TURING 图灵计算机科学丛书

PEARSON Prentice Hall

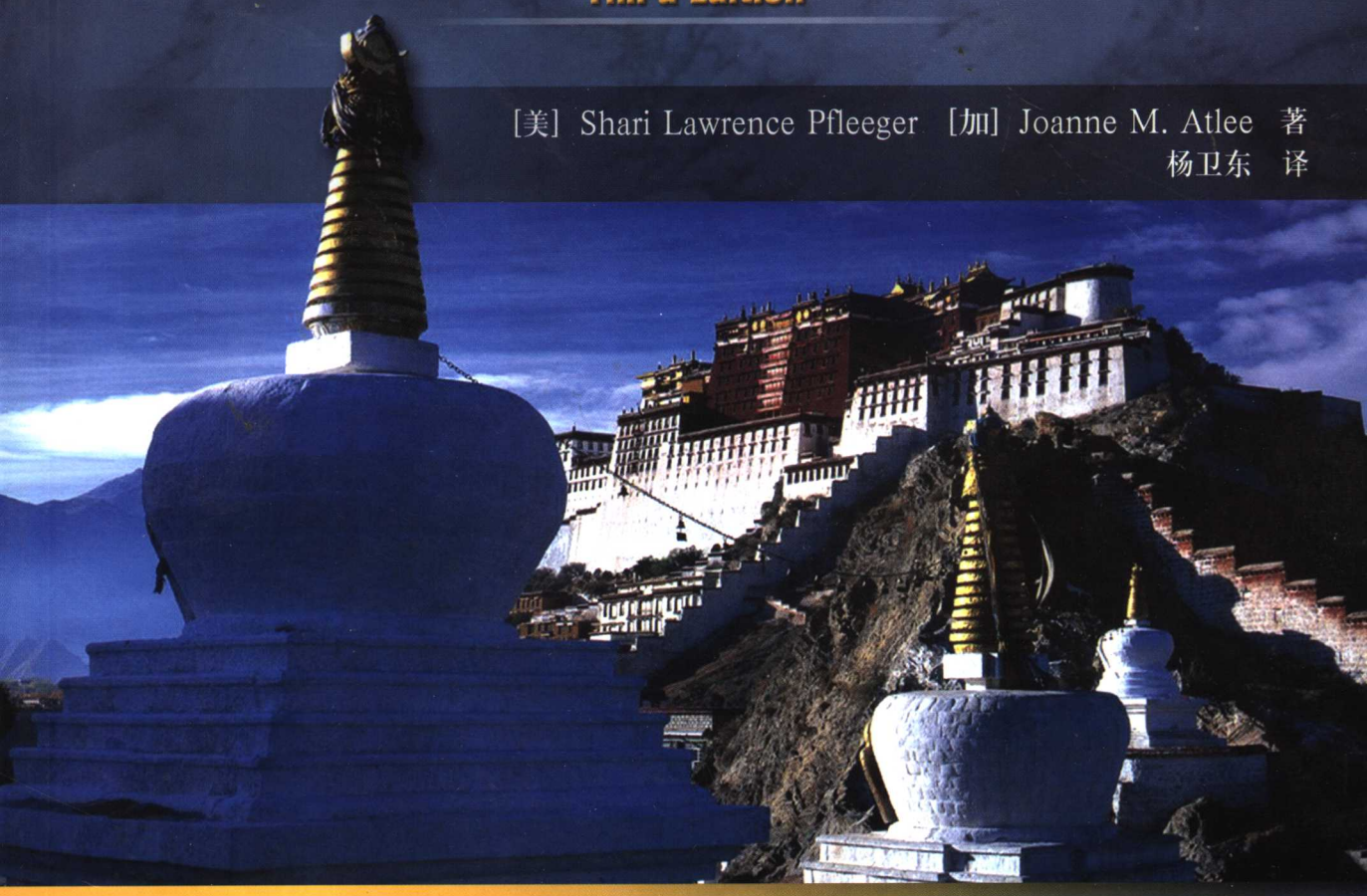
软件工程

（第3版）

Software Engineering: Theory and Practice

Third Edition

[美] Shari Lawrence Pfleeger [加] Joanne M. Atlee 著
杨卫东 译



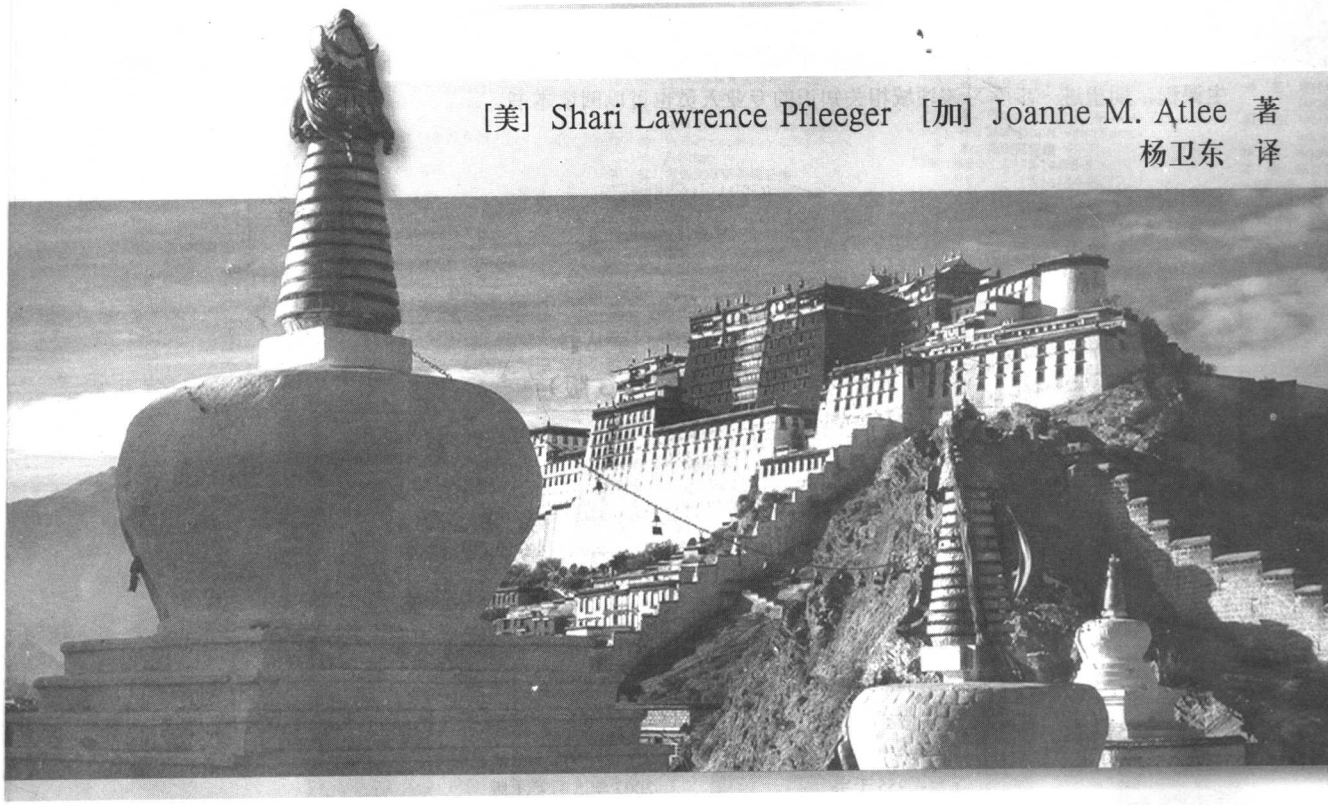
人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵计算机科学丛书

软件工程
（第3版）

Software Engineering: Theory and Practice
Third Edition

[美] Shari Lawrence Pfleeger [加] Joanne M. Atlee 著
杨卫东 译



人民邮电出版社
北 京

图书在版编目（CIP）数据

软件工程：第3版/（美）弗里格（加）阿特利著；杨卫东译.
—北京：人民邮电出版社，2007.5
（图灵计算机科学丛书）

ISBN 978-7-115-15829-1

Ⅰ.软… Ⅱ.①弗…②阿…③杨… Ⅲ.软件工程—教材 Ⅳ.TP311.5

中国版本图书馆CIP数据核字（2007）第016426号

内 容 提 要

本书是软件工程领域的经典著作，国际上众多名校均采用本书作为教材。本书分为3个部分。第一部分旨在引起读者学习软件工程的兴趣，解释为什么软件工程知识对实践者和研究者是同样重要的，然后在论证过程模型的同时，也强调了敏捷方法的必要性，最后讨论项目计划的相关概念，以及它与软件开发过程的相关性。第二部分论述开发和维护的主要步骤：引发、建模和检查需求，设计问题的解决方案，编写和测试代码，以及将软件交付给客户。第三部分主要讲述软件评估和改进，着眼于如何评价过程和产品的质量，以及如何采取措施改进。

本书适合作为计算机相关专业软件工程课程的本科教材，也适用于介绍软件工程的概念与实践的研究生课程，期望进一步学习该领域相关知识的专业人员也可以阅读本书。

图灵计算机科学丛书

软件工程（第3版）

◆ 著 [美]Shari Lawrence Pfleeger [加]Joanne M.Atlee
译 杨卫东
责任编辑 杨海玲

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销

◆ 开本：787×1092 1/16
印张：30.25
字数：955千字 2007年5月第1版
印数：1-5 000册 2007年5月北京第1次印刷

著作权合同登记号 图字：01-2006-0316号

ISBN 978-7-115-15829-1/TP

定价：59.00元

读者服务热线：(010)88593802 印装质量热线：(010)67129223

版 权 声 明

译 者 序

没有简单说教，而是摆事实、讲道理、进行客观评价，这是我喜欢本书的首要原因。本书通过一些翔实的实际案例说明软件开发中成功的经验与失败的教训。对软件工程中的方法与技术，本书不是简单地告诉你“应该如何”、“不该如何”，而是针对不同情况，讨论其正反两方面的作用；不是简单地“高度评价”，而是讨论它“已经做到的”、“不能做到的”以及软件工程界现在和将要努力的方向。这是一种实事求是的治学态度。软件工程中，虽然存在主流的方法、过程和表示法，但是，没有哪一种可以“一统天下”，能够称为“终结者”；过去、现在和将来，都概莫能外。软件工程还在不断发展，以动态的、客观的眼光来对待软件工程，有助于我们在软件开发中正确地进行权衡和决策。

没有忽视人的作用，而是不断地加以强调，这是我喜欢本书的第二个原因。软件开发通常是创造性过程，而不是机械制造过程；是团队协作的过程，而不只是个人行为。改进工具和技术，只能间接提高软件质量，而“人”才是直接动因。项目成员之间的交流、团队士气、项目成员的经验与能力等与人有关的因素，都会对项目目标产生重要影响。另外，本书还提出一些与人有关的法律、道德和责任问题，这能够引发读者进行思考，也很有趣。

没有只“坐而论道”，而是将理论与实际结合起来，这是我喜欢本书的第三个原因。离开实践，理论与原则都只是“空中楼阁”；相反，忽略理论与原则的指导作用，实践工作则会走许多弯路。软件工程尤其如此。本书通过软件工程中的各种模型与原则，架起了软件工程研究与实践之间的桥梁；通过两种类型的应用系统（信息系统和实时系统），论述软件工程理论与原则的具体应用；通过贯穿始终的一个课程项目，使学生能够通过亲身实践来掌握软件工程的基本方法。

本书的另外一些特色是：比较全面地涵盖了软件工程的方法和技术。书中包括过程与生命周期模型、获取需求、项目管理、设计系统、编写程序、测试程序与系统、交付与维护系统、面向对象技术等，读者可以较全面地理解和熟悉软件工程的各个方面，而不只是针对特定的方法、过程和技术；将风险管理和质量工程融合到软件开发的各个阶段，有助于读者理解它们并将其真正应用到实践开发中；讨论了一些近年来新的软件工程方法和技术，如极限编程、设计模式等，并与传统的开发方法进行比较，有助于读者理解和在实际开发中进行选择。

我这里还想说的一点是：不会有哪一本书能够包罗万象。软件工程发展很快，对一些技术，本书只是提及或者进行概括性论述，如设计模式、重构、形式化方法、CASE工具等；而有一些技术，本书很少提及。不过，本书对参考文献的讨论，为读者提供了很好的线索。读者在需要的时候，可以进一步参考其他相关文献。

随着近年来软件工程方法和技术的发展，出现了很多新的术语，而这些术语在国内不同文献中的翻译也不尽相同。译者参阅了众多国内文献和前面的版本，尽量选择与原文本意贴近的、主流的翻译方式。主要术语的翻译可在索引中查阅。本书内容广泛，涉及的素材众多，因而增加了翻译的难度。再者，本书作者是软件工程领域的著名专家和学者，书中有许多独到之处，由于译者水平有限，加之时间过于紧迫，难免有翻译不当之处，敬请读者批评指正。

最后，我要感谢人民邮电出版社图灵公司的编辑陈舜贤、杨海玲等，同时感谢对本书翻译有帮助的人以及我的家人。

杨卫东

2007年1月于复旦大学

前　言

用建模和设计跨越研究与实践之间的鸿沟

自从1968年“软件工程”这一术语在NATO会议[1]上首次使用以来，软件工程已经走过了很长的一段路。如今，软件本身已经以各种形式融入我们的生活，这种局面即使在十年前，也很少有人预料到。因此，坚实的软件工程理论和实践基础，对于理解如何开发优质软件，以及评估软件在日常生活中所遭遇的风险和存在的机遇是不可或缺的。本书体现了当前软件工程领域两个阵营（即实践者和研究者）之间的相互融合。其中实践者主要关注构造完成某些功能的高质量产品；而研究者则努力寻找各种方法改进产品质量以及提高开发人员生产效率。Edsgar Dykstra不断提醒我们：严密认真的研究与实践将检验我们对软件工程的理解，并帮助我们改进思维方式、方法，进而最终改进我们的产品。我们正是本着这种精神对本书进行了增订，为这种不断的探究和改进的过程构造一个基础框架。

要构建这样的框架，在研究与实践之间架设桥梁，关键是建模和设计的理念。软件工程师决不能像程序员那样只遵从设计说明，就如同厨师长决不能像厨师那样只遵从菜单一样。构造优秀的软件是一门艺术，这体现在如何抽象出问题的本质要素，对它建模，并使用这些抽象设计出解决方案。经常能听到优秀开发人员谈论“优雅的”解决方案，这是在说解决方案抓住了问题的核心，因此软件不仅能够解决眼前的问题，而且当问题随着时间演化时，软件也能够很容易地进行修改。正因为这样，本版包含广泛的素材，讨论如何对问题进行抽象和建模，以及如何使用模型设计合适的解决方案。这样，学生就能够学会融合研究与实践、艺术与科学，构造坚实的软件。

科学总是以事实为基础的。本书是为本科生软件工程课程而设计的，注重软件工程研究与实践的实用层面，使学生能够直接将所学知识应用于要解决的现实问题。书中所举的例子针对的是经验有限的学生，但是，这些例子清楚地阐明大型软件开发项目是如何从需求到概念，进而成为现实的过程。例子所描述的许多情形，读者未来都可能经历：大型项目与小型项目、“敏捷”方法与结构化方法、面向对象与过程的方法、实时与事务处理、开发与维护场景。

本书也适用于介绍软件工程的概念与实践的研究生课程，还适合于那些期望进一步学习该领域相关知识的专业人员。尤其是第12章至第14章给出了一些引人思考的资料，旨在引起研究生对当前研究主题的兴趣。

主要特色

本书区别于其他书的主要特色是：

- 其他软件工程图书将测度和建模作为单独的问题考虑，本书则将测度和建模与更全面的软件工程论述结合起来。也就是，把测度和建模当作软件工程策略不可分割的部分，而不作为一个单独的分支。这样，学生能够学会如何抽象与建模，以及如何在日常开发活动中进行定量评估和改进。学生可以利用他们的模型理解所要解决的问题的要素，以及可供选择的其他解决方案；可以利用测度对个人、小组和项目的进度进行评估。
- 类似地，诸如复用、风险管理和质量工程之类的概念都揉合到相应的软件工程活动中，而不

1. 1967年，在北大西洋公约组织（NATO）科学委员会的计算机科学组召集关于“软件工程”主题的国际会议时，软件工程这一术语首次出现。1968年，该国际会议在德国Garmisch地区举行，会议的科技报告可在下面网址下载：http://homepages.cs.ncl.ac.uk/brian.randell/NATO/。——译者注

是作为单独的问题来讲述。

- 当前这一版本强调对敏捷方法（包括极限编程）的使用。它论述了给予开发人员更多自主权所带来的利益和遭遇的风险，并将敏捷方法与传统的软件开发方法进行比较。
- 每章都将相应的概念应用于两个贯穿全书的例子中：一个例子是信息系统，另一个是实时系统。两个例子都基于实际的项目。信息系统的例子描述了一个大型英国电视公司确定广告时间价格的软件。实时系统的例子是阿丽亚娜5型火箭的控制软件，我们将考查该软件见诸报端的一些问题，并探讨如何利用软件工程技术找出其中的问题并加以解决。学生能够随着这两个有代表性的项目的进展，领会本书所描述的各种实践方法如何融入到构造系统的技术之中。
- 在每章的结尾，结论都用3种形式表述：这一章的内容对开发团队的意义，对单个开发人员的意义，以及对研究者的意义。学生可以很容易地复习每章的要点，并了解每一章与研究以及实践的相关性。
- 本书有配套网页，可从Pretence Hall的公司网站http://www.prenhall.com中进入。该网页包含有最新文献中的例子，以及实际项目中的实际制品的例子。该网页还包含到相关工具和方法厂商的网页的链接。学生可以在这里找到实际的需求文档、设计、编码、测试计划以及更多内容。那些想寻找更多、更深入信息的学生，可以通过该网页找到著名的网上出版物及其网站。这些网页会定期更新，以保证本书中的资料是最新的。读者还可在此向作者和出版社提供反馈意见。
- 本书还配备一张教学辅助CD，可从Prentice Hall的院校代表或通过本书网站申请得到。这张CD包括学生学习指导、习题解答，以及帮助教师备课的一些素材（如带有书中图和表的PowerPoint幻灯片）。
- 本书含有来自各种文献的大量案例研究和例子。书中以“补充材料”形式给出的很多只有一页左右篇幅的案例研究都在网页上加以详细描述。据此，学生能够了解本书中的理论概念是如何应用于现实情形的。
- 每章都以引人思考的、软件工程中有关的法律和道德问题作为结束。学生可以根据软件工程所处的现实背景来理解软件工程。与其他的学科一样，软件工程的结果会影响到人，必须据此来考虑软件工程决策。
- 每章都强调过程和面向对象的开发。另外，第6章是专门讲述面向对象开发的，解释了面向对象开发过程的步骤。从需求规格说明一直到程序设计，都使用UML表示法，每一步骤都有实际应用的例子。
- 本书还有一个加了注解的参考文献，其中包含很多软件工程方面开创性的论文。另外，本书的网页有指向含注释参考文献和特定领域讨论组（诸如软件可靠性、容错性、计算机安全性等）的链接。
- 每章包括一个学期项目的描述，它是一个住房按揭处理系统软件的开发，教师可以把这个学期项目或它的变体作为课程作业。
- 每章的结尾都列出这一章概念的主要参考文献，学生能够由此深入地查找这一章中讨论过的特定工具和方法的信息。

本书的内容和组织结构

本书分为3个部分。第一部分引起读者学习软件工程的兴趣，解释为什么软件工程知识对实践者和研究者同样重要。第一部分还讨论理解过程问题的必要性，以及为什么需要决定开发人员要有多大程度的“敏捷性”，为什么需要精细地进行项目计划。第二部分论述开发和维护的主要步骤：引发、建模和检查需求，设计问题的解决方案，编写和测试代码，将软件交付给客户。这些步骤与构造软件所使用的过程模型关系不大。第三部分主要讲述软件评估和改进，着眼于如何评价过程和产品的质量，

以及如何采取措施改进。

第1章：软件工程概述

这一章给出软件工程的发展历程，以激励读者的学习兴趣，并强调在后面的章节中要验证的某些关键问题。尤其是，讨论Wasserman用来帮助定义软件工程的关键因素：抽象、分析与设计方法和表示法、模块化与体系结构、软件生命周期与过程、复用、测度、工具与集成环境以及用户界面与原型化。我们讨论计算机科学和软件工程的差异，解释可能遇到的一些主要问题，并为本书的其余部分奠定基础。还探讨采用系统的方法构造软件的必要性，并介绍每一章都会用到的两个公共的例子。学期项目的背景在这里介绍。

第2章：过程和生命周期的建模

这一章概括介绍不同类型的过程和生命周期模型，包括瀑布模型、V模型、螺旋模型和各种原型化模型；论述敏捷方法的必要性。与较传统的软件开发过程相比，在敏捷方法中，开发人员拥有许多自主权；还描述几种建模技术和工具，包括系统动态建模，SADT以及其他常用方法。使用本章介绍的一些技术分别对两个公共的例子进行部分建模。

第3章：计划和管理项目

在这一章，我们着眼于项目计划和进度安排，介绍诸如活动、里程碑、工作分解结构、活动图（activity graph）、风险管理、成本以及成本估计等概念。用估计模型估计两个公共例子的成本和进度。我们重点进行实际案例研究，包括F-16战斗机和DEC的alpha AXP程序的软件开发管理。

第4章：获取需求

本版中，这一章的内容进行了较大修订，强调在优秀的软件构造中抽象和建模的关键作用。尤其是，我们使用模型澄清用户需求中的误解和遗漏的细节，并使用模型与其他人员进行沟通。这一章中探讨了多种不同的建模范型；针对每一种范型，研究表示法实例；讨论什么时候使用哪种范型，并对如何做出某些建模和抽象决策给出建议。这一章还讨论了需求的不同来源和类型（功能性需求、质量需求与设计约束）；解释如何编写可测试的需求，并描述如何解决其中的冲突。其他讨论的主题包括需求引导、需求文档、需求评审、需求质量及其测度，并举了一个如何选择规格说明方法的例子。这一章最后将其中几个方法应用于两个公共示例。

第5章：设计系统

这一章集中讨论体系结构问题，首先讨论Shaw和Garlan关于软件体系结构的框架；接着，描述概念性设计和技术性设计的区别。我们讨论执行设计的人员所扮演的角色，并描述组合和分解两种基本的设计方法。然后，确定一个优秀设计所应该具有的特征，介绍几种设计策略，并举例说明几种系统设计技术和工具。这一章中，读者可以学习到客户-服务器体系结构、可复用的设计构件、人机界面设计、安全和可靠系统的设计（包括错误处理和容错）、设计模式、形式化设计方法以及如何评价设计的优劣。在解释如何评估和确认设计质量以及如何归档结果之后，转而讨论程序设计问题。

另外，还阐述了程序设计的一些原则，包括自顶向下与自底向上、模块化和独立性，以及逻辑设计和物理设计之间的区别；讨论并发系统以及安全攸关系统的设计，并分析导致Therac-25故障的设计缺陷；描述了几种设计工具，并详尽讨论设计质量以及如何测量设计质量；介绍设计复用、评审、审查，并解释设计方案的合理归档的必要性。最后，以信息系统和实时系统的设计例子作为这一章的结束。

第6章：细述对象

第6章再次考虑了面向对象开发的特殊性质。这一章从描述用例场景开始，讨论如何从自然语言描述的需求中获取对象及其性质。利用第4章介绍的建模工具，将需求表述为各种模型，这些模型合在一起，从各种视角描述了所要解决的问题。接着，研究系统设计，了解如何使用模型生成解决问题所必需的高级信息。然后，细化系统设计，增加需要的非功能需求以及程序设计所需要的细节。采用UML及其结构，为皇家汽车服务站（Royal Service Station）生成其面向对象的规格说明和设计。

详细讨论面向对象的测度，将一些通用的面向对象度量标准应用于皇家汽车服务站例子，并指出如何利用度量标准中的变化来帮助我们分配资源并查找错误。最后，把面向对象的概念应用到我们的

信息系统和实时系统的例子中。

第7章：编写程序

这一章论述代码层的设计决策，以及实现一个设计以产生高质量代码所涉及的问题；讨论标准和过程，并给出一些简单的编程准则；提供用多种语言编写的例子，包括面向对象语言和过程语言；讨论程序文档和错误处理策略的必要性。这一章最后将一些概念应用于两个公共的例子。

第8章：测试程序

这一章探讨测试程序的一些方面的问题。我们把传统测试方法与净室方法区分开，并着眼于如何测试各种系统；给出软件问题的定义和分类，讨论怎样利用正交缺陷分类使数据汇集和分析更加有效；随后，解释单元测试与集成测试之间的区别；在介绍几种自动化测试工具和技术之后，再解释测试生命周期的必要性，以及如何将这些工具集成到生命周期中；最后，把这些概念应用于两个公共的例子。

第9章：测试系统

这一章首先介绍系统测试的原理，包括测试套件和数据的复用，并讨论精细地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；讨论测试面向对象系统的特殊需要；描述几种测试工具，并讨论测试小组成员的角色；然后，向读者介绍软件可靠性建模，并讨论了可靠性、可维护性和可用性问题。读者将学会如何使用测试结果来评估交付的产品可能具有的特征。这一章还介绍几种测试文档，最后，描述两个公共例子的测试策略。

第10章：交付系统

这一章讨论培训和文档的必要性，并给出信息系统和实时系统例子中可能使用的几个培训和文档的例子。

第11章：维护系统

这一章强调系统变更的结果；解释在系统生命周期的过程中变更是如何发生的，以及系统设计、编码、测试过程、文档须如何与这些变更保持一致；讨论常见的维护问题，以及精细地进行配置管理的必要性；全面讨论使用测度预测可能的变更，并评估变更所产生的影响；还讨论在使遗留系统再生的大背景下的再工程和重组技术。最后，根据变更的可能性对两个公共的例子进行评估。

第12章：评估产品、过程和资源

由于许多软件工程的决策涉及合并和集成现有构件，这一章讨论评估过程和产品的方法；讨论经验性评估的必要性，并给出若干例子以说明如何使用测度建立质量和生产率的基线。我们研究几个质量模型，考虑如何评估系统的复用性，如何执行事后分析，以及如何理解信息技术的投资回报。这些概念都应用于两个公共的例子。

第13章：改进预测、产品、过程和资源

这一章建立在第11章的基础之上，说明如何完成预测、产品、过程和资源改进。包含几个深入的案例研究，以说明如何通过多种调查技术来理解和改进预测模型、检查技术及软件工程的其他方面。这一章最后给出一组指导原则，用于评估当前情形并识别改进的机会。

第14章：软件工程的未来

在最后这一章，讨论软件工程中若干悬而未决的难题。我们重新回顾Wasserman的概念，将它作为一个原则来了解我们做得怎么样；研究技术转让和决策制定中的若干问题，以确定我们是否成功地把重要的思想从研究应用于实践方面。作为本版的新内容，我们研究一些有争议的问题，比如，软件工程师是否应该持证上岗，以及针对具体领域提供解决方案和方法的趋势。

致谢

本书的写作仰仗朋友及家人提供技术与情感方面的支持，本书才得以完成。在此，不可能列举出所有在本书的编写和修订期间帮助过我们的人。若忽略了一些人，在此提前表示歉意。我们很感谢本书早期版本的读者，感谢他们经过详细审查对本书的修正提供了良好建议。就我们所知，所有这样的建议都已结合到本版中。我们一如既往地感谢来自读者的反馈，无论是正面的还是负面的。

Carolyn Seaman（马里兰大学巴尔的摩校区）是本书第1版非常杰出的审稿人。她提出许多使内容简单、明了的方法，使得本书更紧凑、更易于理解。她还准备了大部分的练习解答，并帮助我们建立本书早期版本的网站。我要感谢她的友谊和帮助。Yiqing Liang和 Carla Valle更新了该网站，并为第2版增加了重要的新资料；Maria Augusta Vieira Nelson（巴西米纳斯吉拉斯州天主教大学）修订了本书第3版的网站，特别是有关建模表示法和敏捷方法的内容。

我们非常感谢Forrest Shull（马里兰法朗霍夫中心）以及Roseanne Tesoriero（华盛顿学院），他们编写了本书最初的学习指导，还要感谢Maria Nelson，她对第3版的解答手册及学习指导进行了修订。我们要特别感谢Guilherme Travassos（里约热内卢联邦大学），本书使用了他与Pfleeger在马里兰大学期间共同编写的材料，而且他在以后的课程教学中，又对这些材料进行了大量扩充。

对我们有所帮助且颇有创见的本书所有3个版本的审稿人有：Barbara Kitchenham（英国基尔大学）、Bernard Woolfolk（朗讯公司）、Ana Regina Cavalcanti da Rocha（里约热内卢联邦大学）、Frances Uku（加利福尼亚大学伯克利分校）、Lee Scott Ehrhart（MITRE公司）、Laurie Werth（得克萨斯大学）、Vickie Almstrum（得克萨斯大学）、Lionel Briand（渥太华卡尔顿大学）、Steve Thibaut（佛罗里达大学）、Lee Wittenberg（新泽西基恩学院）、Philip Johnson（夏威夷大学）、Daniel Berry（加拿大滑铁卢大学）、Nancy Day（滑铁卢大学）、Jianwei Niu（滑铁卢大学）、Chris Gorringe（东英吉利大学）、Ivan Aaen（奥尔堡大学）以及Prentice Hall公司提供的一些不具名的评阅者。与下列人员的讨论使本书得到了许多改进和增强：Greg Hislop（德雷克赛尔大学）、John Favaro（意大利Intecs Sistemi公司）、Filippo Lanubile（意大利巴里大学）、John d'Ambra（澳大利亚新南威尔士大学）、Chuck Howell（MITRE公司）、Tim Vieregge（美国军方计算机应急反应组）以及James Robertson 和 Suzanne Robertson（英国大西洋系统协会）。

再一次感谢Toni Holm以及Alan Apt，他们使本版的创作显得有趣并相对轻松。还要感谢James Robertson和Suzanne Robertson让我们使用皮卡地里例子，还要感谢Norman Fenton同意使用我们合著的《软件度量》一书中的资料。

这里，我们非常感谢一些出版商允许我们引用其中的一些图和例子。来自于*Complete Systems Analysis*（Robertson and Robertson 1994）和*Mastering the Requirements Process*（Robertson and Robertson 1994）的资料是从网站www.dorsethouse.com上抽取的，这得到了Dorset House出版社的使用许可。练习1-1中的文章是在美联社的许可下，从《华盛顿邮报》复制的。图2-15及图2-16是在John Wiley and Sons公司的许可下，从Barghouti等人的著作（Barghouti 1995）中复制的。图12-14和图12-15是在John Wiley and Sons公司的许可下，从参考文献（Rout 1995）重印的。

在IEEE的许可下，我们重印了第2、3、4、5、9、11、12和14章中标注IEEE版权的那些图和表。类似地，第14章中标注有ACM版权的3个表，是在ACM的许可下进行重印的。表2-1和图2-11是在软件生产力联合技术发展中心（Software Productivity Consortium）的许可下，复制于参考文献（Lai 1991）。来自参考文献（Graham 1996a）的图8-16、图8-17是在Dorothy R. Graham的许可下重印的。在公众利益科学中心（Center for Science in the Public Interest）的许可下，图12-11和表12-2改编自参考文献（Liebman 1994）。表8-2、表8-3、表8-5和表8-6是在McGraw-Hill公司的许可下复制的。来自于参考文献（Shaw and Garlan 1996）、（Card and Glass 1990）、（Grady 1997）以及（Lee and Tepfenhart 1997）的图和例子是在Prentice Hall公司的许可下复制的。

经过Norman Fenton的部分或全部许可，表9-3、表9-4、表9-6、表9-7、表13-1至表13-3和图1-15、图9-7至图9-9、图9-14、图13-1至图13-7复制或改编于Fenton和Pfleeger的著作（Fenton and Pfleeger 1997）。在Norman Fenton的大度的许可下，图3-16、图5-19和图5-20复制或改编于Norman Fenton的课程讲义。

我们特别感谢各自的单位——兰德（RAND）公司以及滑铁卢大学[1]，在我们准备本版的时候，他

1. 请注意这本书并不是兰德公司的产品，并且没有经历兰德的质量保证过程。我们是以个人身份而不是各自单位职员的身份写作本书。

们都给我们以鼓励。我们感谢朋友及家人的关心、支持和耐心，写作此书使我们不能像平常那样经常相聚。尤其是，Shari Lawrence Pfleeger要感谢皇家汽车服务站的经理Manny Lawrence，也感谢他的簿记员Bea Lawrence，不仅因为与她及她的学生在皇家系统规格说明上的合作，而且因为他们作为父母所给予的关爱和指导。Jo Atlee特别感谢她的父母Nancy和Gary Atlee，他们对她所做的或尝试要做的任何事情都给予支持和鼓励；也感谢她的同事及学生，他们在本书的主要创作期间，愉快地承担了更多的额外工作。我们最要特别感谢的是Charles Pfleeger和Ken Salem，他们是我们获得赏识和持续不断的支持、鼓励和好心情的源泉。

Shari Lawrence Pfleeger
Joanne M. Atlee

目　录