

规范敏捷交付

企业级敏捷软件交付的方法与实践

Disciplined Agile Delivery

A Practitioner's Guide to Agile Software Delivery in the Enterprise

(加) Scott W. Ambler Mark Lines 著

阎小兵 徐蓓蓓 译



机械工业出版社
China Machine Press

规范敏捷交付

企业级敏捷软件交付的方法与实践

Disciplined Agile Delivery

A Practitioner's Guide to Agile Software Delivery in the Enterprise

(加) Scott W. Ambler Mark Lines 著

阎小兵 徐蓓蓓 译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

规范敏捷交付: 企业级敏捷软件交付的方法与实践 / (加) 安布勒 (Ambler, S. W.), (加) 莱恩斯 (Lines, M.) 著; 阎小兵, 徐蓓蓓译. —北京: 机械工业出版社, 2013.5

(华章程序员书库)

书名原文: Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise

ISBN 978-7-111-42387-4

I. 规… II. ①安… ②莱… ③阎… ④徐… III. 软件开发—项目管理 IV. TP311.52

—中国版本图书馆 CIP 数据核字 (2013) 第 093941 号

版权所有·侵权必究

封底无防伪标均为盗版

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: 01-2012-5602

本书是敏捷软件开发领域的经典著作, 由 IBM 首席敏捷方法专家、世界级敏捷软件开发大师撰写, 系统深入地讲解了规范敏捷交付 (DAD) 的概念和原理, 以及规范敏捷交付过程框架的使用方法和技巧, 为交付大型的企业级敏捷软件提供了行之有效的方法与最佳实践。

本书共 21 章, 分为六部分: 第一部分 (第 1~3 章) 介绍了 DAD 的概念、原理和根基, 以及敏捷与精益开发方法的核心价值与原则; 第二部分 (第 4~5 章) 讲解了 DAD 方法中的角色、权利和责任, 以及如何组建 DAD 团队; 第三部分 (第 6~12 章) 讲解了启动 DAD 项目的步骤和方法; 第四部分 (第 13~17 章) 讲解了以增量的方式构造可利用的解决方案的步骤和方法; 第五部分 (第 18~19 章) 讲述了如何发布解决方案, 第六部分 (第 20~21 章) 介绍了企业环境中的 DAD, 讨论了如何治理 DAD 团队, 以及敏捷开发实践中需要遵守的纪律。

Authorized translation from the English language edition, entitled Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise, 1E, 9780132810135 by Scott W. Ambler; Mark Lines, published by Pearson Education, Inc., as IBM Press, Copyright © 2012.

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 China Machine Press Copyright © 2013.

本书中文简体字版由 Pearson Education (培生教育出版集团) 授权机械工业出版社在中华人民共和国境内 (不包括中国台湾地区和香港、澳门特别行政区) 独家出版发行。未经出版者书面许可, 不得以任何方式抄袭、复制或节录本书中的任何部分。

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

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: 谢晓芳

北京市荣盛彩色印刷有限公司印刷

2013 年 6 月第 1 版第 1 次印刷

186mm×240mm·26 印张

标准书号: ISBN 978-7-111-42387-4

定 价: 79.00 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

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

本书赞誉

“这是一本实用且理论相结合的指南书。它在诠释敏捷价值观和敏捷准则的同时，承认企业业务的现实和未来的愿景。你会发现这里没有纯粹主义者的教条，也没有任何夸张的宣传。Scott和Mark所展示的是，团队如何在纷繁变化的团队环境和企业环境及其诸多限制下，不断前进，最终达到敏捷开发方法的‘甜点’，并收获可持续敏捷实践带来的真正好处。我真希望在10年前就能读到本书！”

——Brad Appleton，大型财富150电信公司敏捷/精益开发的倡导者

“我们发现，我们公司的项目管理办公室（PMO）在定制敏捷项目治理策略时，从规范敏捷交付这一指导方法中获得了极大的帮助。如果团队要使用敏捷交付方法，本书则是必读之书。”

——Larry Shumlich，加拿大太平洋铁路项目经理教练

“本书注定会成为事实上的参考标准，它指导组织如何在复杂的业务环境中使用敏捷方法和Scrum方法。Scott和Mark从成功的敏捷团队中总结出了实用的指导原则和实践经验，告诉企业应该如何实现端到端的敏捷交付生命周期。”

——Elizabeth Woodward，IBM敏捷社区领袖，

《A Practical Guide to Distributed Scrum》一书的合著者

“要获得敏捷性带来的好处，有很多方法，但看到这个务实可行的‘涵盖性’方法，则着实令人鼓舞，它把敏捷实践的大部分好处封装成为一个框架。任何处在不断变化、日益复杂的领域中的人，阅读本书，定能受益。”

——Nick Clare，Ivar Jacobson国际公司敏捷教练及首席顾问

“Scott和Mark完成了一个富有挑战的课题。通过对大量成功实践的深入分析，本书既能帮助锐意进取的敏捷专家加速变革，又能为保守的组织管理者提供可伸缩的方法，并在两者之间找到了一个很好的平衡点。”

——Walker Royce，IBM首席软件经济学家

“规范敏捷交付作为一种基于经验的混合型软件交付方法，反映了实用主义的发展趋势，并远离了困扰软件开发行业40多年的反融合思想。我推崇Scott和Mark所写的这本书，赞

赏他们表现出的领导力，是他们让我们的专业能力上升到了一个新的台阶。”

——Mark Kennaley, Software-Development-Experts.com首席技术官，
《SDLC 3.0: Beyond a Tacit Understanding of Agile》的作者

“我注意到，‘认证式敏捷’在组织中已经泛滥有加，它所带来的严重问题，甚至超过已解决的问题。终于，我们看到这个让人眼前一亮的办法，它告诉我们如何通过规范地使用敏捷方法来缔造成功。谢谢Scott和Mark。”

——Carson Holmes, 第四媒体咨询公司服务交付执行副总裁

中文版序

软件开发人员正在经历一个激动人心的时代，因为他们的工作方式正在发生翻天覆地的变化。伴随2001年《敏捷宣言》的诞生而正式为人们所认识的敏捷软件开发至今已经在各种开发模式中占据了统治地位。然而，没有最好，只有更好，敏捷开发仍然有继续提升的空间。本书向软件开发人员展示了如何更加有效地工作的方法。我们可以停下每隔几年就要重新审视如何工作的脚步，相反，踏踏实实地继续我们伟大的软件开发征程，为组织创建真正为其所需的价值。

2012年6月，IBM出版社出版了本书英文版《Disciplined Agile Delivery》，我们非常高兴地看到，今天本书中文版也与大家见面了。正如书名所示，本书介绍了规范敏捷交付（Disciplined Agile Delivery, DAD）过程框架，它是下一代敏捷软件开发方法。DAD框架描述了敏捷软件开发应该如何在实际中切实执行，包括从项目启动到把可行的解决方案交付到最终用户手中的全过程。它包含了我们在世界各地诸多组织中通过艰辛工作而获得的宝贵经验，这些组织包括世界上最大的软件开发公司之一，IBM。实践证明，DAD框架是一种混合型敏捷开发技术，它展示了Scrum如何通过吸收来自极限编程（XP）、看板（Kanban）、敏捷建模、统一过程以及从外到内开发等方法中的各种实践而得以扩展。DAD框架还展示了这些技术如何适当地融合在一起，帮助组织更快、更有效地采纳敏捷方法。

DAD目标驱动的方法支持DAD成为通往可扩展敏捷的基石，为组织裁剪适合自己具体情形的软件过程提供了清晰而明确的建议。尽管许多敏捷团队属于小型和地理位置集中的团队，但当今更常见的情况是横跨多个地域的大型团队，它们面对着巨大的业务复杂性，或者技术复杂性。DAD过程框架为这些人们描述了克服挑战的各种策略。

自从本书问世以来，我们已经看到在世界各地越来越多的人在采用DAD方法，包括中国。我们曾经在中国开展过多次DAD研讨会，也希望今后有更多的机会与你的组织合作，并提供帮助。我们相信规范敏捷交付策略会在世界范围内的IT组织中成为主导性敏捷开发方法，而阅读本书正是你从一开始就见证和参与的时机。同时，我们还高兴地告知大家，目前规范敏捷协会（Disciplined Agile Consortium）已经推出了一个认证计划，帮助大家进一步获取规范敏捷知识，增加自己对目前或潜在雇主的价值。有关认证信息，可以访问<http://DisciplinedAgileConsortium.org>。该站点预计在今年9月份能从中国访问，我们在此欢迎大家到时能访问该站点，对本书进行评价和讨论。

Scott W. Ambler, Mark Lines

2013年5月

序

软件开发过程之争硝烟已散，赢家是敏捷。当年在Forrester工作时，我们就注意到敏捷方法已经成为主流，大多数的组织认为，他们至少有38%的项目都在使用敏捷开发。但是斯科特和马克指出，在现实中，敏捷的使用与17位思想领袖在2001年描绘出的原本想法还相距甚远，相反，因为组织的行动迟缓、政治因素、人员技能、管理实践、供应商和外包的发展而被大大削弱。我注意到，现实中的敏捷更容易变成瀑布和Scrum混杂（water-scrum-fall-water-scrum）的应用模式。这表明，组织在启动一个项目时，还没有真正摆脱掉传统上冗长的启动过程，即预先定义所有需求，为项目制定详细的规划，甚至还要进行一部分设计。在软件版本发布尚不频繁的情形下，“先Scrum，后瀑布”（Scrum-fall）模式所定义的发布实践策略还可以为大多数组织所采用，但无疑是代价昂贵，实施复杂，并且需要许多手动部署和测试工作。瀑布两头Scrum中间或称为瀑布式Scrum（water-scrum-fall）模式并不坏，它的优点是开发团队可以以迭代和基于Scrum的方式开展工作，但瀑布式Scrum无法真正释放敏捷方法本应产生的能量。企业敏捷不仅建立了最有效的软件开发过程，而且更重要的是能够交付具有更大商业价值的软件。所以我断言，扩展的企业级敏捷开发不仅仅对企业中的软件交付组织至关重要，对企业的业务成功同样至关重要。对瀑布式Scrum模式的改进将大大提升企业的商业价值，并最终激活组织的竞争力。本书为此提供了一个真正的实现框架。

在本书中，斯科特和马克，这两位经验丰富的软件交付变革推动者，为大家详细描述了一个企业级敏捷伸缩框架。他们展示了变革领袖如何释放和放大敏捷开发的能量，使它不仅仅服务于团队，而且能为软件的交付提供全面的价值流。市面上介绍敏捷应用的书籍比比皆是，但它们对企业治理和组织监控有关的问题却常常避而不谈，因为这些问题往往是真正棘手的问题，它们不关注如何改变人们做事的方式，反而乐于深究为什么人们要去做一些犯傻的事情。斯科特和马克没有这么做。他们非常明确地把目光集中在困难重重的可伸缩性主题上，力图为治理模型、人员编制问题和管理策略提供一些切实可行的方法。在这个过程框架中他们融入了精益开发思想，扩展和延伸了精益开发思想的精髓，这样，对于变革领导者来说，他们不仅可以提升自己的产品交付能力，而且能使这种能力直接贡献于企业的商业价值。但需要注意的是，这些问题的解决并不像想象中那么容易，采用这些想法不仅需要敏捷开发的技能，还需要动用其他过程模型、变革性技术和优秀的工程实践。

斯科特和马克不仅启发我思考，还使我重新回忆起许多业已忘却的事情，例如，那些敏捷时尚人士没把握谈及的事情。本书所讲述的不是敏捷的时尚，而是严肃的敏捷变革，对任何变革领导者来说，它都是值得品读的一本书。

Dave West@davidjwest

Tasktop首席产品官，Forrester Research前副总裁和研究总监

前言

在客户眼中，信息技术（IT）行业的声誉着实令人尴尬。几十年来，我们浪费了太多稀缺的预算和资源，违背了自己的承诺，而交付的产品功能却并不是客户的真正之需。旁观者对我们的职业一定困惑不已。我们有那么多过程框架和各种各样的知识体系，以至于连我们自己都很难理解那些层出不穷的、只有首字母缩写的词语，更不用说隐含在其后的丰富的知识和资源。看一下：PMBOK、SWEBOK、BABOK、ITIL[®]、COBIT、RUP、CMMI、TOGAF、DODAF、EUP、UML和BPMN等。即使范围缩小到敏捷这样的社区，也还有Scrum、XP、CI、CD、FDD、AMDD、TDD、BDD等许多缩写语。在这些策略之间存在着相当多的重叠，同时也不乏巨大的差异。我们真的需要一起采取行动做些什么了。

为什么要敏捷

在传统/经典项目中，甚至很悲哀，在采用“厚RUP方法”的项目中，为了各种标准机构的各种标准，基本的业务需求和系统需求的探索经常会以各种时尚的文档作为收尾。尽管在一些监管较严格的环境中，这些做法被证明是良好的实践，但在许多情况下，它更被证明是对时间和精力的巨大浪费，而且它经常极少能为组织带来组织所追求的终极价值，所以组织不得不对过程和方法进行定制，从而满足自身的业务需求。

幸运的是，过去十年间，在软件开发领域涌现出了各种敏捷方法，让我们有机会把自己从上述愚蠢的方式中拯救出来。敏捷开发方法的美妙之处在于，它们关注的是如何为我们的客户交付具有高商业价值的可工作软件，而且这种关注在项目的开发过程中会来得更早、更频繁。我们可以随着客户业务需求的变化，随时调整项目的目标，并在总体上，鼓励最大限度地减少编制文档的工作量，最大限度地减少甚至消灭官僚机制。谁不喜欢呢？

更重要的是，敏捷策略在实践中似乎很奏效。在过去几年里，斯科特在IT行业内做过多次调查^①，并不断地发现，敏捷和迭代软件开发策略的表现一直优于传统和一些临时的开发策略。尽管敏捷方法仍有改进余地，而且本书会为此提出多条改进建议，但很显然，敏捷这一步是在正确方向上迈出的一步。例如，2011年的IT项目成功率调查（IT Project Success Survey）显示，受调查者认为，67%的敏捷项目是成功的（因为满足所有

① 所有调查问卷的题目、源数据（但出于保护隐私的考虑，移去了与个人身份有关的信息），以及总结性文档都可以从www.ambysoft.com/surveys/上免费下载。

的成功标准），27%的敏捷项目是不太成功的（交付了，但没有满足所有的成功标准），只有6%的敏捷项目是失败的。同一调查显示，50%的传统项目是成功的，36%的传统项目是不太成功的，14%的传统项目是失败的。2008年的IT项目成功率调查发现，敏捷项目团队更善于交付高质量的、具有良好投资回报率（ROI）的、满足利益相关者期望的解决方案，而且比传统团队更快交付。但是，这些只是平均值，敏捷成功率可能因具体项目不同而不同，不过总的来说，这些结果仍足以引人注目。我们现在与大家分享这些数据，旨在激励大家认真对待敏捷，但更重要的是，这是为了说明贯穿于全书的一个主题：我们将尽力回避在许多软件过程书籍中过于热情的“宗教”式讨论，相反，我们的讨论面向事实，基于实验和研究证据。因为研究还在进行中，所以仍有一些证据会有漏洞，但尽管如此，我们仍然会远离那些在其他地方还在进行的争论，诸如“我的过程胜过你的过程”之类。

阿利斯泰尔·科伯恩，《敏捷宣言》起草人之一，认为敏捷方法有三个主要的方面：

- 自律。极限编程（XP）中的典型方法
- 自组织。Scrum中的典型方法
- 自我意识。Crystal（水晶）思想中的典型方法

在本书中，规范敏捷交付（Disciplined Agile Delivery, DAD）将会探讨科伯恩所提出的这三个方面。

为什么需要规范敏捷交付

虽然敏捷策略似乎优于传统策略，但它已清楚地向我们表明，钟摆又摆到了另一个极端，即我们从过于形式主义和以文档为中心的这一端，摆到了几乎只关注代码的另一端。公平地讲，敏捷团队确实在规划方面投入了不少的精力，虽然不可能建立详尽的计划；他们确实在建模方面投入了不少的精力，虽然不可能建立所有的模型；他们确实编写了产品文档（如操作手册和系统概述文档），虽然不可能创建完善的规格说明。然而，在敏捷团队中，迭代方法所产生的效果几乎并没有太多的改善。2011年的IT项目成功率调查发现，69%的迭代项目是成功的，25%的迭代项目是不太成功的，6%的迭代项目是失败的，统计结果与敏捷项目的实际情况基本一致。同样，2008年的IT项目成功率调查发现，从统计上看，采用敏捷方法的团队和采用迭代方法的团队在交付质量、交付能力与交付及时性方面都相差无几，只是敏捷团队在ROI上略优于迭代团队。在对敏捷策略与迭代策略进行比较时，我们发现，敏捷开发的现实表现，确实没有辜负大家用在它身上的华丽辞藻，而且敏捷有可能做得更好。

我们的经验表明，“核心”敏捷方法，如Scrum，适合于非常小的项目团队，解决相对简单的问题，它几乎不存在失败的风险或后果。然而，这些方法显然没有充分考虑交付大型企业解决方案的相关风险，作为结果，我们看到组织正在投入大量精力去组合各种来源的技术，创建一些混合型方法。本书所描述的规范敏捷交付（DAD）过程框架也正是一种混合

型方法，它扩展了Scrum，融合了敏捷建模（Agile Modeling, AM）、极限编程（eXtreme Programming, XP）和统一过程（United Process, UP）中已证明有效的策略以及其他一些方法。Scrum专注于软件的构造，而DAD扩展了Scrum的生命周期，致力于从项目启动到为最终用户交付解决方案这一完整的、端到端的交付生命周期^①。DAD过程框架包含了Scrum有意避而不谈的技术实践以及在Scrum和XP中业已消失的建模、文档和治理策略。更重要的是，在许多情况下，DAD会提供建议，帮助组织选择切实可行的替代方案和折中方案，使得组织能够对DAD进行有效裁剪，使之适合自己的特定情形。通过描述什么可行，什么不可行，以及原因是什么，DAD可以帮助组织更好地采用行之有效的策略。

事实上，高调项目的失败在不断增多，暴露出的原因大都与敏捷策略有关。对于大规模敏捷项目来说，如果我们不着手用更严格的方法来补充和完善核心敏捷实践，那么就可能失去敏捷先驱们所创建的、来之不易的兴旺之势。

本书并不是试图改写现有的敏捷思想（这些例子可以从参考资料部分找到），相反，本书旨在成为实用的指南，使大家能够从今天起，就开始践行结构化和规范化敏捷方法，开发大规模的、承载企业关键使命的项目，满足企业的商业需求。

DAD的历史

规范敏捷交付（DAD）过程框架的概念是由斯科特在2007年提出的，他时任IBM[®] Rational[®]首席敏捷和精益方法学家。那时他与世界各地的客户打交道，帮助客户在大规模场景中掌握敏捷技术的应用。在这个过程中，他一次又一次地观察到，组织在采用主流敏捷方法（比如极限编程（XP）和Scrum）时是多么辛苦。与此同时，马克在帮助组织采用和应用敏捷技术实践的过程中，也注意到了同样的问题。在许多情况下，固有的发号施令式文化会阻碍企业采用这些混合技术。此外，尽管许多组织在敏捷试点项目中很成功，但离开这些试点团队，他们则很难较好地应用敏捷策略。其根本原因在于，这些方法普遍没有致力于解决IT部门所面临的范围广泛的问题，更不用说IT部门以外更广泛的组织机构了。所以说，有些地方看起来不太对劲。

于是，我们开始分别着手解决这些问题。横向上，斯科特与更多的组织一起工作，对敏捷团队进行广泛的观察；纵向上，马克深入多个组织，对敏捷团队进行长期指导。2009年斯科特带领IBM Rational团队开发出了DAD过程框架，直到今天，许多相关工作仍在继续，例如，开发DAD课件，编写白皮书，在IBM developerWorks[®]上发表博客^②。

① 完整的系统/产品生命周期会经历最初想法的形成、交付，直到运维和支持整个过程，并且在交付生命周期中会进行很多次迭代。在DAD中，尽管我们会讨论系统生命周期的其他方面如何影响交付生命周期，但重点关注的是交付。

② <https://www.ibm.com/developerworks/mydeveloperworks/blogs/ambler/>

精益方法简述

精益策略为什么对DAD至关重要，这里有多种原因。

- 精益为简化DAD团队的工作方式提出了深刻的见解。
- 精益为可伸缩的DAD处理复杂情况提供了坚实的基础。这个主题在全书都会有所涉及，而且我们打算在未来要出版的书中再做更详细的讨论。
- 精益原则解释了为什么敏捷实践有效，这也是贯穿于全书的一个主题。
- 精益策略中的看板（Kanban）方法为DAD提供了一种先进的应用策略。

那么，为什么我们不取而代之叫规范精益开发（Disciplined Lean Development, DLD）呢？我们有这样的经验，在目前，精益策略的吸引力和效力还很有可能仅体现在一小部分团队中。这个“小”也许是10%~15%——当然是在20%以下——但只有随着时间的推移才会知道具体百分比。我们发现，大多数开发团队更适合于轻量级的、端到端的过程框架，因为这种框架可以全面、明确且综合地建议团队如何完成工作而不拘泥于过程细节。前面已经说过，为实现DAD过程框架目标，我们提出了各种选择，而它们在本质上是非常清晰和简洁的，我们预计，许多团队会不断演进他们的流程，并逐渐从以敏捷为主转变为以精益为主。

DAD是Scrum和RUP这两个极端之间的折中，而且恰到好处，因为Scrum是一个轻量级的过程框架，仅仅关注在交付过程中的一小部分，而RUP是一个涵盖完整交付周期的综合性过程框架。DAD注重敏捷交付的基础，而同时又保持充分的灵活性，企业可以对它进行定制，从而使其适应企业自身的环境。在许多方面，Scrum是在教敏捷专家怎么爬，DAD是在教敏捷专家如何走，而大规模敏捷（agility@scale）和精益方法（如看板）是在教他们如何跑。

本书能提供什么帮助

我们相信，通过阅读本书，读者会在许多方面受益。

- 它描述了一个端到端的敏捷交付周期框架。
- 它描述了常见的敏捷实践，以及这些实践如何融入生命周期，它们之间如何协同工作。
- 它介绍了敏捷团队如何在整个组织范围内以“企业意识”（enterprise aware）的方式有效工作。
- 它使用了一致且合理的术语，并与其他方法所采用的术语进行了映射对比。
- 它解释了各种策略的权衡，并在许多情况下，提供了可选的替代策略。
- 它提供了一个基础，你可以围绕该基础对敏捷策略进行扩展，以满足现实世界中团队所面临的实际情形。

- 它不是为了哗众取宠，而是用确凿的原因阐述这些技术为何奏效。
- 它确实回答了这个问题：“所有这些敏捷技术是如何组合在一起工作的？”

我们来自哪里

我们俩都注意到，组织在采用Scrum时，总会对其进行扩展，所利用的正是XP、敏捷建模和其他方法中的实践，这非常类似DAD的做法，或者，组织会对统一过程进行裁剪，使之成为类似于DAD的方法。不论哪种策略，组织都投入了大量的精力，而这些原本可以轻松避免。有了DAD，我们希望能够帮助团队和组织，避免这类冗长而且要进行反复试验的过程，而同时还能让团队对它进行定制，以满足其独特的需求。

在IBM Rational中，斯科特仍在带领DAD方法不断发展和演进，并利用他的经验，帮助组织理解和采用这些敏捷策略。本书也包含了来自IBM软件集团——一个拥有27 000开发人员的全球化组织——内部的经验教训，以及IBM全球服务部专业人员采用的规范式敏捷（Agile with Discipline, AwD）方法。规范式敏捷是一种建立在快速解决方案交付（Accelerated Solution Delivery, ASD）框架之上的方法。在2009年秋天，IBM Rational针对DAD推出了一个为期三天的“规范敏捷交付”研讨会。这个研讨会于2010年第一季度首次推介给IBM业务合作伙伴，包括UPMentors。所以作为非IBM员工，UPMentors的马克便成为首批有资格开展DAD研讨会的讲师之一。从此以后，马克不断为DAD做出重要贡献，他把自己的见解和经验带给了DAD。

如何阅读本书

大多数人会愿意从头到尾一页一页地读。然而，有三个例外。

- 经验丰富的敏捷从业者可以从第1章开始，了解DAD的概况。接着阅读第4章，理解团队中的角色。然后再读第6章～第19章，其中详细介绍了DAD的工作方式。
- 资深IT经理可以先读第1章，以便在总体上了解DAD是什么，然后跳到第20章，这一章重点描述了敏捷团队的治理策略^①。
- 愿意从DAD实践示例开始的人可以读案例研究章节，它们是：第12章、第17章和第19章。

我们建议组织采纳敏捷方法所推行的、领先的核心敏捷实践，但同时也建议组织有必要选择和补充一些规范的策略和工具，以适应组织和项目的现实需求。

① 警告：在本书中，我们会使用一些“敏捷祖话”，比如治理、管理、建模，甚至是D词（文档）。我们对此表示歉意。

顺便说一句，本书的一部分销售额将捐给囊性纤维化基金会和多伦多儿童医院，谢谢你对这些善举的支持。

规范敏捷交付网站

www.DisciplinedAgileDelivery.com作为一个社区网站，囊括了有关DAD的所有信息。马克和斯科特两人都是版主。在这里还可以下载其他资源，如有关DAD的教育资料、服务供应商清单和支持材料等。我们诚邀任何愿意为DAD做贡献的人来这里写博客。加入我们的讨论吧！

本书中使用的缩略语

AD	Agile Data	敏捷数据
AM	Agile Modeling	敏捷建模
AMDD	Agile Model Driven Development	敏捷模型驱动开发
ASM	Agile Scaling Model	敏捷伸缩模型
ATDD	Acceptance Test Driven Development	验收测试驱动开发
AUP	Agile Unified Process	敏捷统一过程
AwD	Agile with Discipline	规范式敏捷
BABOK	Business Analysis Book Of Knowledge	业务分析知识指南
BDD	Behavior Driven Development	行为驱动开发
BI	Business Intelligence	商务智能
BPMN	Business Process Modeling Notation	业务流程建模标记法
CASE	Computer Aided Software Engineering	计算机辅助软件工程
CD	Continuous Deployment	持续部署
CI	Continuous Integration	持续集成
CM	Configuration Management	配置管理
CMMI	Capability Maturity Model Integrated	集成软件能力成熟度模型
COBIT	Control Objectives for Information and Related Technology	信息及相关技术控制目标
DAD	Disciplined Agile Delivery	规范敏捷交付
DDJ	Dr. Dobb's Journal	《Dr. Dobb's Journal》杂志
DevOps	Development operations	DevOps
DI	Development Intelligence	开发智能
DODAF	Department Of Defense Architecture Framework	美国国防部体系结构框架

DSDM	Dynamic System Development Method	动态系统开发方法
EUP	Enterprise Unified Process	企业统一过程
EVM	Earned Value Management	挣值管理
FDD	Feature Driven Development	特征驱动开发
GQM	Goal Question Metric	GQM方法（目标问题度量）
HR	Human Resources	人力资源
IT	Information Technology	信息技术
ITIL	Information Technology Infrastructure Library	信息技术基础架构库
JIT	Just In Time	即时方式
MDD	Model Driven Development	模型驱动开发
MMR	Minimally Marketable Release	最简市场发布（版）
NFR	Non-Functional Requirement	非功能性需求
NPV	Net Present Value	净现值
OSS	Open Source Software	开放源代码软件
PMBOK	Project Management Book Of Knowledge	项目管理知识体系指南
PMO	Project Management Office	项目管理办公室
ROI	Return On Investment	投资收益率
RRC	Rational Requirements Composer	Rational Requirements Composer
RSA	Rational Software Architect	Rational Software Architect
RTC	Rational Team Concert™	Rational Team Concert
RUP	Rational Unified Process	Rational统一过程
SCM	Software Configuration Management	软件配置管理
SDLC	System Development LifeCycle	系统开发生命周期
SLA	Service Level Agreement	服务水平协议
SWEBOK	SoftWare Engineering Book of Knowledge	软件工程知识指南
TCO	Total Cost of Ownership	总体拥有成本
TDD	Test-Driven Development	测试驱动开发
TFD	Test First Development	测试先行开发
TOGAF	The Open Group Architecture Framework	开放小组架构框架
T&M	Time and material	计时与计料
TVO	Total value of ownership	总体拥有成本价值
UAT	User acceptance testing	用户验收测试
UML	Unified Modeling Language	统一建模语言
UI	User Interface	用户界面
UP	Unified Process	统一过程
UX	User eXperience	用户体验

WIP	Work In Progress	在制品，进行中的工作
XP	eXtreme Programming	极限编程

致谢

我们要感谢以下人士，感谢他们为本书提供反馈：Kevin Aguanno、Brad Appleton、Ned Bader、Joshua Barnes、Peter Bauwens、Robert Boyle、Alan L.Brown、David L.Brown、Murray Cantor、Nick Clare、Steven Crago、Diana Dehm、Jim Densmore、Paul Gorans、Leslie R.Gornig、Tony Grout、Carson Holmes、Julian Holmes、Mark Kennaley、Richard Knaster、Per Kroll、Cherifa Liamani、Christophe Lucas、Bruce MacIsaac、Trevor O.McCarthy、M.K.McHugh、Jean-Louise Marechaux、Evangelos Mavrogiannakis、Brian Merzbach、Berne C.Miller、Mike Perrow、Andy Pittaway、Emily J.Ratliff、Oliver Roehrsheim、Walker Royce、Chris Sibbald、Lauren Schaefer、Paul Sims、Paula Stack、Alban Tsui、Karthikeswari Vijayapandian、Lewis J.White、Elizabeth Woodward和Ming Zhi Xie。

我们也要感谢以下人士，感谢他们在网上论坛分享自己的想法，而且这些想法最后都在书中得以采纳：Eric Jan Malotaux、Bob Marshall、Valentin Tudor Mocanu、Allan Shalloway、Steven Shaw、Horia Slusanschi和Marvin Toll。

推荐阅读

软件工程：实践者的研究方法（原书第7版）

作者：（美）Roger S. Pressman ISBN: 978-7-111-33581-8 定价：79.00元
英文版：978-7-111-31871-2 定价：75.00元

软件工程（原书第9版）

作者：（英）Ian Sommerville ISBN: 978-7-111-33498-9 定价：75.00元

软件工程：面向对象和传统的方法（原书第8版）

作者：（美）Stephen R. Schach ISBN: 978-7-111-36273-9 定价：65.00元

设计模式：可复用面向对象软件的基础

作者：（美）Erich Gamma 等 ISBN: 978-7-111-07575-7 定价：35.00元
英文版：978-7-111-09507-3 定价：38.00元

面向对象软件工程

作者：（美）Stephen R. Schach ISBN: 978-7-111-25502-4 定价：48.00元
英文版：978-7-111-26526-9 定价：49.00元

UML和模式应用（原书第3版）

作者：Craig Larman ISBN: 978-7-111-18682-6 定价：68.00元

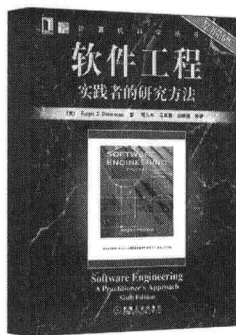
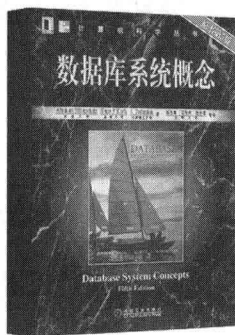
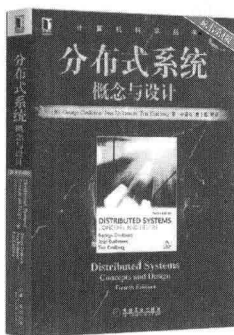
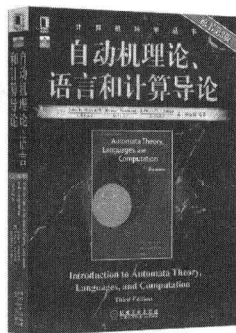
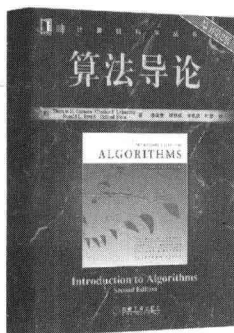
软件测试（原书第2版）

作者：Ron patton ISBN: 978-7-111-18526-9 定价：30.00元

软件测试基础教程

作者：（美）Aditya P. Mathur ISBN: 978-7-111-35188-7 定价：75.00元
英文版：978-7-111-24732-6 定价：49.00元

推荐阅读



算法导论（原书第2版）

2006、2007 CSDN、《程序员》杂志评选的十大IT好书之一 算法中的经典权威之作

编译原理（原书第2版）

编译领域无可替代的经典著作，被广大计算机专业人士誉为“龙书”

自动机理论、语言和计算导论（原书第3版）

1996年图灵奖得主经典巨著升级版

分布式系统：概念与设计（原书第4版）

本书是衡量所有其他分布式系统教材的标准

数据库系统概念（原书第5版）

数据库系统方面的经典教材，被美誉为“帆船书”

软件工程：实践者研究方法（原书第6版）

全球上百所大学和学院采用 最受欢迎的软件工程指南