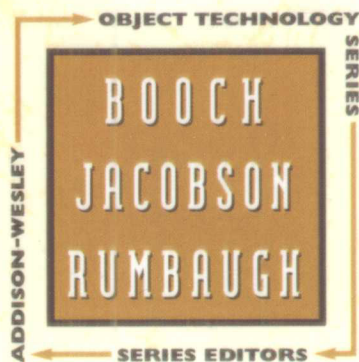


Managing Iterative Software Development Projects

迭代软件开发 项目管理

(美) Kurt Bittner 著
Ian Spence

罗景文 罗灿锋 张弘毅 译
陈庆春 审校



迭代软件开发项目管理

(美) Kurt Bittner 著
Ian Spence
罗景文 罗灿锋 译
张弘毅

清华大学出版社

北 京

Authorized translation from the English language edition, entitled *Managing Iterative Software Development Projects*, 978-0-321-26889-1 by Kurt Bittner, Ian Spence, published by Pearson Education, Inc, publishing as Addison-Wesley, Copyright © 2007.

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.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and TSINGHUA UNIVERSITY PRESS Copyright © 2010.

北京市版权局著作权合同登记号 图字: 01-2009-0129

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

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

迭代软件开发项目管理/(美)毕特纳(Bittner, K.), (美)思朋斯(Spence, I.)著; 罗景文 等译.

—北京: 清华大学出版社, 2010.3

书名原文: *Managing Iterative Software Development Projects*

ISBN 978-7-302-20952-2

I.迭… II.①毕… ②思… ③罗… III.软件开发—项目管理 IV.TP311.52

中国版本图书馆 CIP 数据核字(2009)第 163758 号

责任编辑: 王 军 于 平

装帧设计: 孔祥丰

责任校对: 成凤进

责任印制: 孟凡玉

出版发行: 清华大学出版社

<http://www.tup.com.cn>

社 总 机: 010-62770175

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

地 址: 北京清华大学学研大厦 A 座

邮 编: 100084

邮 购: 010-62786544

印 刷 者: 北京密云胶印厂

装 订 者: 三河市溧源装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 23.5 字 数: 572 千字

版 次: 2010 年 3 月第 1 版 印 次: 2010 年 3 月第 1 次印刷

印 数: 1~4000

定 价: 49.00 元

产品编号: 027684-01

译者简介

罗景文 从 1998 年至今专注于 UML、统一软件开发过程，以及 Rational 系列产品。对面向对象技术、软件工程以及交互式设计有深入研究。曾任职于 IBM developerWorks 中国网站、Ivar Jacobson 博士在中国开办的雅各布森软件（北京）有限公司，及 CSDN 《程序员》杂志。邮箱 misdreaders@luojingwen.cn，欢迎来信讨论相关问题。

罗灿锋 多年专注于 JavaEE 领域开发，热衷于开源技术研究，曾担任项目经理职位，熟悉 UP 开发流程和敏捷开发，有实际的迭代式项目管理经验。现在主要致力于应用开源技术为企业 IT 化平台提供解决方案。邮箱 jsf.com@gmail.com，欢迎一起讨论迭代管理和基于 JavaEE 平台的软件架构设计的话题。

张弘毅 多年来专注于需求开发与管理、架构设计与改进、迭代项目管理与控制方面的工程实践与研究。曾经为霍尼维尔技术研发中心、新加坡科技、华为、中兴通讯、中国工商银行、中国建设银行、中国银行等提供软件工程的咨询服务。曾任职于雅各布森国际。邮箱 zhangjun_edwin@hotmail.com，欢迎来信讨论软件工程的相关问题。

审校人简介

陈庆春，超过 12 年软件开发和咨询经验的软件工程专家，目前专注于需求、敏捷等方向；从 2000 年起，开始参与软件工程的宣传和实施。作为顾问、讲师和项目经理，组织和参与多家知名公司的大规模软件工程变革，涉及软件工程的各个方面，并为超过百家的企业实施相关的服务和培训。对软件开发的全生命周期都很有研究，在用例、构件化开发、迭代开发等方面有专长。现任雅各布森中国咨询总监，曾参与 Rational 中国技术团队、IBM Rational 中国服务团队的组建。

中文版序

Kurt 和 Ian 都是我在 Rational 时的多年老同事，很荣幸有机会为他们这本大作的中文版作序。

迭代的思想其实由来久矣：在 20 世纪 80 年代中期，Barry Boehm 提出了螺旋模型，Ivar 提出了 Objectory 方法(Objectory 是 RUP 的前身)，但是今天往往被划入敏捷的范畴，反而被包装成新的理念。在敏捷的十二条基本原则当中，有四条是直接和迭代相关的。迭代也因此成为敏捷中最重要的实践。可以说，绝大多数项目都应该采用迭代的方式去进行，因此我们 IJI 公司的软件工程咨询项目中全部采用开发方式。

- 我们最优先要做的是通过尽早的、持续的交付有价值的软件来使客户满意
- 即使到了开发的后期，也欢迎改变需求，敏捷过程利用变化来为客户创造竞争优势
- 经常性地交付可以工作的软件，交付的间隔可以从几个星期到几个月，交付的时间间隔越短越好
- 工作的软件是首要的进度度量标准

然而，迭代并不是一件简单的目标，为了有效地迭代，需要解决许多问题。例如，如何迭代地精化需求，如何将需求细化成可迭代的单元，如何规划迭代，如何组织迭代团队，迭代对测试有何新要求，如何在大型团队中推行迭代，等等。

我相信，读者一定能从此书中获得有益的启示！

吴 穹
IJI 中国公司董事总经理

序 言

软件开发经历了一个漫长的发展过程。

“在黑暗中摸索”的年代里，我们墨守着沿袭传统项目管理方法的旧模型，即首先指定需求，然后将其“冻结”，直到结束前都不再改动需求。虽然开发周期十分清晰，即计划工作，然后再按计划工作，但其结果却乱作一团。

问题常变得扑朔迷离，软件开发组织声称成功完成了任务，并演示是根据需求规范交付的。但从商业的角度来看，系统必然是失败的，而且这在交付时会变得显而易见。为什么呢？

因为事情并非是一成不变的。所有大型软件开发项目都会持续相当长的时间，它们势必受到外界环境变化的影响：新技术的涌现、新竞争对手的出现、经济驱动力量的变化等。有时在外界压力的驱使下追求创新，而外界压力并非始终恰好落在软件开发周期范围内。在今天的快节奏世界里，如果您完全照搬最初的需求构建系统，势必会构建出一个错误的东西，导致自己在市场竞争中一败涂地。

虽然也存在执行不当的问题(根本原因可能也是原始需求模棱两可)，但是一个经过无数次讨论总结出的结论是：即使执行过程完美无缺，也可能不会成功，原因是最初的方法足以导致所有管理者一败涂地。显然，我们需要一种新方法，这种方法应该着重于取得商业方面的成功，而不只是软件开发方面的成功。

现在，如果以顺应变化的能力来确定一个软件开发组织是否成功，我们将对这个问题有一个新的看法。变化会在需求中体现出来，因此软件开发组织所面临的最根本问题是处理需求波动的能力。此外，需求变化会随机出现，而且变化范围和出现时间由外界因素决定。也许您的项目节奏和周期井井有条，但需求的变化可能与项目的进展不同步。就像生活中无法加以控制的其他事情一样，您需要采取应对之策。有句话掷地有声：“要么顺应变化，要么被逼上绝路！”

本质上，这正是迭代开发关注的问题。它是软件开发对“交付有价值的软件系统”这一根本商业问题做出的回应。是的，“系统”是一个移动靶。我们放弃了将这个靶子钉死在墙上的不明智想法；我们努力地跟踪它并击中靶心。

迭代开发是一种较新的方法，它从 20 世纪 80 年代开始起步。到了 20 世纪 90 年代，采用该方法的公司发现它比旧方法能够更好地交付价值。事后看来，迭代开发的发展历程似乎一帆风顺，但实际上，迭代方法与其他技术进步一样，经历了一条坎坷不平的发展之路，其中一些方法如昙花般短暂地盛开后迅速凋谢了，有很多方法经试用后终被抛弃。留下的方法是吸收许多项目的经验的结晶，我们要向那些先驱管理者和坚持使用该方法的管理人员致敬，感谢他们给后来者留下了一套可靠的现代软件开发项目管理方法。

历经了数十年的发展，总要停下来反思和总结一番。思想领袖们危襟正坐，苦苦地总结学到的精髓。Walker Royce 于 1998 年编写的《软件项目管理，一个统一的框架》(由机

械工业出版社引进并出版)堪称经典大作。我本人在 *The Software Development Edge* (Addison-Wesley 于 2005 年出版)中发表了自己的见解。现在, Kurt Bittner 和 Ian Spence 合著的此书又给大家带来了新的惊喜。

Kurt 和 Ian 是此领域的行家里手, 他们拥有多年的迭代开发经验。Kurt 曾在 Rational Software 最早的“统一过程”开发组中工作, 并且指导许多团队应用迭代方法。Ian 的技术经历与 Kurt 相似, 他多年来亲自加入到应用迭代开发的项目团队中。两人都工作在一线, 本书呈现的指导内容浓缩了他们的宝贵经验。

他们也是经验老到的作家。两人于 2002 年合著的《用例建模》(由清华大学出版社引进并出版)使一代分析师和开发人员了解到如何合理地指定要求。此书中的方法全面、系统, 论述透彻, 不会给人带来沉重或迂腐之感。这本实用的书籍受到了广大读者的热烈欢迎。当得知他们要撰写这本描述迭代的权威著作时, 我迫不及待地想给他们写书评。

这是帮助读者理解原则并在实践中取得成功的书藉。它不是泛泛而谈的科普类著作, 而是货真价实的权威指南。在此向所有热切期望提升职业素质的软件开发管理者推荐本书。不管您拥有多少经验, 您总能从本书中学到不少东西。最后谨祝各位工作愉快!

Joe Marasco
Ravenflow 公司董事长兼 CEO

前言

“任何作战计划在遭遇敌人后都会失效。”

陆军元帅 Helmuth Von Moltke, 1800-1891¹

如果计划无懈可击，如果一切有条不紊地进行，如果事情的进展始终与计划相符，就没必要执行迭代开发了。之所以使用迭代开发，是因为我们认识到，必须采取一种方法，以便面对变化取得进展，或即使发生变化也能取得进展。迭代开发本质上是一个动态规划和管理过程，在整个项目开发过程中，它会纳入甚至设法获得新信息来管理风险并连续交付增量价值。

迭代开发并非崭新方法，它由来已久，历经若干次的独立演进。它更注重适应环境，通常不注重完善的文档记录，更像是一种“即席”方法。本书将澄清这种认识——将系统地介绍您可以在项目中予以应用的迭代开发管理方法。

迭代方法的核心是将明确管理风险作为项目管理的基本指导原则。它将项目分解成一系列迭代，每个迭代通过实现“场景”减轻一组风险，“场景”必须制衡和控制风险。我们的目标是运用这些概念，并予以落实从而为其赋予生命力。我们要提供一种简明、直观且实用的方法来组织、评估和管理项目并且为项目配备人员，此方法不仅适用于微型项目，也可以扩展到宏大计划中。为此，需要推进一种切合实际的方法，以便按可预测、可重复、井然有序的方式获得结果。

在我们的从业经历中，我们使用 IBM Rational Unified Process (RUP)² 帮助多家公司采用了迭代开发方法，但迭代技术并非仅适用于统一过程³ 项目，实际上，它代表适用于任何迭代软件开发工作的通用方法。在使用极限编程⁴、针对敏捷软件编程的 Microsoft Solution Framework 或其他任何敏捷或迭代软件开发过程时，迭代技术同样行之有效。

本书旨在使任何拥有软件项目的基本管理背景的读者都可以理解迭代开发管理技术。它致力于呈现必需的实用指南信息，以便您开始以可控的、敏捷的、训练有素的方法管理迭代和增量项目。

1 Von Moltke 撰写过多部军事理论著作。他的主要观点是：由于只能规划战事的开局阶段，军事策略是由许多变数组成的系统。他认为军官的主要任务是洞悉所有可能的结果。

2 请参见 <http://www.306.ibm.com/software/rational/>。

3 统一过程的其他变体包括 Essential Unified Process(请参见 <http://www.ivarjacobson.com>)和 Agile Unified Process(请参见 <http://www.ambysoft.com/unifiedprocess/>)。

4 请参见 <http://www.extremeprogramming.org> 或 <http://www.xprogramming.com>。

管理软件项目面临的挑战

管理软件开发项目充满了挑战：适应快速变化的业务和技术环境意味着团队在不断突破极限。由于总是缺少拥有适当技能的人员，交付结果所承受的压力变得更大；另外，技术的演进速度极快。但团队必须对此做出响应，而且必须取得成功，因为这事关重大。如今，商业领域中的所有工作几乎都依赖于软件的使用，商业上的成功往往依赖于一些开发解决方案(由一系列软件构成)的开发团队的成功。

商业需求和实施技术在无休止地、飞快地变化着，这就要求软件开发项目对此做出快速敏捷的响应。软件开发敏捷性是当今和未来能否取得商业成功的决定性因素，对未来的影响尤其重大。

由于软件在支持日常业务运营和赢得竞争优势方面起到的至关重要作用，“软件治理”趋势呈现方兴未艾之势。简单说来，由于商业领域的成功依赖于软件领域的成功，商业决策者需要了解在信息技术和软件开发方面的投资如何换回利润。他们需要可见性与可说明性，如果他们不了解项目状态和发展轨迹，将面临极大风险。另外，在很多环境中，软件对于安全性和业务流程的重要性意味着需要提供审计性(证明完成工作已经批准)。软件开发的治理是值得一提的事项。

最后，软件开发是一项极富创造性和协作性的工作，要取得成功，需要缜密协调来自多个不同学科(经常在不同地理区域和时区工作，有时还生活在不同的文化圈中)的不同人员的工作。发明和创新水准是优秀软件开发的基石，这意味着，开发并非是一个机械呆板的过程(即在开头制定完美计划，然后在执行期间予以监控)。这也导致了软件开发项目遇到令人寒心的因经营规模扩大而增加成本的问题——即大型项目每行代码的成本远高于小型项目⁵——此问题表明软件行业尚未学会如何完好地管理创新性、新颖性和复杂性。

迭代开发使这三个棘手问题迎刃而解，它提供能够满足业务需要的敏捷方法，提供治理开发过程所需的监控能力，同时培育解决复杂业务问题需要的创新能力和协作能力。

学习软件开发的管理艺术

每项工作的开展都离不开出色的管理。三十多年前，Frederick Brooks 写下了这段至今仍如醍醐灌于众生头顶的名言：

“在很多方面，管理大型计算机编程项目与管理所有其他大事业一样，相似的方面比大多数编程人员想象的多。但在其他很多方面，它又与众不同，不同的方面比大多数职业管理人员想象的多⁶。”

这段话一语中的的地点明了人们在考虑软件开发项目管理时容易掉入的两大陷阱。

5 Walker Royce 编写的《软件项目管理，一个统一的框架》(由机械工业出版社引进并出版)一书中对此做了深入讨论。

6 Frederick P. Brooks 《人月神话：软件工程随笔》二十周年纪念版(由清华大学出版社引进并出版)。

- 开发人员可以没有认识到职业管理实践给软件项目带来的价值，觉得项目管理人员的角色类似于 Henry David Thoreau 的作品“论公民的不服从权利”中谈及的政府角色：“管事最少的政府最英明”。
- 管理人员可能不懂技术细节的重要性，也没有意识到无法完全委托这些细节；如果没有真正了解正完成的工作，将无从管理任务项目。

要取得成功，项目管理人员不仅要拥有扎实的传统管理实践基础，还要了解软件开发项目与其他项目类型的差异。项目管理人员只有熟谙工作才能判断工作是否走在正轨上。同时，只懂技术却不晓得如何激励和领导团队的管理人员也不称职——他们未能使团队拧成一股绳。

管理软件开发与管理其他事项类似，只是更庞杂而已。时间压力如此之大，资源如此有限，可能的选项令人眼花缭乱，管理人员的能力需要经受严峻考验。我们发现，迭代开发方法是解决其中很多问题的关键所在。我们还发现，软件开发的成败在很大程度上取决于项目管理艺术⁷。我们无意贬低团队其他成员或他们所遵循的方法的价值，只是强调一点：缺乏良好的管理，项目团队就难以取得成功。

读者对象

本书主要面向负责成功交付解决方案中涉及多个软件的项目的人员。这些人员的职称和职责不尽相同，从日常工作中以不很正规的方式管理小型开发团队的领导者，到负责生产主要软件产品版本的项目经理，再到监督很多协作项目(每个项目对应安排了项目经理)的计划经理。一言蔽之，本书面向必须确定要做什么、何时做和为何做的决策者。

本书将使项目管理团队的所有成员和其他关注项目能否取得整体商业成功的人员受益匪浅。对于面临管理迭代开发项目难题的新项目经理或经验老道的项目经理而言，本书堪称良师益友。

- 对于项目管理新手而言，本书陈述的实用方法提供以可控迭代方式成功管理项目的指南，还提供获取所欠缺技能的学习系统。
- 对于积累了丰富经验却是首次面对迭代软件开发项目的项目管理人员而言，有了本书在手，不仅可以发挥已有管理经验和最佳实践的优势，还能转而使用更加敏捷的迭代和增量软件开发方法。如果已经尝试过迭代方法但仍质疑其好处，那么通览这本具有权威性的书籍，您可以弥补欠缺的信息，成为迭代方法的信徒。
- 对于迭代开发团队的领导者，本书将提供领导团队的方法，使团队能够更高效地支持更广泛的项目管理团队和项目委员会，并与其开展交流。
- 对于项目投资方、计划经理或用户代表而言，本书将帮助了解如何顺应、维持和发掘对授权的软件项目进行迭代开发的好处。
- 对于科目团队(例如需求分析、体系结构等)的领导者而言，可通过本书来了解项目管理团队在确立和维护建立和维持科目团队擅长的迭代开发环境中扮演的角色。

7 从 Standish Group 的问题报告中可以观察到这一点；请参见 <http://www.standish-group.com/>。第 2 章将讨论 Standish 问题报告和项目成功主题。

本书通过描述单个层次化、易用、敏捷、自适应和可伸缩的方法，填平了有关管理迭代开发项目的两个最常见却通常对立的立场之间的鸿沟：以开发人员为中心的方法论传统上倡导的不正规方法与项目管理社区传统上倡导的较正规方法。

我们还扩大了讨论范围，以便纳入有关商业投资的内容。商业投资经常被人忽视，从而损害了项目的商业成效。这是软件开发项目最终与其他投资项目展开竞争的领域，迭代项目必须与商业投资者的观点保持一致。

我们真诚地期望，您能够通过采用这种通透的能够体现出管理意识的方法，将原本协作效果不尽如人意的团队凝聚在一起，确保项目取得成功。

本书结构

本书旨在提供管理迭代项目和计划的实用指南。全书分为两个部分。

第 I 部分介绍管理迭代软件开发项目的基本原理。将浓墨重彩地描述成功管理迭代和增量软件开发项目的原理和最佳做法。迭代开发新手可将第 I 部分作为了解迭代项目的基本管理知识的良好起点；已经熟悉迭代方法的读者也可以从中琢磨出一些新信息。

第 1 章通过研究不同项目角色(如开发人员、用户和项目经理)的迭代开发体验来诠释以迭代和增量方式开发软件的含义。

第 2 章通过分析驱使迭代项目取得成功的要素以及促进成功的重要项目特点来讨论迭代和增量项目的运作方式。

第 3 章将讨论如何控制迭代项目。引入时间盒(迭代开发项目的标准项目生命周期)以及客观衡量结果的概念。本章还介绍了统一过程中的“阶段”概念，并解释此概念如何独立于遵循的迭代开发方法来促进可控迭代开发。

第 4 章将讨论采用迭代开发项目前需要做哪些准备工作。通常必须调整团队结构并转变个人态度，以便营造一个协助和支持采用迭代和增量项目管理实践的环境。

第 II 部分讨论如何规划、管理和评估迭代项目，其中包括如何规划项目和迭代、评估结果及改编计划。最后讨论如何通过改编使迭代方法适用于特小和特大项目与计划，以及如何向新项目团队引荐迭代开发方法。

第 5 章分析层次化项目和规划，这对于执行高效、可控、迭代和增量开发以及提供可伸缩管理解决方案至关重要。本章还引入了“演变”概念，此概念是迭代开发软件产品的重要版本的机制。本章为介绍更具体规划和评估内容的后续章节提供了框架。

第 6 章描述如何创建迭代项目的总路线图。分析用于规划整个项目生命周期和要生成的发布版本集合的技术和机制。本章也介绍了一些简单的生命周期规划原理，并运用它们来阐明创建整个项目规划所涉及的过程。

第 7 章介绍规划和管理单独演变及其阶段的原理。分析演变模式以及如何根据项目工作重点改编模式。本章介绍的模式包括管理风险、评估以及按迭代形式组织工作等。

第 8 章介绍在阶段中规划和管理迭代的原则。分析迭代管理模式以及如何根据项目的工作重点改编这些模式。

第 9 章分析评估过程在多个迭代项目层中的应用方式，特别讨论了迭代和阶段评估的

作用，以及它们如何影响项目的自适应规划并使整个项目得以有效控制。

第 10 章分析如何将先前章节陈述的原理和模式应用于不同规模的项目中，是介绍规划和管理迭代项目准则的收官之作。其中包括如何缩小过程以便管理较小或极小项目，以及如何扩大过程以便管理大型项目和计划。计划管理人员以及所在项目有助于完成更宏大计划的人员将受益匪浅。

第 11 章是收尾章节，给出了如何改用迭代软件开发过程以及如何逐渐采用迭代开发实践的建议。无论是首次采用迭代项目的人员，还是尝试将迭代管理技术引入项目或组织的人员，都可以从本章学到不少知识。

另外还提供 3 个易于理解的附录。

附录 A 简要介绍用例驱动的开发方法，它为附录 C 中及全书其他章节中的示例给出了具体框架。

附录 B 提供本书正文介绍和参考的纲要、模板和清单的可重用的详明版本。

附录 C 更详细地介绍本书正文摘录的示例规划。

目 录

第 I 部分 迭代项目管理原理

第 1 章 什么是迭代开发	3
1.1 迭代与科学方法	3
1.2 迭代的含义	4
1.2.1 迭代是一个独立的微型项目	4
1.2.2 迭代有一组独特的活动	5
1.2.3 每次迭代的结果都在“发布版本”中	6
1.2.4 迭代开发的特点	7
1.3 迭代体验	8
1.3.1 站在核心开发团队的角度分析迭代	8
1.3.2 站在客户的角度分析迭代	12
1.3.3 站在管理团队的角度分析迭代	19
1.4 小结	24
第 2 章 迭代项目的特性	25
2.1 迭代开发：最大化项目成功的机会	25
2.1.1 定义项目成功标准	25
2.1.2 成功和迭代项目	28
2.1.3 成功与迭代：收集项目成功的证据	29
2.2 一个成功的迭代项目的主要特征	30
2.2.1 可验证的、可客观度量的过程	30
2.2.2 避免“特征蠕变”	34
2.2.3 持续增加功能	34
2.2.4 不断提高质量	36
2.2.5 持续降低风险	38
2.2.6 控制变更	39
2.2.7 越来越精确的估算	40
2.2.8 越来越高昂的热情、士气、协作和团队工作	43
2.2.9 致力于一种正确的商业解决方案	44
2.3 小结	45
第 3 章 控制迭代项目	47
3.1 制约项目的变量：范围、质量、时间和成本	48

3.2	利益相关方：项目成功的真正驱动者	49
3.3	控制单个迭代	50
3.3.1	时间盒	50
3.3.2	范围盒	53
3.3.3	控制迭代指南	55
3.4	从总体上控制项目	55
3.4.1	阶段和里程碑的重要性	57
3.4.2	迭代、阶段和里程碑	58
3.5	统一过程阶段	59
3.5.1	初始阶段	59
3.5.2	精化阶段	59
3.5.3	构建阶段	59
3.5.4	移交阶段	60
3.5.5	阶段和里程碑的可选视图	60
3.5.6	对统一过程生命周期的常见误解	63
3.6	客观度量成果：在项目生命周期内控制迭代	65
3.7	度量和迭代：项目的反馈控制	65
3.7.1	度量和阶段	66
3.7.2	通过度量控制项目	71
3.8	小结	71
第4章	为迭代项目管理做好准备	73
4.1	交付价值：成功的关键之处	74
4.1.1	迭代专注于交付价值	74
4.1.2	用例：统一迭代开发方法	75
4.1.3	期望成果、风险、场景和迭代规划	76
4.2	迭代项目的团队建设	78
4.2.1	团队技能和态度	78
4.2.2	领导团队	79
4.2.3	架构的职责：奠定一个坚实的基础	81
4.2.4	与扩展团队一起工作	82
4.2.5	迭代态度和价值观	83
4.3	改变考虑规划的方式	84
4.3.1	传统规划理念	84
4.3.2	为什么将传统的规划理念应用到软件中是错误的	85
4.3.3	渐进规划理念	87
4.3.4	比较两种方法	87
4.3.5	成功的迭代项目经理的七个习惯	89
4.4	小结	90

第 II 部分 规划和管理一个迭代项目

第 5 章 采用分层方法规划和管理迭代项目	93
5.1 管理层次	94
5.1.1 项目群管理层	95
5.1.2 总体项目管理层	95
5.1.3 开发层	96
5.1.4 迭代层	96
5.1.5 各层次的职责	96
5.2 贯穿层次进行规划	97
5.2.1 定位统一过程生命周期	98
5.2.2 计划和里程碑分层	99
5.2.3 定位其他重要的管理工作	104
5.3 分配管理职责	105
5.3.1 重要管理角色	105
5.3.2 分配管理职责	107
5.3.3 以一个综合管理团队的方式工作	109
5.4 层次化管理	109
5.4.1 层次化容忍限度	110
5.4.2 层次化估算	111
5.4.3 监测和控制	113
5.5 小结	116
第 6 章 整体项目规划	119
6.1 计划演变和发布	120
6.1.1 在多个演变之间平衡风险	121
6.1.2 处理连续的演变	122
6.1.3 规划多个演变	123
6.1.4 影响演变数量的因素	124
6.1.5 整体项目计划的组成	125
6.1.6 整体项目计划的形式	126
6.2 生命周期计划的原则	126
6.3 将原理应用于整体项目计划	130
6.3.1 原理 1: 了解期望的结果	131
6.3.2 原理 2: 识别和评估风险	132
6.3.3 原理 3: 确定管理策略	133
6.3.4 原理 4: 创建基于成果的路线图	134
6.3.5 原理 5: 了解解决方案及其作用范围	135
6.3.6 原理 6: 评估和预估要完成的工作	136
6.3.7 原理 7: 针对项目计划的保证约定	137

6.3.8	原理 8: 协调执行计划	140
6.3.9	原理 9: 迭代地演化和质疑各项计划	142
6.4	小结	143
第 7 章	演变和阶段规划	145
7.1	演变中执行的操作	145
7.1.1	在各个开发阶段之间平衡宽度和深度	146
7.1.2	生成的发布版类型因开发阶段而异	147
7.1.3	跨越开发阶段的工作和时间安排	148
7.1.4	迭代持续时间和频率	150
7.1.5	增加迭代和延长开发阶段的推动力	152
7.1.6	遵守时间表	153
7.2	规划演变	154
7.2.1	逐步推进演变计划	154
7.2.2	演变迭代模式	155
7.2.3	逐步演化演变计划	161
7.3	使用规程和工件进行工作	163
7.4	估算和工作分解结构	167
7.4.1	估算工作量	168
7.4.2	配备员工级别和技能集合	169
7.4.3	改写和修订估算及计划	171
7.5	小结	171
第 8 章	迭代规划	173
8.1	认同迭代计划	174
8.1.1	评估项目风险当前的状态	174
8.1.2	认同迭代的范围	176
8.1.3	认同对迭代的评价标准	182
8.1.4	将所有工作汇集成一个简单的计划	184
8.2	计划迭代的执行	185
8.2.1	认同采取的办法	186
8.2.2	定义迭代里程碑	187
8.2.3	认同工作分配	188
8.2.4	认同评估发生的时间	189
8.2.5	将细节作为迭代计划的部分呈献	190
8.3	迭代模式	190
8.3.1	初始阶段的迭代	190
8.3.2	精化阶段的迭代	192
8.3.3	构建阶段的迭代	193
8.3.4	移交阶段的迭代	195

8.3.5 使用迭代规划模式	196
8.4 执行迭代计划	196
8.4.1 规划迭代	197
8.4.2 领导团队	197
8.4.3 保护团队	197
8.4.4 调整计划	198
8.4.5 监控和评估迭代	198
8.5 小结	199
第 9 章 迭代、阶段和项目评估	201
9.1 评估迭代	202
9.1.1 随时进行评估	202
9.1.2 评估过程	204
9.1.3 从不同的角度进行评估	207
9.1.4 规划迭代评估	210
9.2 迭代总结	211
9.2.1 度量与分析	211
9.2.2 进行迭代验收评审并记录迭代的结果	215
9.2.3 常见的迭代问题	217
9.2.4 按照迭代评估结果行事	219
9.3 对阶段进行评估	220
9.3.1 阶段评估过程	221
9.3.2 规划阶段评估	222
9.3.3 评估初始阶段	223
9.3.4 评估精化阶段	224
9.3.5 评估构建阶段	225
9.3.6 评估移交阶段	226
9.3.7 总结阶段	227
9.4 项目评估	228
9.4.1 项目评估过程	228
9.4.2 处理例外	229
9.4.3 提供额外管理控制点	229
9.4.4 项目后评审	229
9.4.5 审查整个项目	230
9.5 小结	231
第 10 章 管理迭代项目的可伸缩方法	233
10.1 管理小型项目	233
10.2 小型项目的含义	235
10.3 扩展项目	236