



HZ BOOKS

PEARSON

开发人员专业技术丛书

持续实施成功软件开发的系统化方法

Successful Software Development
Second Edition

成功的软件开发

(原书第2版)

(美) Scott E. Donaldson 著
Stanley G. Siegel

刘列励 仲田 等译



机械工业出版社
China Machine Press

开发人员专业技术丛书

Successful Software Development
Second Edition

成功的软件开发

(原书第2版)

(美) Scott E. Donaldson 著
Stanley G. Siegel

刘列励 仲田 等译



机械工业出版社
China Machine Press

本书以案例学习的方式讲述了软件开发全过程中涉及的一系列问题,以及持续一致地实施成功软件开发的系统化方法。本书从以下几个方面探讨了软件开发与管理的技术:项目规划过程、软件系统开发过程、变更控制过程、产品与过程的评审、软件度量等,还包含大量生动的图片,可对软件开发人员提供有益的帮助。

本书适合作为软件工程以及相关专业学生的教材,也适合软件项目经理、软件开发人员参考。

Simplified Chinese edition copyright © 2010 by Pearson Education Asia Limited and China Machine Press.

Original English language title: *Successful Software Development, Second Edition* (ISBN 978-0-13-700777-6) by Scott E. Donaldson and Stanley G. Siegel, Copyright © 2001.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Prentice Hall PTR, Inc..

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。

封底无防伪标均为盗版

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

本书版权登记号:图字:01-2009-4556

图书在版编目(CIP)数据

成功的软件开发(原书第2版)/(美)多纳尔德森(Donaldson, S. E.)等著;刘列勋等译. —北京:机械工业出版社,2010.1

书名原文:Successful Software Development, Second Edition

ISBN 978-7-111-29423-8

I. 成… II. ①多… ②刘… III. 软件开发 IV. TP311.52

中国版本图书馆CIP数据核字(2009)第244523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:周茂辉

北京京北印刷有限公司印刷

2010年1月第1版第1次印刷

186mm×240mm·19.5印张

标准书号:ISBN 978-7-111-29423-8

定价:45.00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

购书热线:(010) 68326294; 88379649; 68995259

投稿热线:(010) 88379604

读者信箱:hzjsj@hzbook.com

译者序

自 20 世纪六七十年代“软件危机”爆发以来，如何进行成功的软件系统开发，一直是软件业界经久不衰的热门话题。无数专家学者倾注心力，无数工程人员大胆探索，他们的努力结出了累累硕果。今天，我们有了更加完善的软件开发方法论，更加精准的技术方法，更加细致的管理规程，更加便捷的开发工具，软件业界的气象为之一新。

但无论如何，这一世纪难题远远没有解决，虽然软件开发效率有了革命性的提高，但软件的规模也变得更加庞大和复杂，即使在 21 世纪的今天，软件项目失败的消息仍不绝于耳，身处软件业界的我们，颇有切肤之痛。软件工程大师早有断言：“软件系统开发没有银弹”，“软件系统本身的复杂性，远远超过其他类型的系统”，“软件开发是一个复杂的系统工程”……

从无数成功和失败经验中，我们发现，面对日趋复杂的软件需求和日益庞大的开发队伍，必须采用系统化的开发管理方法，才能有效降低开发风险，提高成功几率。

“一花独放不是春，百花齐放春满园。”单个软件项目的成功开发，可能只是精英人物的卓越表现，不一定代表整个组织的能力。在全组织范围内、持续一致地达到成功开发，才是软件业界追求的终极目标。

本书是两位作者的心血结晶，他们根据多年公司运营开发管理经验，提出自己的观点。他们认为：成功的软件开发，是由管理、开发、产品保证等多个团队在开发全过程中，持续保持沟通、持续降低风险的一种组织化、系统化的工程实践。成功不是依靠个人，不是依靠单个项目组，而是依靠组织提供的各种管理和保证机制，依靠组织制定的标准规范过程。因此，本书贯穿始终的主线就是“组织化”，将组织的软件系统开发过程“处方式”地应用到具体项目的开发中，其最高境界是以企业文化的形式渗透到每个人的工作行为规范中。

本书主要讨论了以下内容：

1. 成功的软件开发：是指一致地生产优良软件系统的能力。
2. 项目规划过程：介绍项目规划的要点和内容，讨论在项目之初搭建管理、开发和产品保证的组织架构，建立持续沟通、风险评估的计划机制的必要性。
3. 软件系统开发过程：从过程总体框架、客户、过程工程组、变更控制委员会、高层管理等几方面，介绍软件开发过程的要点和内容。
4. 变更控制过程：定义计划内与计划外变更的概念，考察变更控制委员会本身的机制，介绍各种表单的作用和格式。

5. 产品与过程的评审：以变更控制委员会为核心，对产品本身与开发产品的过程进行评审，介绍了评审的关键要点，以及评审过程和表单格式。

6. 度量：以组织熟悉的日常用语为表达方式，对产品和过程进行度量。

本书包含许多生动的图片，这也是其他同类书籍少有的，相信能为读者阅读和理解本书观点提供有益的帮助。

感谢吴畏、汪燕参与本书的翻译。译文如有疏漏之处，欢迎读者不吝赐教。

仲田

2009 年 12 月

前言

千万不要承诺我们不应该做的事，以免被迫去做我们不能做的事。

——亚伯拉罕·林肯 1856 年 5 月 29 日在首届伊利诺伊州共和党
代表大会之前的演讲。摘自《亚伯拉罕·林肯文集》
1905 年版第 2 卷第 249 页，Arthur B. Lapsley 编

成功的软件开发是指“‘一致地’生产‘优良的’软件系统的能力”

客户要求软件系统具备预期的功能，按时交付，费用不超支，并且满足他们明确提出的其他各项标准。卖方要求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标准。满足客户和卖方双方标准的软件系统是“优良的”系统。客户和卖方还要求“一致地”满足他们的标准。软件开发工作不应该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是指“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唯一“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、实践和工具上作出了许多贡献，这种丰富的多样性导致用不同“方式”去“一致地”开发“优良的”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客户和卖方，软件开发工作是以客户和卖方间的关系为中心的。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，本书的读者包括以下人员：

- 开发软件产品和软件相关产品的人员。
- 直接管理以上人员的管理者。
- 管理上述管理者的人。
- 购买或使用上述工作生成的产品的用户。
- 培训上述人员的人。

不少人已经从本书的第 1 版学到了所需的技能。客户使用本书优化完善了与卖方打交道的方式。卖方使用本书建立或优化完善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，使他们能更好地理解其企业当前的产品。客户和卖方使用本书去培训人力资源部的人员，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沟通的机制，交流沟通中说明：1) 你需要什么；2) 何时需要；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沟通机制：1) 与客户交流沟通时对客户需求的理解；2) 在项目团队成员中进行交流沟通时，应怎样给客户他所要的东西。

本书与其他软件开发书籍的区别

要实现成功软件开发有很多因素，其中，最为突出的是：有效交流沟通、风险降低和组织化的成功软件开发“方式”，这些是贯穿全书的线索。

有效交流沟通是指“传达信息、思想或感情，使之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沟通造成很大风险。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。”因此，对我们而言：

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

风险降低是指“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) 无法不超过预算地交付；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是有风险的。可是，许多人并不评估风险，也不安排合适的资源去缓解风险，监控风险并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，卖方作出响应说：“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。

组织化的成功软件开发“方式”是指“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，正如我们前面所说的，构造软件系统不止一种方式。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做的工作，直接开始编码吧！”卖方说：“我们知道用户需要什么，所以我们开始吧！”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：1) 增进整个软件系统开发期间的有效交流沟通；2) 持续降低风险。

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。我们介绍一种基于语言的度量技术，用于评价软件过程及其生产的产品。我们解释如何使用这样的度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。我们解释为什么本书讨论的观点效果很好，在如何有效应用这些观点方面提供一些建议，并使你深入了解哪些方式可能无效。

组织化的工作“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。

如果组织化的“方式”包括了这样的经验教训，已知的和预期的风险就降低了，但不一定能全部消除。并且，客户/卖方的有效交流沟通减少了误解，因而降低了软件产品不满足准则的风险。

因此，本书介绍用于有效交流沟通和降低风险的技术。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，已经在实际的软件系统开发项目中成功地实施了这些技术。这些项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6章，图P-1显示了每一章的标题和目的。具体而言，这6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”工作方式的业务案例，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沟通的主要词汇。

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，规划只是组织化的软件系统开发“方式”的一部分。

第3章 陈述组织化的软件系统开发“方式”（或过程），即组织化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”？本书所陈述的组织化的业务工作“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，它们特别处理了有效交流沟通和降低风险问题。本书所陈述的组织化“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，以及度量过程。我们定义和解释角色、职责、活动和沟通途径。我们提出的这个开发软件系统的“方式”，供你在确定自己的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持续进行的降低风险的实践。在第3章中，我们提出一种软件系统的开发“方式”供你考虑，对此关键原则，我们强调如下推论：

在某种特定情况下，如果你决定不遵循组织化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。不论计划得多好，一旦项目开始进行，事情就会变化，这几乎是一个公理。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沟通问题。有时，客户自以为已经明确地告知开发者他的需求了；有时，开发者自以为已经了解客户需求了；但在开发者实现需求之后，客户和开发者的分歧可能会很大。本章提供一些指导，供你在自己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、产品保证等系统专业范畴。第5章从这三个系统开发专业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作出明智的、有事实依据的决策。

第6章 为度量而度量只能浪费时间和资源。本章提供实用的指导，介绍怎样用目标读者同意和了解的日常用语来表述有意义的度量。有意义的度量将对以下几项作出贡献：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。

目 的	
第1章 业务案例	介绍建立一致的软件系统开发方式的业务案例，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	采用根据项目风险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	对于组织化的、能够导向成功的软件系统开发过程，定义其组成原则，并通过定义一个顶层过程来说明这些原则，你可以使用该顶层过程形成自己环境中的过程框架。
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沟通问题。
第5章 产品和过程评审	描述与第3章所述各种评审相关的基本过程，它是降低软件系统开发风险而获得成功的一种方法。
第6章 度量	在度量产品“优良度”和生成产品的软件系统开发过程的“优良度”方面提供实用指南。本章主要集中于如何使用度量以实现一致的产品和过程的“优良度”——即实现成功软件系统开发。

图 P-1 本书的组织结构图，为“怎样才能实现成功软件系统开发”提供实际而可信的答案

表 P-1 用更详细的条目说明每章中将学到的内容。

表 P-1 本书各章重点

章	标题和目的	学习内容
1	业务案例——1) 介绍建立一致的软件系统开发方式的业务案例；2) 介绍本书其余部分所需的基本概念	- 成功的软件开发是指什么 - 为什么在软件过程改进上投资以一致地获得“优良的”产品能产生良好的业务价值 - 对工作方式的优化完善或转换，首先且主要是一种文化变革实践 - 成功的软件开发必定是一个客户/卖方合作的过程，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或用户 - 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（无法解脱地）依赖个人英雄主义完成任务 - 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化的工作方式有利于好的业务 “处方式运用组织化的工作方式”的含义是什么，以及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“优良度”、软件过程改进、生命周期、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/卖方交流沟通是成功的软件开发的关键要素 - 实现有效客户/卖方交流沟通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 - 获得成功所必备的软件系统开发专业范畴——管理、开发、产品保证 - 实现文化变革的障碍 - 为了实现软件开发成功，要做的远远不止如下工作：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，埋头开发。 - 优化完善与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

(续)

章	标题和目的	学习内容
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	- 项目计划是一个实时更新的联结客户 / 卖方合作关系的合同, 它规定卖方的管理层、开发人员和产品质保人员要完成的工作和客户管理层要批准的工作 - 生命周期在项目规划中的作用 - 规划是在买方 / 用户和卖方间不断进行的协商 - 怎样解读在整个项目生命周期内管理、开发和产品保证这三个必要专业范畴间的相互作用 - 如何规划变更 - 比较不同的开发工作完成方式——理想的、实际的和现实的, 以及它们对项目规划的影响 - 如何构造一个简单而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降低风险 - 怎样构造降低风险的项目计划 - 如何开发定义组织项目规划过程的 ADPE 元素
3	软件系统开发过程—— 1) 定义一种构成过程框架的原则, 它能在整个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	- 软件开发业务中可能产生的合同协议 - 如何编写“优良的”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并确定相关的职责以完成项目规划工作 - 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, 它在整个过程中明显地包括客户还包括任何产品开发生命周期 - 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 - 如何处理某些环境中的过程问题, 在这些环境中, 大量软件系统开发项目或多或少以并行方式进行 - 如何将生命周期与软件系统开发过程相结合 - 在定义组织化的软件系统开发过程时, 要考虑两个主要方面, 即其细节层次和组织范围 - 如何将软件系统开发过程纳入系统开发过程中 - 设计一个表单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 ADPE 元素, 用以定义组织化软件系统开发过程 - 为什么这个元素是着手建立 ADPE 的良好出发点
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	- 为什么不良沟通困扰着所有软件系统开发项目 - 为什么客户和卖方对产品状态会有截然不同的看法, 为了降低出现不同看法的可能性, 需要做什么 - 怎样管理计划外的变更以及计划内的变更 - 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: - 你要什么新的或不同的东西吗? - 有什么事出错了吗? - 我们应该将产品基线化吗? - CCB 的机制 (例如, 在 CCB 会议上记录些什么, 通过项目生命周期时 CCB 的作用, 谁应该是 CCB 主席, CCB 的投票机制应该是什么, CCB 章程里包含什么, 怎样执行 CCB 会议, CCB 的成员应多长时间开碰头会等) - 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, 以及应怎样设置它 - 如何设计变更控制表单格, 使其有利于组织的工作 - 如何制定用来定义软件系统开发过程中 CCB 工作的 ADPE 元素

章	标题和目的	学习内容
5	产品和过程评审 ——描述与第3章所述各种评审相关的基本过程，它是降低软件系统开发风险，从而获得成功的一种方法	• 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 • 第三方产品保证组织所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，从而把软件系统开发过程导向成功 • 什么是软件审计，以及它与评审的关系 • 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 • 高层管理者对于可视性的需要，以及评审怎样有助于满足这些需要 • 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产生损害 • 一些技术编辑建议，用于指导构造组织化的技术编辑 • 如何制定用于阐述：1) 第三方产品保证；2) 同行评审；3) 验收的 ADPE 元素
6	度量 ——在度量产品“优良度”和生成产品的软件系统开发过程的“优良度”方面提供实用指南。本章主要集中于如何使用度量以实现一致的产品和过程的“优良度”	• 了解何时度量才有利于改进软件系统开发过程 • 如何避免“为度量而度量”的活动 • 如何使用易学易用的目标度量技术，可以用它去度量几乎所有的事情，包括产品和过程的“优良度” • 如何建立产品和过程度量含义的测定基准 • 如何度量客户满意度 • 如何量化产品完整性的概念，用以评估软件产品的“优良度” • 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“优良度” • 如何使用产品“优良度”度量去跟踪产品通过软件系统开发过程的演进，以发现产品开发的问题 • 如何建立在任何环境中度量产品和过程“优良度”的取值规模 • 如何度量第3章所描述的过程“优良度” • 如何将目标度量运用到其他软件业务改进方法中，如软件工程研究所（SEI）开发的那些方法（CMM / CMMI） • 除采用目标度量开发的那些度量指标之外，如何定义其他的产品度量指标和过程度量指标 • 如何将度量集成到软件系统开发过程中 • 如何将产品度量和过程度量相结合，以促进在软件系统开发方式方面的改进 • 如何开发一个 ADPE 元素，用来：1) 量化组织在什么地方是产品优化且过程优化的；2) 量化与这个基线评估的差距；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程度；5) 在度量活动基础上定义综合过程和产品改进的方法

本书的主要特点

本书具有以下特点，可以帮助读者在自己的环境下实现成功的软件开发：

- 带注释的提纲，帮助你在准备编写所希望的业务工作“方式”时，克服“头脑一片空白无从下手”综合症。
- 150 多幅图，帮助读者迅速融会贯通本书的思想及其相互关系。
- 每章开头的关键点清单，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。
- 一组过程图，它们适应性强，易于调整，可以使得：1) 如果你是卖方，那么可以建立一致地生产“优良的”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，以按时按点、不超预算地交付你要的东西。

- 已完成的业务示例，细节充分，你稍加调整即可把相关概念应用于自己的组织。
- 一种易学易用的度量技术，称为**目标度量**（Object Measurement, OM），可以用它去度量几乎所有的事，包括产品“优良度”和过程“优良度”。

我们强调一点，本书独立于任何特定的软件系统开发技术。无论你正在使用面向对象技术、统一建模语言（UML）、自动化工具、原型法，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。

怎样在组织中应用本书的工程和过程原则

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 由下面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，因为它的元素定义了软件系统开发过程。我们告诉你如何理解软件开发的观念，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——因为变革基本上是感性的，而非理性的。

作者简介

本书作者 Scott E. Donaldson 和 Stanley G. Siegel 均为美国科学应用国际公司 (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。目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 个订单和技术内容。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。他还负责开发和完善用于指导客户软件系统开发的方法论、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 SEPG 公司的成员，不仅负责某部门的文化变革，而且开发和维护系统工程环境 (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 业务案例	1
1.1 引言	1
1.2 业务案例的关键要点	6
1.3 是什么造就了良好的业务价值	8
1.4 软件系统开发的概念	17
1.5 产品“优良度”与过程“优良度”	21
1.6 软件系统开发的必备专业范畴	23
1.7 通用的四阶段软件系统开发生命周期	27
1.8 软件系统开发涉及的用户、买方和卖方	29
1.9 软件系统开发文化改进面临的障碍	30
1.10 其他软件过程改进方法	35
1.11 本书后续内容预览	39
第2章 项目规划过程	42
2.1 引言	42
2.2 项目规划的关键要点	43
2.3 参与项目规划的生命周期角色	44
2.4 理想的、真实的和现实的项目规划	52
2.5 风险评估和项目规划	63
2.6 项目规划过程	69
2.7 项目计划内容	76
2.8 项目规划总结	79
第3章 软件系统开发过程	83
3.1 引言	83
3.2 软件系统开发过程的关键要点	86
3.3 软件系统开发过程概述	88
3.4 客户	91
3.5 卖方过程工程组	96
3.6 客户/卖方开发团队和变更控制委员会 (CCB)	98
3.7 卖方高层管理者	116
3.8 软件系统开发过程总结	117
第4章 变更控制过程	123
4.1 引言	123
4.2 变更控制过程的关键要点	126
4.3 计划内和计划外的变更	128
4.4 变更的处理	133
4.5 考察变更控制委员会	145
4.6 变更控制委员会的书面工作支持	159
4.7 变更控制过程总结	183
第5章 产品与过程的评审	191
5.1 引言	191
5.2 产品与过程评审的关键要点	193
5.3 产品与过程评审分类	195
5.4 用于软件审计的组合评审	217
5.5 产品与过程评审总结	275
第6章 度量	282
6.1 引言	282
6.2 度量的关键要点	290
6.3 度量总结	292

第1章 业务案例

方法对预防业务故障来说大有帮助：因为它使任务变得容易，能克服混乱，能节约大量的时间，并指导那些与业务有关的人。

——William Penn (1644—1718)，摘自其作品《静心独处的反思与谏言》第249页第403条（1903年版，1976年重印）

1.1 引言

软件业界开发软件系统和产品的“方式”需要改变。这个断言并非只是我们的观点，软件业界许多人都持有此看法。

下面列举的一些针锋相对的观点反映出人们对这种改变的需要：

A：“对于我要的变更，请按时交付，不要超预算！”

B：“请别在最后一刻要求增加新的功能！”

A：“请让软件做我要求的事！”

B：“请帮我完成它——说出你真正要什么。”

A：“请别再对长期问题做临时处理。”

B：“请别再将交付日期提前。”

A：“每月发布六个版本是不可接受的。”

B：“请提供给我能做充分测试的资源。”

A：“请减少功能重叠系统的数目。”

B：“请建立一种能让我们顺利使用的体系结构。”

A：“请降低我们组织对个人的依赖性。”

B：“请给他们提供额外的专业发展机会。”

A：“请用较低的生命周期成本开发新系统。”

B：“请提供实施新业务方式的资源。”

上述例子列举了软件开发业务中常见的交锋，这种交锋既会发生在客户和开发者之间，也会发生在一个软件组织的内部。

本书是基于经验对软件业界要求改变的回应。也就是说，本书提供“如何做”的指南，指导组织优化完善或转变其开发软件系统和产品的方式。

客户要求软件系统能满足某些准则。这些准则一般包括：要求系统满足客户需求，按时交付，按协商一致的费用交付。同样，卖方（即构建软件系统的公司）要求其开发的系统满足某些准则，一般来说，这些准则包括：开发的系统满足客户需求，提前或按时交付，产生合理的收益。我们认为，“优良的”软件系统是能满足客户和卖方双方准则的系统。

我们还认为，客户和卖方要求软件开发业务始终如一地满足其准则。软件开发业务的成功不能只是一次性的。软件业务不应该像玩彩票。对我们来说，成功软件开发意味着“一致地生产‘优良的’软件系统的能力”。

成功的软件开发有如下含义：

- 不存在某种唯一的开发软件系统的“方式”。
- 在某种程度上，实现组织上的一致性是一次计划内的文化变革过程，从而实现所期望的工程行为。
- 在软件开发中，人是最重要的因素。
- 在软件开发中，客户或卖方的关系是合作关系，双方要积极参与，并对需要做什么达成一致。
- 客户与卖方之间的有效沟通能增加满足双方一致认同的准则的可能性。
- 软件开发是持续进行的降低风险的实践。
- 人员的职业技能需要提升，他们的经验和知识应被纳入到组织的软件开发方式中。
- 要以一种对特定软件开发项目有意义的方式使组织的运作过程文档化。
- 度量是实现改进软件系统开发的手段。
- 为了使度量有意义，必须用组织的日常用语表述它；否则度量可能无价值可言。

本书将阐述上述所有的因素。

从总体上看，本书是一本整理有序的经验教训汇编，其涉及的内容包括：

- 在整个软件系统开发工作中进行有效的沟通。
- 在整个软件系统开发工作中降低风险。
- 建立一种组织的业务工作方式，它使组织在所有工作中保持一致性。

将上述三个因素综合起来如下所示：

建立一种组织化的业务工作方式“方式”，促进组织内工作实现一致性，意味着在全组织范围内建立有效的交流沟通和降低风险的机制。

因此，如图 1-1 所示，本书提出转变软件系统开发业务方式的思想，将不良定义的业务工作方式有计划地转变成妥善定义的业务工作方式。不良定义的业务工作方式是指“一种基于个人的工作方式，它在组织范围内未被很好地理解，也尚未文档化。”妥善定义的业务工作方式是指“一种基于组织的工作方式，它在组织范围内已得到很好的理解，也已文档化。”

在许多组织中，人们生产软件的过程是杂乱无章的。事实上，经常是必须依靠从事该工作的人员的个人英雄行为才能生产出软件。正如图 1-1 中所指出的，Sam、Pam 和 Ham 每人都有自己完成工作的方式。随着合作时间的增加，他们无疑会学到彼此如何衔接（晚上和周末长时间相聚确实令人感到十分亲密）。Sam、Pam 和 Ham 无疑非常能干，否则他们早就被替换掉。但是从组织的角度看，Sam、Pam 和 Ham 的工作方式是不良定义的。这就是说，组织不清楚如何向外部人员解释组织怎样工作。例如，组织向一个潜在的新客户解释说：“Sam、Pam 和 Ham 将和你一起工作，构建你的软件系统”。这样的解释基本不能令潜在的客户相信组织有能力生产高质量的产品。这个解释也没有说明组织如何开发软件系统。

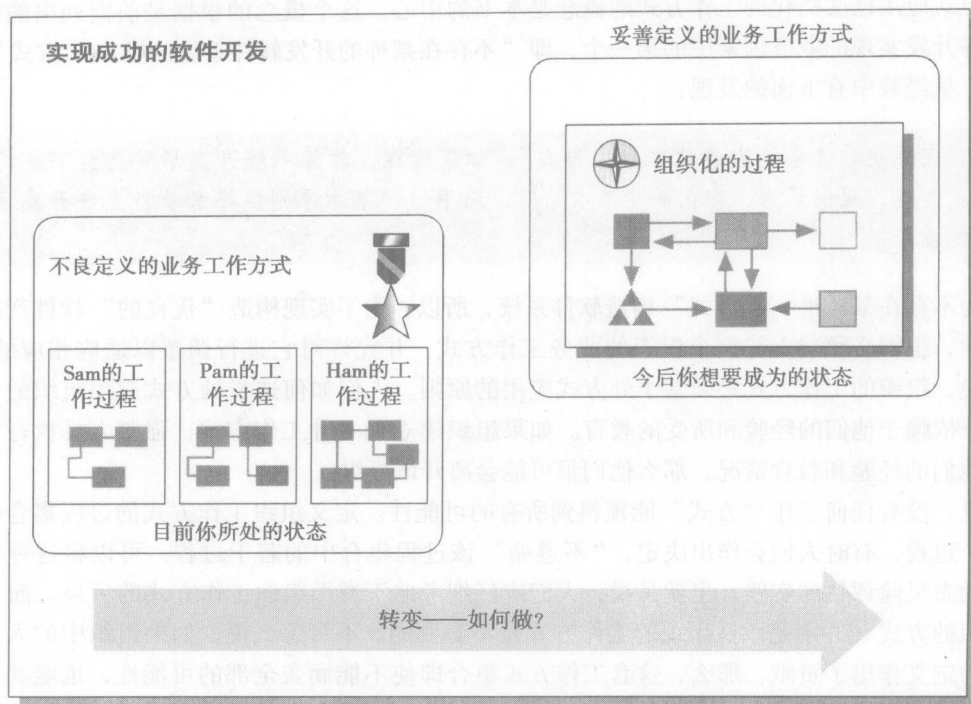


图 1-1 本书给出如何转变到妥善定义的软件系统开发实践的思想，以便能实现成功的软件开发

软件开发是客户 / 卖方的合作过程。因此，一旦一个潜在的新客户成为客户，那么以前的“外部人员”现在成为软件系统开发过程的一部分。举个例子，假如你是客户，如果某天 Sam 无法上班，而只有他知道如何做任务中能影响进度的关键部分，你会不关心吗？或者，如果在任务进行到一半时，Sam 和 Pam 结婚了，并离开公司，而无人知道需要做什么才能完成你的任务，你会不关心其后果吗？

在本书中，我们解释了如何改进客户 / 卖方合作开发软件系统的方式：从个人英雄式的方式，到将个人英雄的实践混合到组织化的工作方式。这种组织化的方式在图 1-1 中标记为：妥善定义的业务工作方式。这种妥善定义的“方式”由下列各项组成：软件的开发方法论、过程、技术、实践和人们用以开发软件系统的工具。本书解释了如何使业务工作的组织方式实施到位，以稳定一个组织。我们解释过，检查机制和平衡机制能让工作保持在正确的轨道上（即它们培育一致性），妥善定义的工作方式将它们纳入到组织化的方式中。而且，这种组织化的工作方式能够提高专业人员的能动性，减少人员流动。由于不存在唯一的开发软件系统的方式，因此本书可以指导读者如何以对组织有意义的方式来实现稳定性。

“转变到一种妥善定义的业务工作方式”的意思是什么？简短的回答是，组织需要在企业文化中纳入已被充分理解的、文档化的工作方式（即制度化），然后人们能“处方式”地把它运用到具体的软件系统开发项目中。对于处方式地运用，我们的意思是：“在可用的时间和资源条件下，应用已文档化的工作方式。”