.2 目标、属性和度量	333
19.3.1 “足够好”的软件	308	21.5 软件质量保证的形式化方法	334
19.3.2 质量的成本	309	21.6 统计软件质量保证	335
19.3.3 风险	311	21.6.1 一个普通的例子	335
19.3.4 疏忽和责任	311	21.6.2 软件工程中的六西格玛	336
19.3.5 质量和安全	312	21.7 软件可靠性	337
19.3.6 管理活动的影响	312	21.7.1 可靠性和可用性的测量	337
19.4 实现软件质量	313	21.7.2 软件安全	338
19.4.1 软件工程方法	313	21.8 ISO 9000 质量标准	339
19.4.2 项目管理技术	313	21.9 软件质量保证计划	340
19.4.3 质量控制	313	21.10 小结	341
19.4.4 质量保证	313	习题与思考题	341
19.5 小结	314	扩展阅读与信息资源	341
习题与思考题	314		
扩展阅读与信息资源	314		
第 20 章 评审技术	316	第 22 章 软件测试策略	343
20.1 软件缺陷对成本的影响	317	22.1 软件测试的策略性方法	344
20.2 缺陷的放大和消除	318	22.1.1 验证与确认	344
20.3 评审度量及其应用	319	22.1.2 软件测试组织	345
20.3.1 分析度量数据	320	22.1.3 软件测试策略——宏观	346
20.3.2 评审的成本效益	320		

22.1.4 测试完成的标准	347	23.4.4 图矩阵	374
22.2 策略问题	348	23.5 控制结构测试	375
22.3 传统软件的测试策略	348	23.6 黑盒测试	376
22.3.1 单元测试	348	23.6.1 基于图的测试方法	376
22.3.2 集成测试	350	23.6.2 等价类划分	378
22.4 面向对象软件的测试策略	354	23.6.3 边界值分析	378
22.4.1 面向对象环境中的单元 测试	354	23.6.4 正交数组测试	379
22.4.2 面向对象环境中的集成 测试	354	23.7 基于模型的测试	381
22.5 WebApp 的测试策略	355	23.8 文档测试和帮助设施测试	381
22.6 移动 App 的测试策略	355	23.9 实时系统的测试	382
22.7 确认测试	356	23.10 软件测试模式	383
22.7.1 确认测试准则	356	23.11 小结	384
22.7.2 配置评审	356	习题与思考题	384
22.7.3 α 测试和 β 测试	356	扩展阅读与信息资源	385
22.8 系统测试	358	第 24 章 测试面向对象的应用	386
22.8.1 恢复测试	358	24.1 扩展测试的视野	387
22.8.2 安全测试	358	24.2 测试 OOA 和 OOD 模型	387
22.8.3 压力测试	358	24.2.1 OOA 和 OOD 模型的 正确性	388
22.8.4 性能测试	359	24.2.2 面向对象模型的一致性	388
22.8.5 部署测试	359	24.3 面向对象测试策略	389
22.9 调试技巧	360	24.3.1 面向对象环境中的单元 测试	389
22.9.1 调试过程	360	24.3.2 面向对象环境中的集成 测试	390
22.9.2 心理因素	361	24.3.3 面向对象环境中的确认 测试	390
22.9.3 调试策略	362	24.4 面向对象测试方法	390
22.9.4 纠正错误	363	24.4.1 面向对象概念的测试用例 设计含义	391
22.10 小结	363	24.4.2 传统测试用例设计方法的 可应用性	391
习题与思考题	364	24.4.3 基于故障的测试	391
扩展阅读与信息资源	364	24.4.4 基于场景的测试设计	392
第 23 章 测试传统的应用软件	366	24.5 类级可应用的测试方法	392
23.1 软件测试基础	367	24.5.1 面向对象类的随机测试	392
23.2 测试的内部视角和外部视角	368	24.5.2 类级的划分测试	393
23.3 白盒测试	369	24.6 类间测试用例设计	394
23.4 基本路径测试	369		
23.4.1 流图表示	369		
23.4.2 独立程序路径	371		
23.4.3 生成测试用例	372		