



成功的软件开发

(原书第2版)

Successful Software Development (Second Edition)

Scott E. Donaldson 著
Stanley G. Siegel

蔡愉祖 邓本江 等译



机械工业出版社
China Machine Press



中信出版社
CITIC PUBLISHING HOUSE

成功的软件开发

(原书第2版)

Successful Software Development
(Second Edition)

Scott E. Donaldson
Stanley G. Siegel

著

蔡愉祖 邓本江 等译



机械工业出版社
China Machine Press



中信出版社
CITIC PUBLISHING HOUSE

实现成功的软件开发是软件工作者的目标。本书从文化变更、基本概念、规划改进软件系统的开发“方式”三个层次系统论述了要实现成功的软件开发所涉及的一系列问题。其中包括：项目规划、软件开发、产品和过程评审、测量、变更控制、系统工程环境SEE（由应用开发过程环境ADPE与应用开发技术环境ADTE组成）。

本书描述的技术可以与任何开发技术协同工作。适合于软件开发人员、项目经理和软件客户使用。

Authorized translation from the English language edition entitled *Successful Software Development (Second Edition)* (ISBN:0-13-086824-4) by Scott E. Donaldson & Stanley G. Siegel, published by Pearson Education, Inc. Publishing as Prentice Hall PTR, Copyright © 2001 by Prentice Hall PTR, Inc.

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 of Pearson Education, Inc.

Chinese Simplified language edition published by China Machine Press and CITIC Publishing House Press.

Copyright © 2003 by China Machine Press and CITIC Publishing House Press.

本书中文简体字版由美国Pearson Education培生教育出版集团授权机械工业出版社与中信出版社联合出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2002-0529

图书在版编目（CIP）数据

成功的软件开发（原书第2版）/唐纳森（Donaldson, S. E.），锡根（Siegel, S. G.）著；蔡愉祖等译。—北京：机械工业出版社，2003.6

（软件工程技术丛书——项目管理系列）

书名原文：Successful Software Development (Second Edition)

ISBN 7-111-08066-1

I. 成… II. ①唐… ②锡… ③蔡… III. 软件开发 IV. TP311.52

中国版本图书馆CIP数据核字（2002）第101724号

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：张金梅

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2003年6月第1版第1次印刷

787mm×1092mm 1/16·35印张

印数：0 001-4 000册

定价：59.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

译者序

实现成功的软件开发是软件工作者的目标。我们希望总是能一致、按时、在预算内生产出满足客户需要的软件产品。本书的作者具有丰富的实际软件开发的经验，在这方面给出了许多很有价值的、具体实用的指导。

作者从三个层次论述成功的软件开发问题，这对我们启发很大。首先从文化变更的高度，指出要实现成功软件开发，必须改变原有的工作方式和习惯。卖方要建立一种一致的全组织从事业务的方式，卖方和客户之间要建立一种合作伙伴的关系。文化变更的思想贯穿于整本书中。其次，作者阐述了有关的基本概念，指出很多事情与实现成功软件开发有关。在这些重要的事情中，最为突出的是有效交流、风险降低和组织的成功软件开发的“方式”。作者指出在整个软件项目期间客户和卖方之间必须不断进行有效交流，并特别讲述了变更控制委员会的作用。在风险降低方面，作者阐述了风险降低的项目规划和产品保证的概念和技术。所陈述的组织的业务工作“方式”由一组有关的过程组成，这些过程体现了基本的工程和过程原理。所陈述的组织“方式”包括项目规划过程、变更控制过程、产品和过程评审过程和测量过程。最后，对如何规划改进软件系统的开发“方式”，提供了详细的、具体的、实用的和已验证的指导。针对在软件开发和过程改进中的实际问题，书中提供了大量的实例和具体的解释。我们在工作中所遇到的一些问题，如需求可测试性问题、需求频繁变更问题、风险降低问题、有效项目规划问题、软件开发质量的度量问题、如何着手软件过程改进问题，都能从书中得到启示。

参加本书翻译的有：蔡愉祖、邓本江、孟章荣、石柱、郭晓慧。全书由蔡愉祖校对和统稿。由于水平有限，错误难免，欢迎读者批评指正。

译者

前言

千万不要承诺我们不应该做的事，免得被迫去做我们不能做的事。

——传为亚伯拉罕·林肯1856年5月29日，在首届Illinois共和党代表大会之前的讲演。

《亚伯拉罕·林肯文集》，Arthur B. Lapsley编，vol. 2, p. 249(1905).

成功的软件开发意味着“‘一致地’生产‘好的’软件系统的能力”

客户要软件系统具备预计的功能，按时交付，费用不超支，并且满足他们曾明确的其他任何准则。卖方要他们开发的系统达到客户要求，提前交付或按时交付，赢得足够利润，并且满足指导其业务模式的所有准则。满足客户和卖方双方准则的软件系统是“好的”系统。客户和卖方还要求“一致地”满足他们的准则。软件开发工作不应该是像玩彩票那样全凭运气。

本书是指导实践者实现成功软件开发的指南。

实现成功的软件开发意味着“实施一种成功软件开发的‘方式’”

不存在某种开发软件系统的“方式”。如果存在，那么软件系统开发早就可以简化为流水线作业。具有不同经验和教育背景的人在推进软件开发的方法论、过程、技术、实践和工具上做出许多贡献。这种丰富的多样性导致用不同“方式”去“一致地”开发“好”的软件系统。

本书是指导实践者以适应自己环境的方式实现成功软件开发的指南。

本书的读者对象

软件产品和服务涉及到客户和卖方，软件开发工作是以客户和卖方的关系为中心的。因此，本书既是为软件客户写的，又是为软件卖方写的。更具体地讲，本书的读者是从事以下活动的人员：

- 开发软件产品和软件相关产品。
- 直接管理做以上工作的人员。
- 管理上述经理。
- 购买或使用上述活动生成的产品。
- 培训上述人员。

不少人已经从本书的第一版学到了他们所需的技能。客户已经使用本书精化了与卖方打交道的方式。卖方已经使用本书建立或精化了为客户开发软件系统的“方式”。卖方还使用本书对其市场人员进行内部培训，使他们能更好理解其企业当前的产品。客户和卖方已经使用本书去培训他们的人力资源部的人员，使他们能更好地理解软件开发业务需要何种技能组合。大学已经在研究生院使用本书去教学生，如何在软件开发业务中取得成功。

对于**软件客户**，我们解释并用事例说明有效地与软件卖方交流的机制，交流中说明（1）你要什么，（2）何时要，（3）打算付多少钱。

对于**软件卖方**，我们解释并用事例说明有效的交流机制，（1）与客户交流你对客户的要求的理解，（2）在项目团队成员中，交流你打算怎样给客户他所要的东西。

对于**教师**，我们提供课堂教学的补充培训材料。这些材料被分成以下几类学习指南：

- 500多页的演示材料，重新组织了本书内容。材料按章组织，内容精心安排。本书的绝大多数图，有的经过修改，均在此材料内。
- 家庭作业问题样本。
- 课堂项目样本。
- 教学大纲样本。

教师可以使用以上材料连同网站www.phptr.com/ptrbooks/ptr_0130868264.html去开发一些基于本书材料的课程。这些课程可以作为公司培训的一部分，或者列入大学的课程中。这份学习指南材料来自以上两种环境下的教学经验。

关于学习指南中的演示材料，无需指导老师，学生就能使用该材料作为本书的参考。具体的使用方法有：

- 在阅读本书的一章或其一部分之前，学生可阅读相应部分的学习指南，以便快速浏览该章或这一部分。
- 当阅读一章或其一部分时，学生可并行地阅读相应部分的学习指南。有些时候，从不同角度学习相同材料能促进学习。
- 在阅读一章或其一部分之后，学生可阅读相应部分的学习指南作为复习，迅速回想起关键点、概念和书内实例。

本书与其他软件开发书的区别

有很多因素与实现成功软件开发有关。在这些重要事情中，最为突出的是：有效交流、风险降低和组织的成功软件开发的“方式”，它们是贯穿全书的线索。

有效交流意味着“传送信息、思想或感情，以致它被圆满地接收或理解[加上重点]”。^①

① 该定义源自字典Merriam-Webster's Collegiate Dictionary, Tenth Edition (Springfield, MA:Merriam-Webster, Inc.,2000) 里“交流”定义中的一条陈述。我们注意到某些字典中在“交流”的定义中包括“有效”的概念。有人会认为我们画蛇添足，将“有效”和“交流”并列起来。其实，我们这样做的理由是要强调以下看法：传送信息、思想或情感的人应该完成这样的传递，使得它能被圆满地接收或理解。

如果人们仅仅去理解创建软件代码的机制,则客户和卖方在相互了解上会存在困难,这样过分简单化的交流造成很大风险。客户说:“我们认为已经告诉了开发者我们要的是什么,但是他们给出的不是我们所要的。”卖方说,“我们认为已经理解了客户力图告诉我们的事,但是结果发现,我们交付的不是客户所要的。”所以,对我们而言:

成功的软件开发首先和首要的是,整个软件项目期间客户和卖方之间不断进行有效的交流。

风险降低意味着“减小软件系统开发出现以下情况的可能性:(1)不能按时交付,(2)无法在预算内交付,(3)不是卖方和客户双方认同的产品。”

简单地说,人们知道创建软件代码时有风险。可是,许多人并不评估风险,安排合适资源去缓解风险、监控风险和决定如何去处理风险。客户说:“我们没有文档化的需求,但是,我们期待系统如何如何”。对这样的客户要求,卖方做出响应说:“没问题!软件会如你所愿的,我们将按时交付。”因此,对我们而言:

成功的软件开发也是一个持续地降低风险的实践。

组织的成功软件开发“方式”的意思就是“组织用于开发和维护软件和软件相关产品的一组过程”。从这个前提出发,正如我们面前所说的,不存在构造软件系统的一种方式。同样,不顾过分简单化的工作方式所具有的风险,客户说,“我们没有时间停下来规划将要的工作,那么就开始编码吧!”卖方说:“我们知道用户需要什么,所以让我们开始吧!”因此,对我们而言:

开发软件系统的“方式”应由这样的过程组成,它们(1)增进整个软件系统开发期间的有效交流,(2)不断降低风险。

我们陈述的过程是基于基本的工程和过程原理的,这些原理包括(1)项目规划,(2)变更控制,(3)产品和过程评审。我们陈述一种基于语言的测量技术^①,用于评价软件过程和其生产的产品。我们解释如何使用这样的测量去改进过程和产品。我们解释如何规划过程改进才能有助于开发软件系统方式的改进。我们解释为什么所提的这些思想效果很好,在如何能使这些思想奏效方面提供一些建议,也使你深入了解什么可能会不奏效。

组织的工作“方式”需要包含从人们经历中和以往的软件开发项目中所获得的经验教训。如果组织的“方式”包括了这样的经验教训,已知的和/或预期的风险就降低了,但不一定能全部消除。并且,客户/卖方的有效交流减少了误解,因而减少了软件产品不满足准则的风险。

所以,本书介绍用于有效交流和降低风险的技术。我们解释和用事例说明基本的工程和过程原理,供读者在自己的环境下实现它时考虑。

我们强调指出,已经在现实世界的软件系统开发项目中成功地实施了这些技术。这些项

① 这里,基于语言的测量技术的含义是:它是一种测量技术,能把目标听众所熟悉的语言与在标尺上标出的数字值联系起来。

目的规模从上万美元到上亿美元不等。

本书的组织结构

本书共八章，图P-1显示每一章的标题和目的。具体地讲，这些章分别阐述下列主题：

第1章 介绍建立软件系统开发的“一致”工作方式的业务实例，还引出全书所用的某些基本术语和概念。这些术语和概念是促进有效交流的主要词汇。

第2章 提供用于项目规划和降低风险的技术。许多组织先制定项目计划，然后才开始工作。对我们来说，规划只是组织的开发软件系统“方式”的一部分。

第3章 陈述组织的开发软件系统的“方式”（或过程），即组织的软件系统开发过程。实际上，这个工作“方式”有助于给本书的余下部分设置舞台。对于软件开发，存在许多“最佳实践”。问题是，“这些实践怎样彼此衔接在一起”？所陈述的组织的工作“方式”由一组有关的过程组成，这些过程体现了基本的工程和过程原理，它们特别处理了有效交流和风险降低问题。所陈述的组织“方式”包括项目规划过程、变更控制过程、产品和过程评审过程和测量过程。我们定义和解释角色、职责、活动和交流通信链接。我们提出的这个开发软件系统的“方式”供你在确定业务工作方式时参考。前面我们已讲述了软件开发的一条关键原则——成功的软件开发是一个不断进行的风险降低的实践。在第3章中，我们提出一种软件系统的开发“方式”供你考虑，我们强调对此关键原则的如下推论：

在某种特定情况下，如果你决定不遵循组织的工作方式可能有更好的效果，那么你应该记住，这样做可能增加软件系统开发的风险。

第4章 俗话说，计划赶不上变化。不论规划得多好，一旦项目开始进行，事情就会变化，这几乎是个公理。因此第4章讲述基本的变更控制过程。这一章还讨论了困扰所有软件系统开发项目的交流问题。有时，客户自以为已经明确地告知开发者他的需求了；有时，开发者自以为已了解客户需求了；但在开发者实现需求之后，客户和开发者可能分歧却很大。本章指导在你的环境中减小产生这种分歧的可能性。

第5章 对我们而言，“一致的”软件开发涉及管理、开发、产品保证等系统范畴。第5章从这三个系统开发范畴的角度，讲述产品和过程评审。这一章关注的焦点是评审怎样帮助客户和/或卖方增加对项目进程和风险的可视性，从而能对下一步需要做什么做出明智的、有事实依据的决策。

第6章 为测量而测量只能浪费时间和资源。在怎样用目标听众会同意和了解的日常用语来表述有意义的测量方面，这一章提供实用的指导。有意义的测量将对以下项做出贡献：（1）成功的软件系统开发项目，（2）软件系统开发“方式”上的改进。

第7章 像竞争那样的压力通常推动组织不断改进其技能集合、过程和产品。这一章阐述人员问题，它与使组织的工作“方式”成熟起来一事有关。将软件系统开发过程写在

纸上，试图文档化这个“方式”，是一个难题。可是，要让组织中的人员帮助构造这个“方式”，然后遵循它，则是更为艰难的事情。我们将会介绍应对这种挑战的方法。

第8章 最后，对如何规划改进软件系统的开发“方式”，这一章将提供指导。这一章会帮助你从以前几章中选出一些概念，制定一种过程改进方法。从遵循或不遵循你的工作方式的实例中，可以得到一些经验教训，然后我们会讨论如何把经验教训引入到改进活动中。在改进软件系统开发“方式”方面，我们还将提出其他候选实践供考虑。

目 的	
第1章 业务实例	介绍建立一致的软件系统开发方式的业务实例，并介绍本书其余部分所需的基本概念。
第2章 项目规划过程	采用从项目风险中提取因子去安排项目资源的方法，能有效地规划软件系统的开发工作，本章在这方面提供实用指南。
第3章 软件系统开发过程	定义构成能导致成功的全组织范围的软件系统开发过程的原则，并通过定义一个顶层过程来说明这些原则，你能使用该顶层过程形成环境中的过程框架。
第4章 变更控制过程	定义变更控制委员会（CCB）机制，并指导如何建立一个适用于软件系统开发过程的CCB。CCB是最关键的过程元素，因为它处理困扰任何软件项目的通信问题。
第5章 产品和过程评审	描述与第3章中所引出的各种评审相关联的基本过程，评审能作为降低软件系统开发风险的一种方法，因而能增强成功的可能性。
第6章 测量	在测量产品“优度”和生成产品的软件系统开发过程的“优度”方面提供实用指南。主要集中于如何使用测量以实现一致的产品和过程的“优度”——即实现成功软件系统开发。
第7章 文化变更	阐述在实施系统工程环境（SEE）期间，与促使组织文化变更有关的人的问题，这里SEE定义你所希望的将软件系统工程化的方式。
第8章 过程改进规划	指导如何编写SEE实施计划以建立能实施以前各章所讨论的问题的框架。

图P-1 这本共有8章的书，其组织的方式如图，使读者可以回答问题
“你如何才能实现成功软件系统开发”

表P-1用更详细地说明每章中将学到的东西。

表P-1 每章的重点

章	标题和目的	学习内容
1	业务实例—— (1) 介绍建立 一致的软件系统 开发方式的业	• 成功的软件开发意味着什么 • 为什么在软件过程改进上投资以一致地获得“好的”产品有利于好的业务 • 首要的和最主要的文化变更实践是工作方式的精化/转换

(续)

章	标题和目的	学习内容
	务实例；(2) 介绍本书其余 部分所需的基 本概念	• 成功的软件开发必定是一个客户/卖方合作过程，这里“卖方”是软件系统开发企业，“客户”是卖方向其提供产品的买方/用户 • 本书的思想是可扩展的——它们可运用于几乎任何规模的客户/卖方合作开发软件 • 本书的思想适用于任何业务环境中（例如，商业、政府）的客户/卖方合作关系 • 为什么软件开发业务一定不能像玩彩票那样全凭运气 • 为什么成功的软件开发不能主要（无法解脱地）依赖个体业务骨干完成任务 • 为什么不存在构造软件系统的现成方式，这个观点怎样影响实现成功软件开发的方式 • 为什么处方式运用组织的工作方式有利于好的业务 • “处方式运用组织的工作方式”的含义是什么和为什么处方式运用抓住了使工作方式制度化的关键 • 本书其余部分所需关键术语的定义（例如，软件、软件过程、软件过程能力、软件过程成熟度、处方式运用、产品和过程的“优度”、软件过程改进、生存周期、文化） • 组织的承诺在实现成功软件开发中的作用 • 有效的客户/卖方交流是成功的软件开发的关键元素 • 实现好的客户/卖方交流的关键机制是变更控制委员会（CCB） • 最重要的成功因素是人，而不是自动化工具 • 为获得成功必要的软件系统开发范畴——管理、开发、产品保证 • 实现文化变更的障碍 • 实现软件开发成功的努力远远超出如下工作：（1）用管理命令，（2）组建有经验且高素质的人员团队，（3）与客户谈5分钟，然后3个星期无交往 • 精化/转变组织工作方式的其他方法 • 系统工程环境（SEE）提供实现成功的软件开发的方法——无论系统是顺序开发还是并行开发 • SEE由过程组件（应用开发过程环境（ADPE））和技术组件（ADTE）组成 • 项目计划是一个活着的联结客户/卖方合作关系的合同，它规定卖方的管理、开发和产品保证要完成的工作和客户管理者要批准的工作 • 生存周期在项目规划中的作用 • 规划是在买方/用户和卖方向不断进行的协商 • 怎样说明在整个项目生存周期内必要范畴间的相互作用，必要范畴为管理、开发和产品保证 • 如何规划变更
2	项目规划过程——对有效地规划软件系统开发工作提供实用的指导	

章	标题和目的	学习内容
3	软件系统开发过程——(1) 定义一种构成过程框架的原则,它能在全组织范围内导致成功的软件系统开发; (2) 通过定义一个顶层过程来说明这些原则,这个顶层过程可以作为在环境中定义软件开发工作方式的出发点	• 比较不同的开发工作完成方式——理想的、实际的和现实的,和它们对项目规划的影响 • 如何构造一个简单的、但有效的风险评估方法供项目规划使用 • 怎样在项目计划的预算中明显地考虑风险降低 • 怎样构造风险降低的项目计划 • 如何开发定义组织项目规划过程的ADPE元素 • 软件开发业务中可能产生的合同协议 • 如何编写“好的”工作陈述(SOW),这里SOW是客户用于和卖方交流自己需求的载体 • 卖方怎样才能组建一个项目团队和确定相关的职责完成项目规划工作 • 客户怎样做才能有效地与卖方项目团队交互 • 卖方怎样才能定义如下软件系统开发过程,它(1)在整个过程中明显地包括客户,(2)能包括任何产品开发生存期 • 如何将卖方组织和客户组织纳入到软件系统开发过程中,使双方均了解将怎样开展工作 • 关于软件系统开发过程的“处方式运用”的进一步讨论 • 如何处理在一些环境中的过程问题,这些环境中的大量软件系统开发项目或多或少以并行方式进行 • 如何将生存周期与软件系统开发过程相结合 • 在定义组织软件系统开发过程时,两个主要的考虑是其细节层次和组织范围 • 如何将软件系统开发过程纳入到系统开发过程中 • 设计一个表格去跟踪产品如何曲折地通过软件系统开发过程 • 卖方在将产品交付给客户后,客户和卖方的职责是什么 • 如何开发一个定义组织软件系统开发过程的ADPE元素 • 为什么这个元素是着手建立ADPE的好的出发点 • 为什么不良交流困扰着所有软件系统开发项目 • 为何客户和卖方对产品状态会有截然不同的看法,为了减少出现不同看法的可能性需要做什么 • 怎样管理未经规划的变更以及经规划的变更 • 为什么对于实现成功的软件系统开发,管理所有的变更是至关重要的 • 怎样从管理所有变更的需要产生CCB概念,它远远超出传统的配置管理控制委员会的概念 • 软件系统开发过程的变更控制机制 • 如何通过CCB建立卖方和客户责任 • 所有的变更控制出现在以下三种情景中: 1. 你要什么新的或不同的东西吗? 2. 有什么事错了吗? 3. 我们应该将产品基线化吗?
4	变更控制过程——定义变更控制委员会(CCB)机制,并对于建立软件项目的CCB提供实用指导	

(续)

章	标题和目的	学习内容
		• CCB机制(例如,在CCB会议上记录些什么、通过项目生存周期时CCB的作用、谁应该是CCB主席、CCB的投票机制应该是什么、CCB宪章里包含什么、怎样执行CCB会议、CCB的成员应多长时间相聚等) • 对CCB会议纪要的信息要求 • 怎样写CCB会议纪要 • 什么时候有一个CCB层次结构是合适的和应怎样设置它 • 如何设计变更控制表格,使其有利于组织的工作 • 如何制定一个定义软件系统开发过程中CCB工作的ADPE元素
5	产品和过程评审——描述与第3章所引出的各种评审相关联的基本过程,它作为降低软件系统开发风险的一种方法,因而能获得成功	• 与评审目的有关的原则 • 怎样解决一些主要的、涉及同行评审过程的有关问题 • 独立产品保证组织执行的文档评审和验收测试机制 • 如何使得软件需求是可测试的,以致能引导软件系统开发过程成功 • 什么是软件审核,它与评审的关系 • 对于交付的软件系统和支持数据库的内容,在使卖方和客户达成一致理解方面,验收测试起关键作用 • 高级管理者对于可视性的需要和评审怎样帮助满足这些需要 • 如何将技术编辑纳入软件系统开发过程,以避免对好的工程工作的损害 • 一些技术编辑建议,指导构造组织技术编辑 • 如何制定一些ADPE元素,阐述(1)独立产品保证,(2)同行评审,(3)验收测试周期
6	测量——在测量产品“优度”和生成产品的软件系统开发过程的“优度”方面提供实用指南。主要集中于如何使用测量以实现一致的产品和过程的“优度”	• 了解何时测量才有利于改进软件系统开发过程 • 如何避免“为测量而测量”的活动 • 如何使用易学和易用的目标测量的测量技术,可以用它去测量几乎所有的事情,包括产品和过程的“优度” • 如何建立产品和过程测量含义的测定基准 • 如何测量客户满意度 • 如何量化产品完整性的概念,它作为评估软件产品“优度”的手段 • 如何将量化的产品完整性概念扩展到软件系统开发过程领域以评审过程“优度” • 如何使用产品“优度”度量去跟踪产品通过软件系统开发过程的演化,以发现产品开发的问题 • 如何建立在任何环境中测量产品和过程“优度”的指标尺 • 如何测量第3章所描述的过程“优度” • 如何将目标测量运用到其他软件业务改进方法中,如软件工程研究所开发的那些方法 • 除采用目标测量开发的那些测量之外,如何定义其他的产品和过程度量 • 如何将测量集成到软件系统开发过程中

章	标题和目的	学习内容
7	文化变更—— 阐述在实施系 统工程环境 (SEE)期间， 与促使组织文 化变更有关的 人的问题	• 如何将产品测量和过程测量相结合以促进在软件系统开发方式方面的改进 • 如何对以下活动开发一个ADPE元素：(1) 量化组织在什么地方是产品状态和什么地方是过程状态，(2) 量化对这个基线评估的差异，(3) 建立定量的产品和过程目标，(4) 量化实现这些目标的进展，(5) 在测量活动基础上定义综合过程和产品改进的方法 • 拟订通过建立SEE去精化或变更软件系统开发过程的程序，那么如何预计和管理与这样的程序携手并进的文化变更 • 组织文化变更的作用是编写ADPE元素，并实施和不断改进它们 • 对ADPE的实施，卖方项目层的人必须适应那些管理他们工作的ADPE实践，那么应如何处理由于这些人为适应而引入的对ADPE实施的挑战 • 对ADPE的实施，客户中的某些人负责指导卖方项目经理完成项目工作，那么应如何处理由于这些人的这项工作而引入的对ADPE实施的挑战 • ADPE实施造成的对客户高级管理者的影响 • 卖方高级管理者在实施ADPE完成软件系统开发的文化变更方面所起的关键作用 • 业务压力如何影响卖方高级管理者对实施ADPE的支持，如何减轻这些压力 • 软件系统开发过程的“处方式应用”如何与授权相关联，并如何导致文化变更 • 客户在实施ADPE中的作用 • 培训在完成与实施ADPE相关的文化变更方面的作用 • 如何将宣传ADPE实施当作个人事业发展的机会 • 组织因素与文化变更所需时间的关系 • 为什么定义ADPE元素的开发和更新过程的ADPE元素紧密地与组织文化变更结合在一起 • 如何开发一个ADPE元素，它定义ADPE元素的开发和更新过程
8	过程改进规划——对如何编写SEE实施计划提供实用指南。这个计划作为做前几章所讨论的事的路线图	• 如何编写SEE实施计划，它的完成能导致全组织软件过程的改进 • 和以下19个关键的SEE实施问题有关的因素，如何在SEE实施计划中处理这些问题： 1. SEE实施任务及其分阶段执行的时间线考虑是什么？ 2. 应怎样分阶段执行ADPE要素？ 3. SEE中应包括哪些ADPE要素？ 4. 应如何构成ADPE——(1) 由少量的要素（即近于10个），每个由几十页组成，或(2) 从大量要素（即几十个以上），每个由几页组成，或(3) 由(1)和(2)的某种组合？ 5. ADPE要素应该以怎样的频率更新？

(续)

章	标题和目的	学习内容
		6. 在单个ADPE要素中应该包括多少细节数量? 7. 如何为组织定义一个应用开发技术环境 (ADTE) 计划? 8. 如何包装ADPE要素及相关项? 9. 如果组织规模较小的话, 应如何处理ADPE实施? 10. 严格的SEE实现方法是什么? 11. 应如何加强辅导和训练, 以促进ADPE实践的实施? 12. 能采用什么策略去满足由SEE实施提出的文化变更难题? 13. 在实现ADPE实践的过程中, 如何处理金钱至上的商业现实? 14. 如何考虑组织成员在适应工程环境的自觉性方面存在很大的差异这一事实? 15. 在组织中应由谁来开发SEE? 16. 如何制定SEE实施方针? 17. 如何在项目级规划ADPE实施改进? 18. 如何将过程和产品测量集成到组织的过程中? 19. 应如何构造一个SEE实现计划?
附录A	如何测量战略 信息管理 (SIM) ——使你 深入了解目标 测量在软件上 下文之外的应用	• 什么是战略信息管理 (SIM) • 如何用SIM测量组织绩效的改进 • 为什么SIM的量化是有价值的

本书的主要特点

本书的以下各项特征, 可以帮助读者在自己的环境下实现成功的软件开发:

- 在承诺写下所希望的业务工作“方式”时, 带注释的提纲有助于读者克服不知所措的毛病。
- 200多幅图帮助读者迅速融汇贯通本书的思想及其相互关系。
- 每章开头的关键思想清单帮助读者以适合自己的方式阅读该章, 并帮助组织你要从中吸取的东西。
- 易于调整以适应你的环境的过程图, 使得: (1) 如果你是卖方, 那么可以建立一致地生产“好的”软件系统的方式, (2) 如果你是客户, 那么可以指导卖方和你一起工作, 以按时、在预算内交付你要的东西。
- 带有充分细节的已完成的业务事例, 使得你能调整该事例说明的概念以适应组织。

- 一种易学和易用的测量技术，称为目标测量，^①可以用它去测量几乎所有的事，包括产品“优度”和过程“优度”。
- 易于融汇贯通的技术，能用于分析组织文化，并确定如何进化组织文化，以期成功实现软件业务。
- 一种简单的强有力的技术，用于（1）评估软件项目风险，（2）生成一个降低风险的项目计划，因而增加项目成功的可能性。
- 详尽处理19个关键过程改进问题，这19个问题解释如何构造一个对组织有意义的过程改进大纲。

在本书的最后，是带注释的文献目录。文献目录中的绝大多数文献得到其作者的同意。这个文献目录的目的是：（1）指出与我们所讨论的专题有关的其他处理方法，（2）帮助你深入理解我们所阐述的专题，（3）帮助你进一步探讨那些我们仅简单提及而你可能有较大兴趣的专题。

我们强调一点，本书独立于任何特定软件系统开发技术。无论你正在使用面向对象技术、统一建模语言（Unified Modeling Language, UML）、自动化工具、原型化还是这些技术与其他技术的某种组合，都会发现本书是有帮助的。

组织如何制度化它的工程和过程原则

为了设置组织的从事软件系统开发工作的“方式”，本书的核心概念是系统工程环境（SEE）。SEE由下列两个互补的组件组成：

- 应用开发过程环境（ADPE）
- 应用开发技术环境（ADTE）

ADPE是一组定义组织业务工作“方式”的方针、指南、规程和标准。我们称这些实体为“ADPE要素”。ADTE是用于开发组织软件产品和软件有关产品的技术集合（例如，个人计算机、网络和自动化工具）。

本书的焦点是ADPE，因为它的元素定义软件系统开发过程。我们告诉你如何收集软件开发的概念。指导如何在这些元素中挑选内容，并将它们融合到组织中。要使组织的软件开发能力成熟起来，这个融合（即制度化）可能是最难的——因为变更基本上是感性的，而非理性的。现在开始介绍第1章。

^① 目标测量，或OM，是Scott E. Donaldson and Stanley G. Siegel的注册商标。

作者简介

本书作者Scott E. Donaldson和Stanley G. Siegel均为Science Applications International Corporation(SAIC)的副总裁。SAIC是全球500强企业之一，也是美国最大的雇员所有制研究和工程公司，领先的IT服务公司，具有40多年的历史，雇员超过4万人，年收入超过60亿美元，其业务遍布全世界，在技术开发和分析、系统开发和集成、技术支持服务、高技术硬件和软件产品等方面具有广泛的经验，其客户包括政府、商业和国际方面，其涉及的市场业务领域包括能源、环境、政府、医疗、技术、信息技术、因特网等，SAIC公司本身以软件过程改进而著称。

Scott E. Donaldson具有24年的软件工程的经验，负责过许多大型的项目或程序，服务的单位有公众的、私人的和商业部门。当前，他是该公司软件工程过程组（Software Engineering Process Group, SEPG）的负责人。他负责将要生产4000多个交付品的近100个订单、技术内容。他还负责形成大纲的关键技术方法，包括对所有的订单进行规划并配置人员。他还负责开发和精化，用于指导客户软件系统开发的方法论和监控质量和绩效度量。

Stanley G. Siegel在软件工程方法论方面是公认的专家，在系统分析和软件工程领域有30多年的经验。他作为演讲者活跃在国际软件产品保证和软件过程改进方面的学术报告会上。作为高级技术客户和指导者，他指导过广泛的项目，其领域包括：软件工程方法论评估、软件需求分析、软件测试和质量保证、对软件方法开发的支持以及技术保证。他目前是公司SEPG的成员，不仅负责某部门的文化变更，而且开发和维护了系统工程环境（SEE），并将SEI的软件CMM中的概念纳入到SEE中。

目 录

译者序

前 言

作者简介

第1章 业务实例.....1

- 1.1 引言.....1
- 1.2 业务实例的关键思想.....6
- 1.3 什么对好的业务工作是有意义的.....8
 - 1.3.1 一致性.....10
 - 1.3.2 投资回报率.....12
 - 1.3.3 信息生产率.....15
 - 1.3.4 管理增值.....16
 - 1.3.5 平衡得分卡.....16
 - 1.3.6 业务的快速步伐.....18
- 1.4 软件系统开发概念.....19
- 1.5 产品“优度”和过程“优度”.....23
- 1.6 必要的软件系统开发范畴.....26
- 1.7 通用四阶段软件系统开发生存周期.....30
- 1.8 软件系统开发中所涉及的用户、
买方和卖方组织.....32
- 1.9 在改进软件系统开发文化上的障碍.....33
- 1.10 候选的软件过程改进方法.....37
- 1.11 预览本书其余部分.....41

第2章 项目规划过程.....45

- 2.1 引言.....45
- 2.2 项目规划的关键思想.....46
- 2.3 生存周期在项目规划中的作用.....47
- 2.4 理想的、实际的和现实的项目规划.....55
 - 2.4.1 传统的系统工程生存周期范例.....57
 - 2.4.2 原型化生存周期范例.....61
 - 2.4.3 信息工程生存周期范例.....63
 - 2.4.4 项目规划视图.....66

- 2.5 风险评估和项目规划.....68

- 2.6 项目规划过程.....73

- 2.7 项目计划内容.....79

- 2.8 项目规划总结.....82

第3章 软件开发过程.....87

- 3.1 引言.....87
- 3.2 软件系统开发过程的关键思想.....90
- 3.3 软件系统开发过程概述.....92
- 3.4 客户.....95
- 3.5 卖方过程工程组.....100
- 3.6 客户/卖方开发团队和变更
控制委员会.....102
 - 3.6.1 客户项目经理.....102
 - 3.6.2 卖方开发团队.....103
 - 3.6.3 变更控制委员会.....119
- 3.7 卖方高级管理者.....120
- 3.8 软件系统开发过程总结.....121

第4章 变更控制过程.....127

- 4.1 引言.....127
- 4.2 变更控制的关键思想.....130
- 4.3 有计划的和无计划的变更.....132
- 4.4 变更的处理.....137
 - 4.4.1 变更控制过程活动——冻结
评审启动器.....141
 - 4.4.2 变更控制过程活动——审计
评审启动器.....142
 - 4.4.3 变更控制过程活动——分析
评审启动器.....142
 - 4.4.4 有计划变更的变更控制过程
例子——对详细设计规格说明
草稿的变更控制.....144
 - 4.4.5 无计划变更的变更控制过程