

Jolt 大奖获得者 Pramod Sadalage 和 IBM Rational 创始人 Scott Ambler 倾情推荐，高价值 DW/BI 系统开发的绝佳实践指南

从架构到管理，从自动化测试到持续集成，借助丰富的实际工作案例，全方位深入讲解敏捷 DW/BI 的关键技术和项目管理实践，有效缓解 DW/BI 项目风险，创新高价值的 DW/BI 解决方案



Agile Analytics

A Value-Driven Approach to Business Intelligence and Data Warehousing

敏捷分析

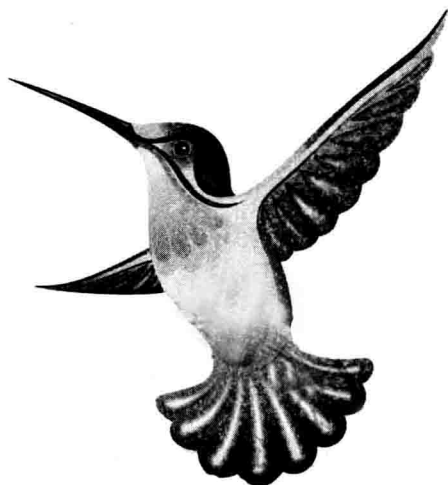
价值驱动的商业智能和 数据仓库系统开发

（美）Ken Collier 著

吴陈欣 杨晓丽 罗旭祥 译



机械工业出版社
China Machine Press



Agile Analytics

A Value-Driven Approach to Business Intelligence and Data Warehousing

敏捷分析

价值驱动的商业智能和
数据仓库系统开发

(美) Ken Collier 著

吴陈欣 杨晓丽 罗旭祥 译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

敏捷分析: 价值驱动的商业智能和数据仓库系统开发 / (美) 科勒尔 (Collier, K.) 著; 吴陈欣, 杨晓丽, 罗旭祥译. —北京: 机械工业出版社, 2014.6

(敏捷开发技术丛书)

书名原文: Agile Analytics: A Value-Driven Approach to Business Intelligence and Data Warehousing

ISBN 978-7-111-46702-1

I. 敏… II. ①科… ②吴… ③杨… ④罗… III. 软件开发 IV. TP311.52

中国版本图书馆 CIP 数据核字 (2014) 第 100464 号

本书版权登记号: 图字: 01-2011-6866

Authorized translation from the English language edition, entitled *Agile Analytics: A Value-Driven Approach to Business Intelligence and Data Warehousing*, 9780321504814 by Ken Collier, published by Pearson Education, Inc., 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 © 2014.

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。

敏捷分析: 价值驱动的商业智能和数据仓库系统开发

[美] Ken Collier 著

出版发行: 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: 关 敏

责任校对: 殷 虹

印 刷: 北京市荣盛彩色印刷有限公司

版 次: 2014 年 6 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 15.5

书 号: ISBN 978-7-111-46702-1

定 价: 69.00 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

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

The Translator's Words 译者序

近些年，互联网经济掀起风暴，大大小小的互联网公司一家家出现并迅猛发展。曾经的“只读式”互联网提供给我们很多的内容和很少的功能，我们只在互联网上阅读信息。那个时代已经远去，而今的 Web 2.0 互联网早已是“可读可写”，我们可以为互联网添加内容，甚至利用互联网来执行任务，实现自己的目标。我们可以在互联网上购买商品、计算房贷利息、预订酒店和机票、搜索问题的答案、分析处理海量数据、下载报表、听音乐、看电影……互联网已完全成为我们生活和工作中不可缺少的工具。随着互联网职能和业务的急速扩充，互联网公司内部的工作方式也发生了巨大变化。传统行业的人员不断地在向互联网转战，互联网从业者的数量达到空前规模。

混迹互联网行业这些年，在实际工作及和同行交流中发现因工作方式导致的不顺比比皆是。拜读 Ken Collier 的书更是深有感触，书中提及的各种负面做法在各互联网公司普遍存在。项目在需求还未明确时就定了上线时间，或是要求一个需 90 天完成的项目在 30 天内完成，只要你在互联网公司外驻足守候，你会发现很多技术人员都在凌晨离开公司。执行团队被要求在项目之初就评估细致的工时，而项目过程中又常发生需求变更，由此导致频繁的返工和加班。执行团队中不同职能的人员互相催赶工作进度，于是需要解释所需的工时。因为互联网常因业绩而追求上线速度，也因为业务量不断攀升，用户通常只涉足前期调研和可用性测试，全程参与项目的情况并不多见。不少管理层在项目全程只简单过问，甚至在项目产出时才第一次提出反馈意见……以上这般种种现象，相信互联网公司的技术同僚们都不陌生吧。

本书中的敏捷项目分析虽然只提到商业智能项目和数据仓库项目，但是其方法广泛适用于互联网中的其他项目，我们推荐所有的互联网从业者（尤其是管理层和执行团队）深入研究并借鉴书中的工作方法。用户故事、用例等这些概念对用户体验从业者来说都不陌生，也希望这些理念能在开发、测试等技术人员当中推广并被接受，这将会让项目大大受益。

感谢机械工业出版社给予我们宝贵的机会来翻译本书，这对我们来说也是一次学习的过程。本书对知识点的讲解十分透彻，引用的实际例子也让我们开拓了思维，看到了国外公司的团队是如何有效运作的。翻译完这本书，我们受益匪浅，希望书中的方法能够惠及各位同行。

最后，虽然我们在翻译过程中慎之又慎并且多次审查，但是书中仍难免存在疏漏和不当之处，恳请读者予以指正。

Forward 序 —

大约在7年前，我的朋友介绍我与 Ken Collier 相识。接下来的日子，我和 Ken 大约每周都一起喝咖啡（我们在亚利桑那州的 Flagstaff 组成了二人敏捷开发小组），我们谈论软件开发，谈论各种各样的敏捷，谈论滑雪、山地骑行，当然也谈论 Ken 的分析项目。早先 Ken 谈到一个停滞不前的项目时，我提及了“敏捷”这一概念，于是 Ken 决定在他的下一个项目中尝试实践敏捷，当时 Ken 冒出这么一句话：“这样看来，以前的项目真是糟透了！”

多年来，我听遍了所有可以想象到的诸如“因为我们是不同的，所以敏捷在公司内部不起作用”之类的话。而 Ken 从没有怀疑过敏捷，一开始他就在想如何让敏捷在商业智能和数据仓库项目上卓有成效，而不是去琢磨、质疑敏捷是否有效。Ken 把每一个障碍都视为契机，并找到一种敏捷方式来攻克障碍。从撰写足以涵盖整个分析软件栈的用户故事，到想出如何在同样的多样栈里进行连续整合，任何时候 Ken 都是那么“敏捷”，就像他时刻在学习运用敏捷一样。今天 Ken 已经成为敏捷（being agile），而不仅仅是执行敏捷（doing agile）。

在接下来的多个分析项目里，有一个项目已运作三年多了，每个季度都会写出报告。Ken 在这个项目里采用基础的敏捷管理方式和开发手段，并且提出创新的方式来将这些付诸实践。商业智能和数据仓库的开发者一直不太愿意使用敏捷（虽然这种现象已经在发生改变），一部分原因是他们不太了解如何在这么大的以数据为中心的项目中执行敏捷。分析项目同样遇到了和很多典型 IT 项目相同的问题——过长的开发周期、过多的费用，更关键的是不能满足客户需求。在这个竞争激烈的商业时代，这样的结果是无法令人接受的。

敏捷分析的一个显著特征是适用范围非常广泛：从产品和待办事宜的管理，到敏捷项目管理技术、自建团队、设计实践革新、自动化测试、管理建设、持续性整合。Ken 的这套敏捷方法适用于广泛的以数据为导向的产品，甚至当你关注的不是分析性项目时，这套方法也是很有用的。

在每一个课题领域，Ken 都是在基本的敏捷实践基础上，根据具体的分析项目对敏捷进

行灵活调整和定制，以适应具体的项目。例如，商业智能和数据仓库团队在配置管理方面远不如软件开发团队；而分析团队在代码管理方面通常不擅长，包括 Java、Ruby 等计算机语言的代码执行、存储步骤程序、SQL 以及特定工具代码。Ken 用几章来回顾软件开发人员一直使用的技术，展示如何让这些技术更好地适应分析环境。Ken 经常问分析团队：“如果你的服务器出了问题，你通常需要花多长时间修复？”他收集到的答案通常是“短的需要几周，长的干脆不修复了”。自动化搭建、整合、测试对于很多分析团队来说都是陌生的，所以 Ken 在版本控制和自动化搭建这两方面各用了一章，阐述如何建立一个快节奏的、连续整合的环境。

本书还专门开辟一章，阐述如何自定义测试驱动的开发流程，以适应分析环境。综合的自动化测试（从单元测试到验收测试）是敏捷开发方法中很重要的一环，也是完整的连续性整合的需要。

Ken 的课题覆盖范围还延伸至架构领域。他倡导架构的革新（设计革新相关内容参见第 6 章），并且描述了具有适应性的架构模式样式。第 6 章介绍了一种有适应能力的分析架构，Ken 在一个大项目里应用了它，这个项目曾经面临的一个主要挑战是项目的不断变更。和传统的“数据推动”不同，这种分析架构倡导“数据拉进”，和 Kanban 系统非常相似。

Ken 的书深得我喜欢的地方可以概括为三点：（1）本书将敏捷的基本原理和操作实践应用于分析项目；（2）本书提出了技术和管理的实践（执行敏捷）和敏捷的核心原则（成为敏捷）；（3）最后，本书涵盖的主题范围之广令人惊讶（从架构到管理），并且都进行了深入的讲述，而不是蜻蜓点水般泛泛而谈。本书无疑是一本优秀著作，不论你正在从事以数据为中心的项目，还是商业分析项目，这本精良的著作都能够让你受益匪浅！

Jim Highsmith

Thoughtworks 有限公司执行顾问

Foreword 序 二

多年前，我带头开发了数据仓库研究所地方协会的网站。在那之前两年时间我确定了规划，并且与很多人一起发展协会，举办会议。

作为这个项目的商业掌舵人，我很了解协会网站需要什么样的功能。我研究了登录系统以及协作系统，并且把这些特性标识到我的特征矩阵中。我已经准备好在三个月内自己建立一个新的系统。

不幸的是，这个项目走向了“合作”。主席分配一个人来管理这个项目，一个 IT 人员来搜集需求，一个营销人来协调整合已经存在的网站。我们确定定期碰面讨论解决方案。而这个项目也随即死亡。

在邮件告诉 IT 开发者我的需求并且和她进行一个简短的对话之后，我阅读了她收集的需求文档。我第一次感觉厄运到来了！当我读这份文档时我都认不出这是我自己的项目了（虽然我很精明）。我知道每个依据这份文档来工作的人（例如外包人员和开发人员）都不能做出接近我们需求的网站。

这个经历让我意识到，商业人士对软件开发中的传统 IT 方法是多么困惑，因为我目睹了 IT 人员如何将商业需求转化为 IT 语言。现在我也明白了为什么如此多的商业智能项目都失败了。

让敏捷来拯救我们吧。第一次读到敏捷开发技术时，我欣喜万分。一些具有商业头脑的人总算融入了 IT 的大家庭。而关于敏捷开发方法论的一切都变得完美。最重要的是，它改变了一个开发项目的权重分布，从 IT 团队转到解决问题的商业运用者上。

虽然敏捷开发方法被认为对经典的事务处理应用软件项目有效，但不幸的是，它在商业智能紧密相连的架构及数据载体开发项目上从来不被看好。

幸运的是，像 Ken Collier 这样的商业智能实践者，应用商业智能敏捷开发方法开拓了新领域，并且愿意传授他们的经验。本书中包含丰富的实战经验，展示了在商业智能中应

用敏捷的注意事项，包括应该做的和不应该做的。

虽然本书包含了丰富的过程知识，但它不是一本教你如何做的手册，而更像是一本在真实商业智能环境中运用敏捷商业智能方法的丰富的故事集。当你读完本书，你仿佛就在与 Ken 并肩工作，可以从这位大师身上学到很多东西。

Wayne Eckerson [⊖]

⊖ Wayne Eckerson 拥有很多身份：TDWI 搜索服务总监，很多深度报道的作者，许多商业以及技术杂志的专栏作家，著名的演讲家，知名博主，以及畅销书《**Performance Dashboards: Measuring, Monitoring, and Managing Your Business**》(John Wiley & Sons, 2005) 的作者。——译者注

当DW/BI项目开始腐坏

大多数数据仓库开发者都经历过不太成功的项目。你甚至可能经历过项目失败的痛苦。几年前，我为一家中型公司工作，公司想用一個正确架构的数据仓库取代现有的本地报告应用。我的项目角色是首席架构师和技术总监。这个项目非常糟糕地结束了，我们的解决方案最终也被遗弃。一开始，项目呈现出成功和用户满意的态势。然而，尽管开发人员、项目经理和利益相关者尽了最大的努力，项目超出预算并超过了计划时间，用户对成果也感到不满。因为这个项目极大地激励我去做到让敏捷原则和实践适应数据仓库和商业智能（DW/BI）开发，所以我会做一个简短回顾，为在本书中展现的敏捷 DW/BI 原则和实践提供依据。它可能和你经历过的项目有类似之处。

关于项目

本节概述了该项目的本质特征，包括下面的内容。

- ❑ **现有的应用程序。**公司现有的报告应用程序在内部叫做“数据仓库”，它明显歪曲了用户对数据仓库应用会提供什么的理解。事实上数据模型是一个遗留操作型数据库的部分复制。这个复制的数据库不包含任何的数据清理，并且被大量自定义的用作生成报告的 Java 代码填充。用户在不同时期都要求增加新的自定义报告功能。应用因为那些很少使用的专业报告功能而变得不堪重负。所有报告都是一成不变的报告。系统没有针对分析活动进行优化，也不提供先进的分析能力。
- ❑ **项目动力。**因为现有的“数据仓库”没有按照数据仓库最佳实践来进行架构，所以它达到了持续满足用户需求需要的可维护性和可扩展性的实际极限。此外，如果一个新的计费系统正要上线，很明显不容易调整现有的系统适应新的数据。因此，可见正确设计数据仓库十分重要。

- **外部驱动。**数据仓库项目是一个销售团队的最初设想，这个团队是全球领先的数据仓库和商业智能软件供应商之一的销售团队。在提供指导和售前支持时，这个销售团队帮助项目赞助商了解项目，并向拥有最好行业实践知识、经验丰富的商业智能咨询师寻求帮助。虽然在销售方面做了很多努力，项目的范围、成本和进度的初步估计，显得过于雄心勃勃。
- **开发团队。**开发团队是外部数据仓库承包商。因为公司现有的 IT 人员有其他优先的工作，公司没有那些具备深厚商业知识或现有操作系统知识的开发人员。然而，开发团队在公司找到了商业和技术专家，从软件供应商那里获得了技术专家。虽然最初的发现工作充满了挑战，但所有的利益相关者都积极参与了。
- **客户。**新数据仓库的主要“客户”是公司财务部门，项目由首席财务官提供资金。他们有一个相对集中的目标——获得更可靠的收入和盈利信息。他们也有大量现成的报告，在常规基础上用于商业分析，为需求分析提供一个合理基础。
- **项目管理。**项目管理（PM）由企业 IT 部门负责，他们使用传统项目管理研究所 / 项目管理知识体系（PMBOK）实践。IT 团队同时参与其他两个大型开发项目，项目都直接或者间接地影响数据仓库范围。
- **托管环境。**因为有限的资源和基础设施，公司 IT 领导最近决定和一个应用服务提供商（ASP）合作，为新开发的生产系统提供托管服务。数据仓库将托管在位于美国西海岸的设施中，而公司总部在东海岸。不能解决的问题是，地理分隔的确对大量数据移动有影响，因为操作型系统仍然在东海岸的企业 IT 基础设施中。

项目成果

最初的项目计划要求初始数据仓库在三个月内完成，但这个发布周期太雄心勃勃了。在项目启动之后，整整 8 个月才完成，晚了 5 个月！用户验收测试并不顺利。用户对项目延迟已经很恼火，当他们终于看到所承诺的功能时，得到的功能和期望的功能有着很大差距。由于项目延迟很常见，所以在努力回到正轨时，一般会增加开发团队的人数。正如 Fred Brooks 说的“在延期项目内增加人员只会让项目更延期”（Brooks 1975）。最终，项目成本远远超过预算，用户不满意，项目被搁置直到下一步计划可以延续开发。

追溯

那么谁是罪魁祸首？每一个人！用户感觉开发者“未击中目标”，没有实现他们所有需求。开发者感觉用户的期望没有被妥善管理，项目范围超出了控制。项目发起人感觉供应商过度承诺、较少实现。供应商感觉这个公司的内部政治和组织问题应该受到谴责。最

后，组织中的多数 IT 人员感觉缺乏所有权，并偷偷地庆祝失败。

项目最终沦为一系列的会议，在会议中进行合同和项目文档的审查，看看谁应该承担责任。你知道吗？涉及的每一个人都应该承担责任。除了数据仓库开发的常规技术挑战外，以下内容被确定为项目失败的根源：

- ❑ 合同没有平衡好范围、时间进度和资源的问题。
- ❑ 需求是不完整的、模糊的和开放的。
- ❑ 批准的需求和设计文档有相互矛盾的解释。
- ❑ 开发者花很多个夜晚和周末试图响应用户更改和新要求，却不断地陷入混乱。
- ❑ 技术团队害怕公开即将到来的失败预警信号，继续努力兑现不切实际的承诺。
- ❑ 开发者没有充分理解用户的需求或者期望，也没有很好地管理需求更改。
- ❑ 现有的知识基于之前的报告应用，使得用户对数据仓库的目的有着明显误解（不是数据仓库的好模型）。
- ❑ 供应商提出了雄心勃勃的承诺，开发者不能在预期的时间内按时交付。
- ❑ 项目经理没有管理用户期望。
- ❑ IT 人员隐瞒了开发者的重要信息。
- ❑ ASP 合作伙伴没有提供开发者期望的连接性水平和技术支持。

事后看来，在项目的最后几天，以下几个事情是显而易见的：

在开发者、用户、利益相关者和内部 IT 专家之间的高度互动，可以确保正确地理解所有参与者的意图。尽早且频繁地交付可用的软件，不论这些软件多么简单，都能大大降低用户的误解，提高他们预期的精度。更加注重用户协作能够避免对需求进行解释时的前后矛盾。关注并响应变化的项目计划比满足一系列“冻结”的合同要求更有利于提高用户对最终产品的满意度。最后，不要去责怪谁，引起这个项目和其他数据仓库项目失败的根源，是开发者和用户之间理解及期望的鸿沟。

关于本书

大约是同一时间，还沉浸在失败痛苦项目的阵痛中，我遇到了 Jim Highsmith，敏捷运动的缔造者之一（《自适应软件开发》、《敏捷软件开发生态系统》和《敏捷项目管理》的作者，本书所属系列的《敏捷软件开发系列》的编辑）。Jim 倾听了我对项目困难的抱怨，传授给我很多经验，我开始思索敏捷方法如何适应 DW/BI 系统开发。不幸的是，我遇到 Jim 时已经太晚了，不能拯救那艘还在下沉的船。从那时起，我和 Jim 成了很好的朋友，几乎每周都边喝咖啡边交换想法。当然，几乎都是他分享很好的点子，我尽可能地吸纳它们。Jim 成了我的敏捷导师，自从我们第一次见面后，我暗下决心在今后的职业生涯中要确保再

也不会遇到失败的 DW/BI 项目。现在看起来这是个很大胆的目标，但是我相信人生苦短，不应该承受那些注定会失败的项目；敏捷开发是一个我们所掌握的最好的项目风险缓解方法；敏捷开发是革新我们可用的高价值、高品质、可用的 DW/BI 系统的最佳方法。这就是本书所讲述的内容：

- 缓解 DW/BI 项目风险
- 创新高价值的 DW/BI 解决方案
- 玩得开心

从上一次痛苦的项目经历之后，我有了很多很好的机会，使敏捷开发方法适应 DW/BI 系统开发的独特特点。和一些非常有才华的敏捷 DW/BI 从业者共事，我成功地适应、实现和细化了一整套项目管理和技术实践，并创造了敏捷分析开发方法。

这种适应是不平凡的，我们要面对很多重要且独特的但是主流软件开发人员不用面对的挑战。DW/BI 开发者需要处理集成了商业软件和编写自定义代码（ETL 脚本、SQL、MDX 及常见应用程序）的混搭组合。DW/BI 开发团队往往拥有广泛且方向不同的技术人员。DW/BI 开发基于大量数据，并基于一个操作型、遗留和专业系统的复杂混合物。DW/BI 系统开发平台经常是高端专用服务器或者服务器集群，复制沙箱开发和测试就会更加困难。因为这些和更多其他原因，敏捷软件开发方法并不总是能够轻松地适应 DW/BI 系统开发，我遇到了一些放弃尝试的 DW/BI 开发者。本书将介绍对敏捷 DW/BI 来说必不可少的关键技术和项目管理实践。每个实践将用一个例子彻底地阐述和论证，我将会告诉你如何修改每个实践，让它更好地适应具体情况。

本书的读者对象是以下三类人员：

- 想学习敏捷技术、学习如何将敏捷技术应用到熟悉且复杂的 DW/BI 开发中的 DW/BI 从业者。我为这些读者提供了涉及商业智能和以数据为中心项目的敏捷技术及项目管理技术细节。
- 那些想知道怎么把熟悉的敏捷实践应用到 DW/BI 系统开发复杂性中的敏捷实践者。为这些读者我阐述了商业智能项目的特点，它们显著不同于软件开发项目的系统，我展示了如何让敏捷准则和实践适应这些独特的特点。
- 有责任监督计划投资组合的 IT 和工程管理者，包括数据仓库、商业智能和分析项目。这类读者既没有深厚的商业智能技术专长，也没有敏捷方法专长。我为这些读者展示了对一个方法的介绍，这个方法可以帮助大家提高项目成功和取悦客户的可能性。

虽然本书不是 DW/BI 系统基础的引子，但是我经常会为了第二类读者，偶尔偏移主题到 DW/BI 基础上。那些熟悉商业智能的读者可以轻松地跳过这些内容。

顺便说一下，虽然我不是所有类型企业 IT 系统的专家，例如企业资源计划（ERP）实

现，但是我有理由相信敏捷分析的原则和实践可以轻松地适应这些类型，在这些环境中很好地工作。如果你是一名 IT 执行者，你可能要思考组织中更广泛的敏捷开发背景。

为什么是一本敏捷DW/BI的书

在过去几年中，敏捷软件开发运动持续爆发。敏捷成功的故事比比皆是。经验证据持续增加，强有力地支持了敏捷软件开发。敏捷社区在过去短短几年中急剧增长，很多大型企业在他们的 IT 和工程部门都采用了敏捷。关于敏捷软件开发各个方面的书籍也在增多。

不幸的是，在数据和商业智能社区，敏捷方法没有得到大面积的普及。由于一些奇怪的原因，数据社区和软件开发社区一直彼此独立地成长和发展。在一个社区中发生的大突破在另一个社区往往不会发生。20 世纪 90 年代面向对象的繁荣就是一个典型的例子。软件开发社区把面向对象融入他们的 DNA 中，已经获得了巨大的收益，但是面向对象的数据库开发仍然是数据社区的非主流。

不论何时，在与 DW/BI 从业者和数据库开发者交谈时，他们常见的反应都是敏捷方法不适用于以数据为中心的系统开发。他们的论据是广泛而多样的，几乎都基于神话、谬论和误解，例如“发展和更改数据模型太耗时了。在开始开发报告和其他用户功能之前，你必须完成物理数据模型。”

现实是，对于以数据为中心的系统，没有任何特殊的地方会让敏捷准则无关或者不当。敏捷实践必须适应挑战，以数据为中心的系统开发必须采用不同的工具集。虽然目前很多关于敏捷概念和技术的书籍直接与数据社区有关，但是它们中的大多数并不直接与数据头脑读者对话。不幸的是，目前的很多敏捷书籍使用最新平台、框架和程序语言，过于狭窄地专注于那些新的、零起点的软件开发。这就很难让读者去推断在这些书中展现的理念，到底是针对数据库开发、数据仓库开发、ERP 实施，还是遗留系统的开发。

敏捷作家和数据库专家 Scott Ambler 撰写了关于敏捷数据库开发和数据库重构（一个敏捷实践）的书籍，以促进数据库社区进行敏捷对话。同样，我写本书邀请 DW/BI 社区进行敏捷运动，因为敏捷是一个在大型复杂 DW/BI 系统方面工作得更好的方式。2008 年 Ralph Hughes 的书《敏捷数据仓库》(Agile Data Warehousing) 上架销售 (Hughes 2008)。Ralph 为 Scrum 和 eXtreme Programming (XP) 技术适应数据仓库的细微差别做出了杰出的贡献，这其中的大多数概念也在本书中得以展现。此外，本书的目标是深入到很多需要以敏捷方式来开发的技术实践中。

敏捷分析——我的用意是什么

我选择“敏捷分析”(Agile Analytics)这个技术术语，除了它能精确地捕捉我的重点外，

更多的是因为它朗朗上口、易于管理。它涵盖了敏捷数据仓库、商业智能和分析包。数据仓库社区使用术语数据仓库指代后端管理和数据分析准备；商业智能指代面向用户的前端应用，从仓库分析展现数据。术语“分析”经常用于表明包含数据定量分析（例如，预测建模、统计分析等）的更先进的商业智能方法。此外，行业术语“商业智能”有时候是一个含糊、包罗万象的术语，包括任何与数据驱动商业流程（商业性能管理、客户关系管理等）或者决策支持（记分卡、仪表盘等）有关的东西。

我使用敏捷分析的称呼并不意味着敏捷方法只适用于某一类面向用户的 BI 应用开发。敏捷方法对数据仓库开发和商业智能以及分析应用开发都是适用和适应的。对于很多人来说，敏捷 BI 开发区域更容易想象，因为人们经常想当然地认为数据仓库已经建好了、填充完了。当然，一个已经存在的数据仓库简化了建立 BI 应用的必要工作。然而，你不应该借此来认为数据仓库必须比 BI 应用优先完成。实际上，敏捷分析是一个用户价值驱动的方法，高价值的 BI 能力驱动支持数据仓库组件的进化发展。这样，我们就能避免过度构建数据仓库去支持超过其预期的目的。

在本书中，我主要专注于 DW/BI 系统的核心——数据仓库。本书中使用的“商业智能”术语或者 BI 术语应包括分析、报告和查询应用。当使用 DW/BI 系统术语时，你应该推断我的意思是核心数据仓库和任何供应仓库的性能应用，例如一个财务仪表板、一个预测门户或者其他一些 BI 应用。然而，DW/BI 的首字母缩写有时候会比较笨重，我偶尔可能会单独使用 BI。大多数情况下，你应该认为我的意思包含 DW。书中也涉及一些先进的 BI 概念，例如数据挖掘和数据可视化。我把具体 BI 项目（例如 CRM 实施）留给读者去推断实践。所有原则仍然适用。

谁应该读本书

敏捷 DW/BI 团队不仅仅由开发者组成。它包含客户（用户）社区（提供需求）；商业利益相关者社区（监视 BI 系统对业务改善的影响）；以及技术社区（开发、部署和支持 DW/BI 系统）。这些社区由一个项目经理、一个业务分析师（或产品负责人）和一个赞助执行人联系在一起。这些社区中的每一个都在项目成功中扮演着至关重要的作用，也都需要一套定义良好的有效敏捷实践。本书将涉及一个或者多个社区的业务和技术读者。

在本书中的一切不是对每个人都有意义，但是有一些内容对每个人都有意义。我和很多组织工作过，他们寻求敏捷训练、辅导和指导。我偶尔会打消敏捷只适于开发者和技术人员的神话。

在一家邀请我去工作的公司里，赞助培训的一个执行者这样说：“如果我们的工程师开始敏捷开发，我们可以更快地完成项目，我们的客户也会更开心。”这句话代表了一些不幸

的误解，可能会让敏捷团队扫兴。

第一，成功的敏捷需要所有团队成员的思维有所改变。客户社区成员必须明白他们的时间被要求用于探索和练习新完成的功能，同时提供持续的写入和反馈。管理团队必须充分考虑项目风险和不确定性，调整他们的期望，团队也要适应不可避免的变化。技术团队必须学会整个包含大量纪律和严密性的新的工作方式。项目界面社区必须致力于日常项目解决，转变他们的角色，给项目成功做出贡献。

第二，敏捷并不总是意味着更快地完成项目。即使是最好的项目团队在完成的工作范围方面也会有局限性。敏捷实践督导团队，尽早传递高价值和最大风险的功能。因此，敏捷 DW/BI 系统能够早点推出是可能的，前提是最关键的功能完成并被接受。然而，我建议不要明显加快项目周期，尤其是在开始时。另一方面，相比传统 DW/BI 开发方法，敏捷能提高质量和客户满意度。

采用敏捷 DW/BI 成功的底线需要所有上述项目社区的成员的认识、理解和承诺。出于这个原因，我尝试把本书设计为提供一些和大家都有关系的内容。

本书是如何组织的

本书分为两个部分。第一部分侧重于敏捷项目管理技术和团队协作，具体包括以下几章。

第 1 章提供了 DW/BI 方法的概述和基线。

第 2 章介绍了一组有效的实践，包括章程、规划、执行和检测一个敏捷分析项目。

第 3 章介绍了一组成立一个高度协作项目团队的指导原则和实践。

第 4 章介绍了故事驱动替代传统需求分析，展现了如何使用案例和用户故事驱使价值持续传递。

第 5 章介绍了一个团队管理和领导的敏捷风格如何有效地替代传统命令控制风格。

第一部分是写给任何涉及敏捷分析项目的人，从赞助执行人、项目经理到商业分析师和产品负责人，再到技术总监和交付团队成员。这几章建立了一套核心实践，可帮助敏捷项目社区开辟一条道路，一起走向成功。

第二部分侧重于必要的、能够持续传递商业价值并有质量保障的技术方法。这一部分包括以下几章。

第 6 章展示了进化设计过程是如何工作的，确保产出高品质的数据模型和最小技术债务的系统组件。

第 7 章介绍了一组自动测试的实践和工具，为了使用一个测试为先的方法，建立数据仓库和商业智能组件。

第 8 章介绍了一系列保持整个 DW/BI 系统在版本控制下和配置管理的技术和工具。

第 9 章展示了如何把测试自动化和版本控制实践组合起来，建立一个持续集成环境，保证不断发展的系统的质量。

第 10 章包括一些剩余的因素和注意事项，它们对成功采用敏捷分析方法至关重要。

我把这部分作为现代开发实践的集合，这些实践用在每个 DW/BI 项目上，做到敏捷或者保持传统（例如，瀑布式开发）。然而，当采用敏捷分析方法时，这些技术实践是必不可少的。这些方法建立了最小化但足够的一组必要的技术实践，以成功地持续、渐进和进化地传递高价值的 DW/BI 系统。

当然，这些技术章节应该由技术团队领导和团队成员阅读。然而，我也建议非技术项目团队成员阅读这几章的每个导言部分。这样做将帮助非技术成员建立对这些实践目的的共识，能欣赏技术团队努力的价值。

本书该怎么读

我认为敏捷分析技术可支持下列要点。

- **敏捷 DW/BI 管理：**一系列致力于帮助你运行项目的实践，包括敏捷前兆、敏捷项目管理方法、敏捷团队、开发者 - 用户界面等。
- **敏捷 DW/BI 技术方法：**一系列实践，致力于开发和传递高价值、高品质、可用的 DW/BI 系统，包括具体的技术实践，例如故事驱动开发、测试驱动开发、构建自动化、代码管理和重构等。

各章被组织成这些主要部分。每一章专门针对一个关键实践或者一组相关实践，以本章要点的高层概述开始，逐渐深入主题。有些章内容足够丰富，足以成为一本书。在这种情况下，我的目标是让读者对这个话题有坚实了解，并为其深入自学提供动力。

如果你想对敏捷 DW/BI 有一个高水平理解，那么阅读每一章开头的初步概述就足够了。在这些概述中我的目标是为每一个敏捷 DW/BI 实践提供一个准确的写照，但是这些部分不讲述所有必要的技术。

如果你是一个数据仓库经理、项目赞助者或者任何需要很好地理解实践而不深入技术细节的人，那么我建议你阅读每一章的中间部分，尤其是项目管理部分。这些部分旨在提供对如何使用这些技术或者理解它们如何用于项目的足够深入的理解。

如果你是一个日常项目团队成员（项目经理、技术团队成员、业务分析师、产品经理等），我建议你阅读每一个项目管理章节（第一部分）的细节和例子。这些旨在给你一组具体的技术，以应用于你的释放规划、迭代集合和所有其他项目管理及用户协作活动。如果你是技术社区的一名成员，第二部分就很适合你。

关于 DW/BI 技术有一句话：和技术无关。我使用很多技术栈来完成 DW/BI 的开发，这