

教育部高等教育司推荐
国外优秀信息科学与技术系列教学用书

软件工程

——理论与实践

(第四版 影印版)

SOFTWARE ENGINEERING
Theory and Practice

(Fourth Edition)

■ Shari Lawrence Pfleeger
Joanne M. Atlee

PEARSON
Prentice
Hall



高等教育出版社
Higher Education Press

教育部高等教育司推荐
国外优秀信息科学与技术系列教学用书

软 件 工 程

——理论与实践

(第四版 影印版)



高等教育出版社

Higher Education Press

图字：01-2009-3270号

Software Engineering: Theory and Practice, Fourth Edition

Shari Lawrence Pfleeger, Joanne M. Atlee

本书封面贴有 Pearson Education(培生教育出版集团)激光防伪标签。无标签者不得销售。

Original edition, entitled SOFTWARE ENGINEERING, 4E, 9780138141813 by PFLEEGER, SHARI LAWRENCE; ATLEE, JOANNE M., published by Pearson Education, Inc, publishing as Prentice Hall, Copyright © 2010 Pearson Higher Education. Upper Saddle River, New Jersey 07458.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and HIGHER EDUCATION PRESS Copyright © 2010.

本书原版为培生教育出版集团出版 SOFTWARE ENGINEERING, 4E, 作者为 PFLEEGER, SHARI LAWRENCE; ATLEE, JOANNE M.。著作权© 2010。

版权所有。未经培生教育出版集团(Pearson Education, Inc.)的许可,任何人不得通过任何电子或机械的手段(包括影印、录音)或者通过任何信息存储检索系统翻印或传播本书的任何部分。此中文版由培生教育出版集团和高等教育出版社合作出版。著作权© 2010。

原版 ISBN: 0-13-814181-3

For sale and distribution in the People's Republic of China exclusively (except Taiwan, Hong Kong SAR and Macau SAR).

仅限于中华人民共和国境内(但不允许在中国香港、澳门特别行政区和中国台湾地区)销售发行。

图书在版编目(CIP)数据

软件工程:理论与实践 = Software Engineering: Theory and Practice:

第4版:英文/(美)弗莱格(Pfleeger, S.L.), (美)阿特利(Atlee, J.M.). 影印本. —北京:高等教育出版社, 2009. 12

ISBN 978-7-04-027947-4

I. 软… II. ①弗…②阿… III. 软件工程—英文 IV. TP311.5

中国版本图书馆CIP数据核字(2009)第183461号

出版发行	高等教育出版社	购书热线	010-58581118
社 址	北京市西城区德外大街4号	咨询电话	400-810-0598
邮政编码	100120	网 址	http://www.hep.edu.cn
总 机	010-58581000		http://www.hep.com.cn
经 销	蓝色畅想图书发行有限公司	网上订购	http://www.landaco.com
印 刷	北京民族印务有限责任公司		http://www.landaco.com.cn
开 本	787×1092 1/16	畅想教育	http://www.widedu.com
印 张	49.75	版 次	2009年12月第1版
字 数	700 000	印 次	2009年12月第1次印刷
		定 价	53.00元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 27947-00

序

20 世纪末，以计算机和通信技术为代表的信息科学和技术对世界经济、科技、军事、教育和文化等产生了深刻影响。信息科学技术的迅速普及和应用，带动了世界范围信息产业的蓬勃发展，为许多国家带来了丰厚的回报。

进入 21 世纪，尤其随着我国加入 WTO，信息产业的国际竞争将更加激烈。我国信息产业虽然在 20 世纪末取得了迅猛发展，但与发达国家相比，甚至与印度、爱尔兰等国家相比，还有很大差距。国家信息化的发展速度和信息产业的国际竞争能力，最终都将取决于信息科学技术人才的质量和数量。引进国外信息科学和技术优秀教材，在有条件的学校推动开展英语授课或双语教学，是教育部为加快培养大批高质量的信息技术人才采取的一项重要举措。

为此，教育部要求由高等教育出版社首先开展信息科学和技术教材的引进试点工作。同时提出了两点要求，一是要高水平，二是要低价格。在高等教育出版社和信息科学技术引进教材专家组的努力下，经过比较短的时间，第一批引进的 20 多种教材已经陆续出版。这套教材出版后受到了广泛的好评，其中有不少是世界信息科学技术领域著名专家、教授的经典之作和反映信息科学技术最新进展的优秀作品，代表了目前世界信息科学技术教育的一流水平，而且价格也是最优惠的，与国内同类自编教材相当。

这项教材引进工作是在教育部高等教育司和高教社的共同组织下，由国内信息科学技术领域的专家、教授广泛参与，在对大量国外教材进行多次遴选的基础上，参考了国内和国外著名大学相关专业的课程设置进行系统引进的。其中，John Wiley 公司出版的贝尔实验室信息科学研究中心副总裁 Silberschatz 教授的经典著作《操作系统概念》，是我们经过反复谈判，做了很多努力才得以引进的。William Stallings 先生曾编写了在美国深受欢迎的信息科学技术系列教材，其中有多种教材获得过美国教材和学术作者协会颁发的计算机科学与工程教材奖，这批引进教材中就有他的两本著作。留美中国学者 Jiawei Han 先生的《数据挖掘》是该领域中具有里程碑意义的著作。由达特茅斯学院 Thomas Cormen 和麻省理工学院、哥伦比亚大学的几位学者共同编着的经典著作《算法导论》，在经历了 11 年的锤炼之后于 2001 年出版了第二版。目前任教于美国 Massachusetts 大学的 James Kurose 教授，曾在美国三所高校先后 10 次获得杰出教师或杰出教学奖，由他主编的《计算机网络》出版后，以其体系新颖、内容先进而倍受欢迎。在努力降低

引进教材售价方面，高等教育出版社做了大量和细致的工作。这套引进的教材体现了权威性、系统性、先进性和经济性等特点。

教育部也希望国内和国外的出版商积极参与此项工作，共同促进中国信息技术教育和信息产业的发展。我们在与外商的谈判工作中，不仅要坚定不移地引进国外最优秀的教材，而且还要千方百计地将版权转让费降下来，要让引进教材的价格与国内自编教材相当，让广大教师和学生负担得起。中国的教育市场巨大，外国出版公司和国内出版社要通过扩大发行数量取得效益。

在引进教材的同时，我们还应做好消化吸收，注意学习国外先进的教学思想和教学方法，提高自编教材的水平，使我们的教学和教材在内容体系上，在理论与实践的结合上，在培养学生的动手能力上能有较大的突破和创新。

目前，教育部正在全国 35 所高校推动示范性软件学院的建设和实施，这也是加快培养信息科学技术人才的重要举措之一。示范性软件学院要立足于培养具有国际竞争力的实用性软件人才，与国外知名高校或著名企业合作办学，以国内外著名 IT 企业为实践教学基地，聘请国内外知名教授和软件专家授课，还要率先使用引进教材开展教学。

我们希望通过这些举措，能在较短的时间，为我国培养一大批高质量的信息技术人才，提高我国软件人才的国际竞争力，促进我国信息产业的快速发展，加快推动国家信息化进程，进而带动整个国民经济的跨越式发展。

教育部高等教育司

二〇〇二年三月

前 言

在理论研究和实践应用之间架起桥梁

自1968年“软件工程”这个术语在一次北大西洋公约组织(North Atlantic Treaty Organization, NATO)会议上首次使用以来,软件工程已经走过了很长的路。软件本身也以某种方式进入了我们的生活,而在十年前很少有人能预见到这一点。因此,扎实的软件工程理论和实践基础对理解如何开发出优秀的软件和评价软件在生活中的风险和机遇是非常有必要的。本书体现了当前软件工程两个方面的结合:实践者的主要目标是构建能实现一定功能的优质产品,而研究者则努力找到改进产品质量以及提高产品生产率的方法。Edsger Dykstra常常提醒我们,在理论和实践中的困难能检验我们对软件工程的理解,并有助于改进我们的思想、方法以及最终的产品。

为问题和改进构建一个基本框架,这是我们在本书中所强调的精神。特别地,本书的第四版包含了更广泛的内容,包括如何对问题进行抽象和建模,以及如何应用模型设计原则、设计模式和设计策略来创建适当的解决方案。软件工程师要比仅仅按照指编程的程序员承担更多的工作,正如厨师长要比只按菜单配方做菜的厨师做得更多。构建一个优秀的软件是一门艺术,它体现在如何对问题的本质进行抽象和建模并用这些抽象来设计解决方案的理解上。我们经常听到优秀的开发人员讨论“优雅的(elegant)”解决方法,这意味着这种解决方法反映了问题的核心,软件不仅解决了它当前形式下的问题,而且可以根据问题的发展变化而进行修改。这样,学生学习如何将理论和实践相结合,将艺术和科学相结合,从而构建坚实的软件。

科学总是以事实为基础的。为了设计本科生的软件工程课程,本书绘制了一个软件工程理论和实践的实用图,以便学生能将他们的所学直接应用到他们将要努力解决的真实问题中去。示例说明了学生经验有限,但清楚地表明了大型开发项目从需求到理想再到现实的进程。这些例子体现了读者可能遇到的很多情况:大的或小的项目;“灵活的”或高度结构化的方法;面向对象或面向过程的方法;实时方法或事务处理方法,开发和维护等。

本书也适用于介绍软件工程概念和实践入门的研究生课程,或那些希望拓展

本学科知识的专业人士。特别需要指出的是，第 12、13 和 14 章给出了引发读者思考的材料，它们专为对当前研究课题感兴趣的研究生而设计。

主要特点

本书具有许多区别于其他书的主要特征：

- 与其他书将度量和建模作为单独的问题来考虑不同，本书将用软件工程的一般讨论方法综合考虑度量与建模问题。即将度量和建模作为软件工程中不可分割的部分来考虑，而不是作为一个单独的问题。因此，学生将学习如何抽象和建模，以及如何对日常活动进行评估和改进。他们可以利用模型来理解他们所要解决的问题或解决方案的重要部分，还可以利用度量来评价以个人、小组或项目要素的进展。

- 同样地，将复用、风险管理和质量工程等概念都嵌入到软件工程活动中，它们被看成是受软件工程影响的，而不是将它们作为独立的问题来对待。

- 本版本提出了敏捷的方法，包括极限程序设计(extreme programming)。介绍了能给予开发者更多自治权的益处和风险，并将这种敏捷方法与更多的传统软件开发方法相比较。

- 每一章都将所介绍的概念应用到两个通用实例中：一个例子代表典型的信息系统，一个例子代表实时系统。这两个例子都基于实际的项目。信息系统实例介绍了为一家大型英国电视公司确定广告时间与价格的软件，实时系统的实例则是 Ariane-5 火箭的控制软件。我们考虑提出的问题，探讨软件工程是如何帮助确定并避免这些问题的出现。学生可以跟随这两个项目的进展，理解本书介绍的不同实践方法是如何与构建系统的技术相结合的。

- 每章的最后用三种方式表达结果：本章内容对开发团队的意义、对开发人员个人的意义以及对研究人员的意义。学生可以很容易地回顾每章的要点，并了解本章内容与研究和实践的相关性。

- 本书配套 Web 网站为 <http://www.prenhall.com/pfleeger>。它包括了书中的例子和实际项目的实例，还包括了与相关工具及方法的销售方的 Web 页面的链接。在此，学生可以找到实际的需求文档、设计、代码、测试计划以及更多内容，并为那些想查找具有一定深度信息的学生给出相关出版物和 Web 网站信息。这些 Web 网页会定期更新，以保证书中的材料是最新的，通过网页还可以向作者和出版商提出意见或建议。

- 通过当地的 Pearson 代理商可以获得学生学习指导(Student Study Guide)。

- 教师资源中心(Instructor Resource Center)提供 PPT 幻灯片和完整的解决方

案手册。请与当地 Pearson 代理商获取相关信息。

- 本书采用了大量研究实例和文献中的例子。许多附注中的单页实例都在 Web 网页中进行了扩充。学生可以从中学学习到如何将书中的理论概念应用到真实生活中。

- 本书的每一章都以引人深思的有关软件工程的法律和道德问题作为该章的结束。学生可以根据自己的社会和政治背景来理解软件工程。与其他科学一样，软件工程的决策必须根据它们给人们带来的影响做出评论。

- 每章都讨论了过程式和面向对象的开发。此外，第 6 章解释了面向对象开发过程的步骤。我们讨论了一些设计原则，并使用面向对象实例来展示如何通过采用这些原则来改进设计方法。

- 本书还包括一个带有注释的参考目录，它列出了许多有关软件工程的创造性论文。另外，Web 网页中还包括了带注释的参考目录和专业讨论组，如软件的可靠性、容错性、计算机安全，等等。

- 每一章都包括一个学期课题的描述，它是一个借贷处理系统软件开发任务，教师可以将该项目或稍作变化后作为课外作业布置给学生。

- 每章的结尾都包括一个本章参考文献列表，它可以帮助学生找到本章所讨论的工具或方法的深度信息。

- 第四版包括了一些突出计算机安全性的实例。特别地，我们强调在设计阶段就考虑安全性，而不是在编码和测试期间再加入安全性。

内容及组织

本书分为三部分。第一部分意在激励读者，并向从业人员和研究人员解释了软件工程知识的重要性。还讨论了理解过程的必要性，以决定开发人员的“灵活度”和完成项目计划。第二部分讨论了开发和维护的主要步骤，如需求的引出、建模和检查，设计问题的解决方案，代码的编写和测试，以及软件的交付，但未考虑用来构建软件的过程模型。第三部分讨论了软件的评估和改进，分析如何评价过程和产品的质量，并如何加以改进。

第 1 章 为什么需要软件工程？

本章描绘出全书的脉络，并激发读者的学习兴趣，提示将在后面章节中哪些地方解决有关的关键问题。特别地，介绍了 Wasserman 的帮助定义软件工程的关键要素：抽象、方法和符号的设计与分析、模块化和结构、软件生命周期和过

程、复用、度量、工具和集成环境、用户接口和原型。我们讨论了计算机科学和软件工程的区别，阐明了可能遇到的主要问题类型，为本书后续章节奠定基础。本章还探讨了采用一种系统方法来构建软件的必要性，介绍了每章中将要使用到的两个常用实例以及学期课题的内容。

第2章 过程和生命周期的建模

本章总结了各种过程和生命周期模型的类型，包括瀑布模型、V模型、螺旋模型和各种原型化模型。说明了敏捷方法的必要性，与传统的软件开发方法相比，敏捷方法的开发人员被赋予更大的自主权。我们还介绍了几种建模技术和工具，包括系统动态模型以及几种其他常用的方法，并采用其中几种技术对两个通用实例的相关部分均进行了建模。

第3章 项目计划和管理

本章研究项目的计划和进度管理。介绍了活动、里程碑、工作分解结构、活动图、风险管理、成本及成本评估等概念。采用了评估模型来评估本书两个通用实例的成本和进度。本章主要研究实例，包括F-16战斗机和Digital's alpha AXP程序的软件开发管理。

第4章 需求获取

本章强调了在良好的软件工程中抽象和建模的重要角色。特别地，我们使用模型来从已有的需求中梳理出误解和遗漏的细节，并与他人沟通以了解需求。我们研究了多种不同的建模范例，为每种范例研究样本符号，讨论何时采用何种范例，并为如何做出特定的建模和抽象的决定提供建议。我们还讨论了需求的不同资源和不同类型(功能需求、质量需求、设计约束)，解释如何编写可测试的需求，并描述如何解决冲突。另外，本章还讨论了需求引出、需求文档、需求评审、需求质量及度量，以及如何选择一个规格说明方法的实例。最后还将一些方法应用到本书的两个通用实例中去。

第5章 架构设计

第四版中对介绍软件架构的章节内容进行了完整的修订。首先，本章描述了

架构在软件设计过程以及在大型开发过程中的角色，然后讨论了架构设计的步骤，包括建模、分析、文档化以及评审，最后产生供程序设计人员使用的描述模块和接口的软件架构文档(Software Architecture Document)。本章还讨论了如何分解问题，如何从不同的视角来考察问题的多个方面，从而确定最合适的解决方案。接下来，本章重点介绍了使用一种或多种架构模式，包括管道-过滤器(pipe-and-filter)、对等(peer-to-peer)、客户-服务器(client-server)、发布订阅(publish-subscribe)、仓库(repository)和层次化(layering)来对解决方案进行建模。同时还探讨了组合不同的模式并使用它们实现质量目标，例如可修改性、性能、安全性、可靠性、健壮性以及可用性。

一旦获得了最初的架构，便可以对其进行评估和完善，在本章中，我们说明了如何度量设计质量并在安全性分析、折中分析、成本效益分析中使用评估技术来选择能够满足客户需求的最佳架构。其中强调了对设计过程进行文档记录的重要性，这有助于核实和验证设计是否与需求相符，并创建满足客户需求的架构。在本章结尾，我们阐述了如何建立一种产品线架构，使软件提供方可以在系列产品中对设计进行重用。最后，本章以一个信息系统实例和一个实时系统实例的架构分析作为终结。

第6章 模块设计

本版本对第6章的修改较多，探讨了如何从系统架构描述转化到各个子模块设计的描述。首先我们讨论了设计过程，然后介绍了利用6个重要的设计原则将架构精加工为模块：模块性、接口、信息隐藏、增量开发、抽象性、通用性。接下来，我们深入讨论了面向对象设计以及它如何支持这6个原则。通过使用统一建模语言(Unified Modeling Language)的不同表示法，我们说明了如何表示模块功能和交互的多个方面，从而获得一个健壮而且可维护的设计。此外，本章还描述了一组设计模式(每种设计模式都具有特定的用途)，并举例说明了如何使用它们以强化设计原则。接下来，我们讨论了诸如数据管理、异常处理、用户接口以及框架等全局性问题，从本章，读者可以看到方法的一致性和清晰有利于获得更有效的设计。

本章仔细探讨了面向对象的评测方法，并且将一些常用的面向对象评测尺度应用到一个服务站实例，并注明了由于设计的修改而导致评测尺度值的变化是如何帮助我们确定资源分配并定位错误的。最后介绍了如何将面向对象的概念应用于信息系统实例和实时系统实例中。

第7章 程序编写

本章介绍了代码级设计决策问题，以及有关高质量代码设计的实现问题。讨论了标准和过程，给出了一些简单的编程指南。提供了用几种语言编写的示例，包括面向对象语言和面向过程语言。我们讨论了程序文档和错误处理策略的必要性，最后，将其中一些概念应用于本书的两个通用实例中。

第8章 程序测试

本章探讨了程序测试的几个方面，区分了传统测试方法与“净室”（cleanroom）测试方法的不同，并分析了如何测试不同的系统。我们提出了软件问题的定义和分类，并讨论如何使用正交错误分类使得数据收集和分析更有效。之后解释了单元测试和集成测试的区别。在介绍几种自动测试工具和技术之后，解释了测试生命周期的重要性，以及如何将工具集成到其中。最后，本章将这些概念应用于书中的两个通用实例中。

第9章 系统测试

我们首先介绍了系统测试的原则，包括测试集和数据的复用，以及认真进行配置管理的必要性。介绍的概念包括功能测试、性能测试、验收测试和安装测试。分析了测试面向对象系统的特殊需要，介绍了几个测试工具，并讨论了小组成员的任务。接着，我们向读者介绍了软件可靠性建模，并讨论了软件的易用性、可维护性和可用性。读者从中可以学习如何使用测试结果评估交付产品的性能，同时还介绍了几种测试文档类型，并以介绍本书两个通用实例的测试策略作为本章的结束。

第10章 系统交付

本章讨论了培训和文档的必要性，提供了几个信息系统和实时系统例子中的培训和文档的示例。

第 11 章 系统维护

本章分析了系统改变的后果。解释了在系统的生命周期中改变是如何发生的，以及系统设计、代码、测试过程和文档如何适应变化。我们讨论了典型的维护问题，再次强调了认真进行配置管理的必要性。本章全面地讨论了采用度量来预测可能发生的变化，并评估变化带来的影响。我们考虑了对遗留系统进行再生工程和重构造。最后，根据可能的变化对书中两个通用实例进行了评估。

第 12 章 产品、过程和资源的评价

由于许多软件工程的决策涉及对现有组件的合并和集成，本章给出了评价过程和产品的办法。我们讨论了经验评估的必要性，并给出几个例子以体现如何用测量标准来建立质量和生产率的基础。我们分析了几个质量模型，介绍了如何评估系统的复用性、如何完成算后检查(post-mortems)以及如何理解信息技术投资的回报。所有这些概念都被应用于书中的两个通用实例中。

第 13 章 预测、产品、过程和资源的改进

本章建立在第 11 章的基础之上，展示了如何实现预测、产品、过程和资源的改进。它包括几个深入的实例研究，以说明如何理解预测模型、检测技术以及软件工程其他方面的问题，并使用多种研究性技术来加以改进。本章的结尾给出了一组对评估当前状态和确定改进机会的指导方针。

第 14 章 软件工程的前景

本书的最后一章研究了几个有关软件工程的开放性问题。我们回顾了 Wasserman 的概念，将它作为一个原则来了解我们做得如何。我们还分析技术转移和制定决策的几个问题，以决定我们是否做好了从理论到实践的重要转变工作。最后，我们新增了对几个有争议问题的分析，如软件工程作为专业工程的许可，以及更多特殊领域解决方案和方法的趋势。

致谢

本书的完成要归功于朋友和家人提供的技术和情感的支持。在此，我们不能列出所有在本书的编写和校对过程中曾经支持过我们的人，所以我们首先要为可能的疏忽而致歉。非常感谢本书以前版本的读者，他们的仔细阅读对本书的修订提供了极好的建议。据我们所知，所有这些建议都收编到这个版本中，我们将继续重视读者的反馈，不管是正面的还是反面的。

Carolyn Seaman (Maryland - Baltimore Campus 大学) 是本书第一版的很棒的评论家，提出了既清晰又简单的方法，以使本书更为紧凑，更易于理解。同时，她还为本书的大多数习题提供了解答，并帮助建立了本书初期的 Web 网站。我非常感激她的友谊和帮助。Yiqing Liang 和 Carla Valle 更新了这个网站，并为本书的第二版补充了新的资料，Patsy Ann Zimmer (Waterloo 大学) 为第三版修改了该网站，最明显的是增加了建模符号和敏捷方法的内容。

我们非常感谢 Forrest Shull (Maryland 大学 Fraunhofer 中心) 和 Roseanne Tesoriero (Washington 学院)，他们规划了本书最早的学习指南，还要感谢 Maria Vieira Nelson (Minas Gerais Catholic 大学, 巴西)，他修订了第三版的学习指南和解答手册。感谢 Eduardo S. Barrenechea (Waterloo 大学)，他更新了第四版的相关材料。还要感谢 Hossein Saiedian (Kansas 大学)，他提供了第三版和第四版的 PPT。我们同样十分感激 Guilherme Travassos (Rio de Janeiro 的 Federal 大学)，我们采用了他与 Maryland - College Park 大学的 Pfleeger 开发所用的材料，后者丰富并扩展了子类的使用。

所有 4 个版本中那些提供了很多帮助和很有见地意见的评论家包括：Barbara Kitchenham (Keele 大学, 英国)、Bernard Woolfolk (Lucent 科技)、Ana Regina Cavalcanti da Rocha (Rio de Janeiro 的 Federal 大学)、Frances Uku (Berkeley 的 California 大学)、Lee Scott Ehrhart (MITRE)、Laurie Werth (Texas 大学)、Vickie Almstrum (Texas 大学)、Lionel Briand (Simula Research, 挪威)、Steve Thibaut (Florida 大学)、Lee Wittenberg (New Jersey 的 Kean 学院)、Philip Johnson (Hawaii 大学)、Daniel Berry (Waterloo 大学, 加拿大)、Nancy Day (Waterloo 大学)、Jianwei Niu (Waterloo 大学)、Chris Gorringer (East Anglia 大学, 英国)、Ivan Aaen (Aalborg 大学)、Damla Turget (Central Florida 大学)、Laurie Williams (North Carolina State 大学)、Ernest Sibert (Syracuse 大学)、Allen Holliday (California State 大学, Fullerton)、David Rine (George Mason 大学)、Anthony Sullivan (Texas 大学, Dallas)、David Chesney (Michigan 大学, Ann Arbor)、Ye Duan (Missouri 大学)、Rammohan

K. Ragade (Kentucky 大学), 以及一些由 Prentice Hall 提供的匿名评论家。与 Greg Hislop (Drexel 大学)、John Favaro (Intecs Sistemi, 意大利)、Filippo Lanubile (Università di Bari, 意大利)、John d'Ambra (New South Wales 大学, 澳大利亚)、Chuck Howell (MITRE)、Tim Vieregge (美国 Army Computer Emergency Response Team)、James 和 Suzanne Robertson (Atlantic System Guild, 英国) 的讨论使本书得到许多改进和增强。

感谢 Toni Holm 和 Alan Apt, 是他们使第三版本变得有趣而轻松, 也感谢 James 和 Suzanne Robertson, 他们提供了 Piccadilly 的例子; 还有 Norman Fenton, 他为本书提供了软件度量材料。我们很感谢 Tracy Dunkelberger 在我们编写第四版过程中给予的鼓励, 她的耐心和她的专业精神都让我们心存感激, 此外还要感谢 Jane Bonnell 和 Pavithra Jayapaul 给予我们默契的配合。

非常感谢那些允许将那些图和示例复制到本书中的出版商们。从 Complete Systems Analysis (Robertson and Robertson, 1994)、Mastering The Requirements Process (Robertson and Robertson, 1999) 中选用的材料得到了 Dorset House Publishing 的许可 (www.dorsethouse.com, 版权所有)。练习 1.1 中的材料是经 Associated Press 许可后从 *Washington Post* 复制过来的。图 12.15 和图 12.16 是经 John Wiley 和 Sons Limited 许可后从 Barghouti 等 (1995) 中复制而来。图 12.14 和图 12.15 是经 John Wiley 和 Sons Limited 许可后从 Rout (1995) 中复制而来的。

第 2、3、4、5、9、11、12 和 14 章中注明为 IEEE 版权的图和表是经 IEEE 许可后复制的。同样地, 第 14 章中注明为 ACM 版权的三张表均是得到美国计算机协会许可后复制的。表 2.1 和图 2.11 (Lai, 1991) 是经软件生产力联盟许可后复制的。图 8.16 和图 8.17 (Graham, 1996a) 是经 Dorothy R. Graham 许可后复制的。图 12.11 和表 12.2 (Liebman, 1994) 是经 Center for Science in the Public Interest, 1875 Connecticut Avenue NW, Washington DC 许可后所采用。表 8.2、表 8.3、表 8.5 和表 8.6 是经 McGraw-Hill 公司许可后所复制。源于 Shaw 和 Garlan (1996)、Card 和 Glass (1990)、Grady (1997), 以及 Lee 和 Tepfenhart (1997) 的图和表是经 Prentice Hall 公司许可后所复制。

表 9.3、表 9.4、表 9.6、表 9.7、表 13.1、表 13.2、表 13.3 和表 13.4, 以及图 1.15、图 9.7、图 9.8、图 9.9、图 9.14、图 13.1、图 13.2、图 13.3、图 13.4、图 13.5、图 13.6 和图 13.7 是经 Norman Fenton 部分或完全的许可, 从 Fenton 和 Pfleeger (1997) 中复制或改编过来的。图 3.16、图 5.19 和图 5.20 是从 Norman Fenton 的课程笔记中复制或改编过来的, 感谢他友善的许可。

特别感激我们的职员、RAND 公司和 Waterloo 大学, 尤其是他们在我们准备

此版本时对我们的鼓励^①。同时，也感谢我们的朋友和家人，编写此书占用了他们的时间，感谢他们付出的善意、支持和耐心。特别地，Shari Lawrence Pfleeger 感谢 Manny Lawrence；感谢皇家服务站的经理和他的会计 Bea Lawrence，不仅感谢她的工作和她的学生对皇家系统的规格说明，而且也包括他们其他工作对我们的影响和指导；像她的父母一样。Jo Atlee 特别感谢她的父母 Nancy 和 Gary Atlee，他们支持和鼓励她所做的任何事情(包括尝试)，还感谢她的同事和学生，他们在她写作的主要时期友好地分担了很多工作。最为特别的是，我们要感谢 Charles Pfleeger 和 Ken Salem，谢谢他们的支持、鼓励和幽默感。

Shari Lawrence Pfleeger
Joanne M. Atlee

^① 请注意本书并不属于 RAND 公司的产品，并且未经过 RAND 公司的质量确认过程。我们为该书所做的工作仅说明我们是该书的作者，而不是为了说明我们是各自所属机构的雇员。

作者简介

Shari Lawrence Pfleeger (信息技术和工程博士, George Mason University; 规划专业硕士, Pennsylvania State University; 数学专业硕士, Pennsylvania State University; 数学专业学士, Harpur College) 是 RAND 公司的一位高级信息科学家。她当前的研究兴趣着重于那些可帮助公司和政府机构了解信息技术如何来支持其任务和目标的策略和决策问题。她在 RAND 公司的工作涉及帮助客户构建软件评估程序, 支持政府机构定义信息保险策略, 并为有关计算机安全以及美国国家安全提供决策支持。

在加入 RAND 公司之前, 她是 Systems/Software 公司的总裁, 该公司是专长于软件工程和技术的咨询公司。她曾经担任 City University (伦敦) 和 University of Maryland 的访问学者, 并且是 Howard University 软件技术评估研究中心的发起人和总监。她编写了多部有关软件工程和计算机安全的图书。Pfleeger 以软件工程的经验式研究以及以多学科方式来解决信息技术问题而著称。她曾担任 IEEE Software 的副总编、IEEE Transactions on Software Engineering 的副编辑以及 IEEE Security and Privacy 的副编辑, 同时是 IEEE Computer Society Technical Council 关于软件工程的委员会成员。同时她也是诸多相关领域大会和讨论组的演讲人。Systems and Software 杂志多次提到 Pfleeger, 将她视为世界顶级的软件工程研究学者之一。

Joanne M. Atlee (计算机专业博士和硕士, University of Maryland; 计算机科学和物理专业学士, College of William and Mary; 工程硕士) 是 Waterloo 大学计算机科学学院的副教授。她的研究兴趣着重于软件建模、文档和分析。她以模型检测软件需求规格方面的工作而著称。她的研究还涉及基于模型的软件工程、模块化软件开发、特性交互 (feature interaction) 以及形式软件开发技术的效能分析。Atlee 担任过 IEEE Transactions on Software Engineering、Software and Systems Modeling 以及 Requirements Engineering Journal 的编委会成员, 并且是信息处理国际联盟 (International Federation for Information Processing, IFIP) 工作组 2.9 的副主席 (IFIP 是致力于软件需求工程方面研究的国际性研究组织)。她还是第 31 届国际软件工程大会 (International Conference on Software Engineering, ICSE'09) 程序委员会联合主席。

Atlee 对软件工程教育充满浓厚的兴趣。她是 Waterloo 大学的软件工程学士

教学计划发起人。她还是 ACM/IEEE – CS Computing Curricula – Software Engineering (CCSE 卷) 的指导委员会 (Steering Committee) 成员。该指导委员会负责为本科生软件工程教学计划提供课程指导。她还是加拿大工程资格委员会 (Canadian Engineering Qualifications Board committee) 的委员, 该委员会负责设置软件工程课程提纲, 为加拿大各省工程联盟提供相应的指导, 以帮助它们确定那些从事软件工程专业工程师资格认证应当达到的学术资格标准。