

Software Engineering Series

**DSQM**  
consortium

The cover features a blue background with diagonal white stripes. At the top left, there is a yellow banner with the Chinese text "工程丛书" (Engineering Series). Below it, the English text "Engineering Series" is written in black. In the top right corner, the logo "DS cons" is visible. The title "DSDM Business Focused Development" is repeated multiple times in a light blue font, rotated diagonally across the cover. A large, stylized pink "SE" is centered on the cover. At the bottom left, the text "DS Consortium" is printed. At the bottom center, the author's name "高继荣" (Gao Jirong) is shown next to the Chinese characters "编著". At the bottom right, the translator's name "高继荣" (Gao Jirong) is shown next to the Chinese characters "译".

高继荣 译

# DSDM

## 业务中心框架开发方法

(第二版)

# DSDM Business Focused Development

## Second Edition



電子工業出版社  
Publishing House of Electronics Industry  
<http://www.phei.com.cn>

软件工程丛书

# DSDM

## 业务中心框架开发方法

( 第二版 )

DSDM  
Business Focused Development  
Second Edition

[ 英 ] DSDM Consortium 编著  
Jennifer Stapleton

高继荣 译

电子工业出版社  
Publishing House of Electronics Industry  
北京 · BEIJING

## 内 容 简 介

动态系统开发方法 (DSDM) 是众多敏捷开发方法中的一种, 它倡导以业务为核心, 快速而有效地进行系统开发。实践证明 DSDM 是成功的敏捷开发方法之一。在英国, 由于其在各种规模的软件组织中的成功, 它已成为应用最为广泛的快速应用开发方法。本书主要讲述下列内容: DSDM 如何加快产品的交付, 为什么像 DSDM 这样的敏捷开发方法能够快速体现所开发系统给业务带来的好处, 如何组织用户参与项目以开发出可用的系统, 如何将不同知识背景的人组成一个团队, 如何应对常规的业务约束以进行项目管理。本书提供了在不同规模的软件组织中应用的大量实际案例。

本书适合从事敏捷开发的工程技术人员。

© DSDM Consortium 2003.

This edition of DSDM: Business Focused Development, Second Edition, ISBN: 0321112245 by Jennifer Stapleton, with the DSDM Consortium is published by arrangement with Pearson Education Limited. All rights reserved.  
Simplified Chinese edition published by Publishing House of Electronics Industry. Copyright © 2005.  
Licensed for sale in mainland territory of the People's Republic of China only, excluding Hong Kong.

本书中文简体字翻译版由 Pearson Education Limited 授予电子工业出版社。未经出版者预先书面许可, 不得以任何形式或手段复制或抄袭本书内容。  
此版本仅限在中华人民共和国境内 (不包括香港、澳门特别行政区以及台湾地区) 发行与销售。

版权贸易合同登记号 图字: 01-2003-1083

### 图书在版编目 (CIP) 数据

DSDM 业务中心框架开发方法 (第二版) / (英) 斯特普尔顿 (Stapleton, J.) 等编著; 高继荣译.  
北京: 电子工业出版社, 2005.5

(软件工程丛书)

书名原文: DSDM: Business Focused Development, Second Edition  
ISBN 7-121-01160-3

I. D… II. ①斯… ②高… III. 动态系统 - 软件开发 IV. TP311.52

中国版本图书馆 CIP 数据核字 (2005) 第 042096 号

责任编辑: 赵红燕 特约编辑: 詹文军

印 刷: 北京市天竺颖华印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

经 销: 各地新华书店

开 本: 787 × 980 1/16 印张: 12.25 字数: 254 千字

印 次: 2005 年 5 月第 1 次印刷

定 价: 24.00 元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换; 若书店售缺, 请与本社发行部联系。联系电话: (010) 68279077。质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

# 序 言

自从本书第一版出版到现在，5年过去了。在这段时间里，应用程序的开发出现了巨大的变化。新世纪到来，互联网经历了它的繁荣与萧条，直到2002年全球经济衰落到10年来的最低点。虽然世界在不停地变化，但有一样至今未变：新世纪里的项目依旧失败，其原因与上世纪大同小异，因此，DSDM（动态系统开发方法，也称为业务中心框架开发方法）的继续发展仍然是必需而且重要的。

业务的成功与产品的交付密切相关：在不超出预算的情况下准时交付给客户。这就是创建DSDM的原因：确保在需要时交付出一个合格的业务系统。DSDM强调一定要准时交付满足业务需要的项目，它是一个以用户为中心的项目交付框架。DSDM可以与多种开发方法一同使用，例如极限编程和Prince2。

DSDM联盟于1994年在英国创办，现在在全球范围内有多个社团：英国、北美、比荷卢经济联盟、丹麦、法国和瑞典。DSDM联盟是一个非赢利的会员性质的组织，它的会员多种多样，包括最佳方案提供商、最终用户、独立顾问以及大专院校。有些会员从DSDM建立之初就加入了，有些则是最近才认识到DSDM的好处。

以业务为中心的开发框架DSDM 4.1版发布于2001年。与以往一样，所有DSDM产品和服务都由会员撰写，而且根据会员的反馈进行修改。因为我们的会员一直在使用它来达到商业目的，所以DSDM一直处于改进和变化之中。DSDM的重点和内容会随着市场的改变而改变。DSDM的创立是为了找到一种更好的软件开发方式，今天它已站在了敏捷（Agile）开发活动的前沿，促进了轻量型开发方法学的发展，从而保证了用户的参与并增强对业务变化的响应。在当今生命周期日渐缩短的市场下，DSDM的敏捷性是它至今为止仍然存在的原因。

本书概述了DSDM框架的各个方面。它不能取代联盟会员使用的在线手册。DSDM属于联盟的所有会员，只有会员才能将其用于商业用途。在线手册包括了所有产品、管理工具和技术、开发工具和技术的信息。在任何DSDM实施之前，应该阅读在线手册并充分利用联盟的所有资源。

本书针对于IT类项目中的参与人员，不过这并不排除IT类以外的项目仍然能从中得到有益的帮助。实际上，DSDM致力于用户的参与，它希望任何参与项目的人都可以从DSDM方法中受益。

本书的第一部分是这个框架的概览，以及一些它是如何实施的实例。第二部分是正在运作的 DSDM 案例。这些案例详细讲述了一些成功的例子，以及在使用 DSDM 过程中碰到的问题。当认识到 DSDM 是一种可行的方法后，第三部分会告诉你下一步如何去做，比如从哪里得到更多的信息，如何加入联盟，如何起步等。最后，第四部分包含两个附录。

最后，我要感谢所有参与本书写作的人们。如同 DSDM 一贯的做法，在本书中读到的所有内容都是 DSDM 联盟会员实际经验的总结。特别要感谢 Jennifer Stapleton 为本书所做的工作，就如同她在本书第一版中所做的那样。没有会员全身心的投入，DSDM 联盟将不复存在，框架也将不再合时宜。本框架以用户为中心，由会员来驱动，我们将努力保持这种形式。我希望读者能从本书中获益。同时，鼓励大家访问我们的网站 [www.dsdm.org](http://www.dsdm.org)，通过访问，你会知道如何积极地投身到这个领域中来。

Barry Fazackerley

DSDM 联盟主席

# 前 言

## 背景

为了提高效率和在竞争中占据优势,40年来,商业社会将希望寄托在办公自动化上。在这段时间里,信息技术提供者一直没有能够在不超出预算的情况下准时交付需要的解决方案,或是提供业务需要的功能。

我们花了这么长的时间才明白业务需求的变化非常频繁而且难以定义,对业务流程最为理解的人是流程的日常使用者。我们看到不同的开发项目具有各自不同的生命周期,而且复杂性也不尽相同,不过,我们知道应用程序的开发不是变戏法,它有着一定的结构和规律。

在20世纪的最后20年,为了弄清楚应用程序开发的流程并归纳出避免失败的方法,进行了一系列的尝试。1994年,英国国内来自各个工业领域不同规模企业里的信息系统工作人员,以及来自IT行业中一些大公司的顾问和项目经理聚集在一起,形成了一个非赢利性质的联盟。该联盟专注于理解应用程序开发过程中的最佳实践方法,并对其进行编排以利于在大范围内推广。

其结果就是DSDM (Dynamic Systems Development Method, 动态系统开发方法),一种能够真正满足业务需求的项目交付框架:DSDM项目可以在不超出预算的情况下准时交付,而且不会去掉重要的功能。该框架中的所有资料都来自于会员的实际经验和成功的项目。这种以实践为主的方式使许多人认为该框架“不过是一般常识”,但对于所有正在尝试敏捷开发方法的人而言,这些“一般常识”显然是不一般的。

今天,这个框架在各种项目中得到了广泛使用。在许多国家,无论是小项目、大项目,简单项目、复杂项目,还是IT项目和非IT项目,都可以看到它的身影。联盟一直在修订它的内容。比如,在2001年6月,联盟专门针对电子商务项目发布了一个版本。对这类项目而言,快速交付合格的系统显得更为重要。在2001年夏天,对基本框架进行了一次重要的修改,2002年1月在此基础上做了进一步改进。

假定在建的大多数业务系统与信息技术相关,那么DSDM背后的基本原理也就很简单了:

- 开发活动是一项团队活动。它必须把客户关于业务需求的知识和IT专业技术人员的技术知识结合在一起。
- 高质量要求适用性以及技术上的健壮性。
- 开发活动可以是逐渐递增的——不要求一次交付所有东西,及早交付部分内容比较迟交付所有东西更有价值。
- 受益递减规律适用——应该将资源放在对业务最有价值的功能开发上。

DSDM是关于人而不是工具的。它侧重于真正理解业务需求,交付可行的解决方案——尽快并以尽可能低的成本交付。这个框架不能解决所有项目问题,但对于要在21世纪获得业务所需要的系统而言,它是非常有帮助的。

## DSDM 是什么

DSDM是什么呢,它有哪些不同?虽然它的全称是动态系统开发方法,但它并不被认为是一种方法,而是一种控制框架,重点在于快速交付,补充以如何应用这些控制的指导原则。它定义了一个过程和一系列产物,从这个意义上说,它是一个方法。不过这个过程和这些产物都被有意识地定义在高层次上,由此可以根据不同技术和业务环境对其进行裁剪。对于结构化和面向对象的开发方法,并没有硬性规定某种技术,而是提供一些建议。它之所以不同,是因为它可能是惟一个公共可用的方法,涵盖了从项目的最初提议到项目结束整个过程中的系统开发活动。

DSDM描述了在快速开发以业务为中心的环境中所包含的各个方面——项目管理、预估、原型建立、时光盒法、配置管理、测试、质量保证、角色和职责(包括用户和IT人员)、项目组结构、工具环境、风险管理、可维护性、重用,以及供应商和购买者之间的关系。

该框架的目的是为了快速将产品投放市场:在业务需要时,交付所需要的系统。如果系统是为了满足业务需要的,那么必须达到足够的稳定程度以在实际环境中运行。其次,业务方面的某些紧急要求能够在短时间内得到满足,在以后阶段增加额外的功能。

## 历史背景

IT提供者面临着越来越大的压力,其压力是要求他们更好、更快、更经济地交付系统。在今天这个变化迅速的年代,已经不允许他们花费数年时间完成一个系统:在

系统开发的几年内,业务需求可能早就发生了变化。因此,必须找到不同的方式进行IT系统的开发。现在的技术使开发人员可以更快地交付产品,不过答案不仅仅限于工具的使用上。整个工作过程都需要进行改进。经典的瀑布开发模型不能完全利用现代技术的优势,也难以改变。它已经有大约40年的历史了,本质上是老问题的解决方案——它认为在开始编码之前,如果没有完全理解所要解决的问题,就不会有解决问题的前后一致的方法。

瀑布模型这种严格的顺序开发阶段在今天看来是一种缺点。人们已经进行了多种尝试来改变这一点,包括Barry Boehm的迭代开发方式。它使用一种螺旋模型进行项目计划、风险分析、开发和客户评估。尽管螺旋模型是一种杰出的模型,但它并没有在实际中得到很好的实施。最近几年出现的许多“敏捷”方法证明了需要一种不同的方法。虽然像极限编程这样的方法取得了广泛的认同,但它们没有涵盖项目的所有方面,可能导致软件组织不知道如何将许多现在正在使用的方法进行很好的集成。这也说明了为什么没有多少主要的IT提供商接纳敏捷开发方法。另一方面,可能是他们的客户还没有给他们足够的压力。

在20世纪90年代初,由于James Martin的“Rapid Application Development”一书,IT业认识了快速应用开发。在这本书中给出了许多非常好的主意,但并没有提供整体解决方案。在现在的市场上也有许多工具,不过使用这些工具也就意味着购买了供应商的开发过程。对此,DSDM联盟的创立者认为这阻碍了成功、快速地交付解决方案的发展。

DSDM联盟成立于1994年1月,其目标是建立一个独立于任何工具的、公开的、公认的方法。最初两年Ed Holt是联盟的主席,他曾说过每个购买了RAD工具的软件组织都需要一个新的开发过程。DSDM致力于提供一个过程,这个过程能够在可控的项目环境中,在满足紧迫时间的约束下,建立和维护系统。联盟创立之初有17名会员,他们代表了各式各样的组织:大型IT提供商、小型工具提供商,以及各种规模的用户团体。现在,DSDM联盟已经有上百名会员,在北美、比卢荷经济联盟、瑞典、法国和丹麦都成立了地区性社团。在其他国家,如澳大利亚、印度和中国,对DSDM的兴趣也越来越浓。

在1994年,联盟的技术工作组将开发过程和指导资料(都是基于联盟会员的实际经验和最佳的实践得来的)集合在一起。框架中的一小部分来自于相关领域专家的独到见解,但绝大部分都经过检验——只是从来没有将它们集合到一起,成为一体的开发方法。

框架的第一版于1995年初发布。为了尽快公布,有一些资料在第一版中被省掉了。发布后,建立了一个早期采用者计划(Early Adopter Programme)来监控框架的

使用情况。然后根据这些早期采用者的反馈，再加上省掉的那些资料，组成第二版，于1995年11月发布。此后，又根据更多用户的反馈，在1997年8月发布了第三版。到2001年，在英国政府的白皮书中提到要将DSDM应用到更多的项目中——数据仓库、组件开发和原型业务法，于是又在框架中增加了这些内容。在2000年，人们认识到需要另一个版本专门针对电子商务项目。联盟的技术工作组的工作从未中断过：仍然在不断地搜集用户的反馈信息，并以白皮书的形式提出一些具体需求的解决方法。最近的白皮书覆盖了UML在DSDM项目中的使用情况。

为确保正确理解和应用这个框架，在第一版发布的同时，也提供了一个培训和考试流程。在写作本书时，有超过20 000人接受了认证培训师的培训，越来越多的人通过参加培训和考试流程而成为了合格的DSDM从业者。

## 框架概述

整个框架建立在9条基本原则之上，在本书后面将会对其进行详细讨论。不过，在这里将它们列出来也是非常有用的。前4条原则定义了DSDM的基础，后5条是框架的架构。

1. 用户必须持续参与。
2. 必须授予DSDM团队制定决策的权力。
3. 注重产品的经常交付。
4. 满足业务用途是接受交付品的主要依据。
5. 迭代和增量式开发对得到正确的业务解决方案是必不可少的。
6. 开发过程中的所有变化可逆。
7. 在高层次上制定需求的基线。
8. 测试自始至终贯穿于开发周期之中。
9. 所有项目涉众间的通力合作是不可或缺的。

如果要在指定的时间内提供业务需要的高品质系统，所有这9条基本原则都是必需的。

第5条原则中谈到的迭代和增量过程包括5个开发阶段（有两个是非开发阶段：项目前期——确保项目建立在一个合理的基础上；项目后期——保持交付后系统的运转）。前两个开发阶段是顺序进行的：可行性研究阶段评估DSDM是否适用于要开发的系统，提供成本的初始估计，等等；接下来是业务研究阶段，在此阶段中为项目

以后的工作在业务和技术两方面奠定基础；之后是第一次迭代阶段——功能建模迭代，开始于业务研究阶段中的分析工作在此阶段会进一步细化。本阶段的分析工作是通过系统架构中的功能进行渐进式的原型建模来实现的，系统架构的轮廓也是在业务研究阶段定义的。当很好地理解了一个领域内的功能后，就会在设计编程阶段进行实现，并将达到一定品质的产物交付到实施阶段。实施不仅仅只是在实际环境中建立系统，还包括培训用户。实施阶段结束时，会对本次增量进行复审。从业务上决定是否还有更多的工作需要在接下来的增量中完成。

没有人的参与，任何过程都不可能是完整的。第1条原则声明用户必须在整个开发过程中积极参与：经常提出意见和反馈是至关重要的。DSDM定义了参与人员的角色，包括用户和开发人员。比如，其中一个用户角色是构想师，一般是由于担任这个角色的人的构想（觉得在某个业务领域需要IT的支持）而使得项目得以启动。IT人员中的一个关键角色是技术协调人，这个角色通常由系统架构师担任，他对技术进行把握。结合这两个角色，就可以确保项目在业务和技术两方面的基础是牢靠的。另外，还有很多角色是针对这两个领域的各个专项而定义的。

DSDM的目标是在限定时间内交付系统，这在瀑布模型下是不可能的。由此带来的影响是必须以不同的方式对工作过程进行管理，在这些过程中使用的技术也需要精挑细选，以尽可能地降低损耗。主要的控制手段是时光盒。在DSDM中，时光盒是一段短小的时间段（几天或几周），在这段时间内制造出符合质量要求的产物。这就满足了第3、4、8条原则。以基于产物的观点而不是基于活动的观点，DSDM就能够使得控制注重于制造出的产物而非生产的方法。这也使得在该框架中能够灵活地选择不同的技术。

第6条变化可逆的原则意味着所有制造出的产物都应该进行良好的控制，当发现任何产物有错时，可以回退到一个已知的状态。

DSDM的重点在于满足业务需求，而不是从IT的角度考虑问题。数千个项目业已证明，DSDM的这种以用户为中心、迭代和增量式的方法有许多优点，包括：

- 用户更可能对系统起主导作用。
- 降低了构造错误系统的风险。
- 最终系统更可能满足用户的实际业务需求。
- 用户会得到更好的培训。
- 系统的实施更加顺畅。

## 为什么 DSDM 比瀑布模型快

DSDM 可开发出“工业级”的系统，这种系统能满足用户的需要，并且在很长一段时间内是可扩展、可维护的——不是一次性的、临时的。从业务角度上看，DSDM 开发出的系统可以和用瀑布模型开发出的系统相媲美，但所需的开发时间更短。

这有两个原因。一是做到了更少。向人员传达信息，以及再三地让他们提高速度方面所花的时间更少了。用户之间或开发人员之间移交任务所用的时间也更少了。更重要的是，只开发实际需要的功能。

第二个原因是问题、误解和错误的方向能在早期发现并得以纠正。这就避免了在瀑布项目中经常碰到的大量的后期返工现象。此外，在 DSDM 中所开发的代码是一致的；而在瀑布模型下，到了项目后期，代码经常打补丁，而且与文档不同步。其结果是，DSDM 的代码更易维护。

## 关于本书

如果要更深入地理解框架，必须阅读在线帮助手册。重点是理解框架的内涵，而不是框架在不同环境中使用的素材和评论。DSDM 联盟出版的“DSDM: The Method in Practice”第一版提供的是关于此框架的实践，而不是理论。现在它已经过时了，因为框架本身和框架所应用的环境发生了变化。随着框架的发展，它被用在更多的项目中，从业务过程改变项目到广告，有的项目并不是 DSDM 最初设计的应用对象。本书重点在于应用程序开发项目，不过也引入了一个非 IT 项目的案例。

本书第一部分为框架概览，正如在线帮助手册中所描写的。不过更重要的是，还收录了来自于真实项目中的见闻和信息。第二部分包括一些由项目参与者提供的案例。各案例的作者都描述了他们认为在项目中重要的方方面面。这实际上是一个长短不一、深浅不同的杂文集，不过我们希望读者能够从每个案例中找到有价值的东西。第三部分讲述如何联系联盟，如何成为会员，如何获取联盟的资源。第四部分讲述敏捷宣言的诞生以及未来之路。

## 敏捷软件开发系列丛书

敏捷软件开发系列丛书凸现了一些高效的、轻量级的、充分发挥人的主动性的开发技术，它们的核心基于两个想法：

- 不同的项目需要不同的开发过程或方法学。
- 相对于注重过程本身，注重技术、交流和团体能使项目更敏捷、更高效。

两本书奠定了敏捷软件开发系列丛书的基调：

- “Agile Software Development” (Cockburn, 2002) 描述了敏捷开发在经济学和心理学方面的基本原则。它引入了两个思想：方法学是开发团队认同并采用的一组约定；系统和软件开发被看作为在限定的资源条件下进行创作和交流的合作活动。基于这些观点和原则，实践者可以根据自己的实际情况选择一种敏捷方法。
- “Agile Software Development Ecosystems” (Cockburn, 2002) 描述了在敏捷软件开发宣言 (<http://agilemanifesto.org>) 背后的人，他们所开发的方法学，以及使用敏捷技术的经验。

这套丛书有 3 个分支：

- 提高从事某种具体工作的人的效率的技术。这可能是从事用户界面设计、需求收集、项目计划、设计或测试的人。从事这些工作的人都想知道，这方面的专家是如何工作的。“Writing Effective Use Cases” (Cockburn, 2001)、  
“Configuration Management Principles and Practices” (Hass, 2002) 和 “GUI with Glue” (Hohmann, 写作中) 就是这样的技术书籍。
- 提高一组人的效率的技术。这可能包括团队建设、项目回顾、决策制定或召开高效的会议。“Improving Software Organizations” (Mathiassen, 2001) 和  
“Surviving Object-Oriented Projects” (Cockburn, 1998) 就是这样的书籍。
- 成功方法学案例。谁都想选择一种在与自己所处的环境相似的情况下获得成功的方法学，然后根据自己的实际情况稍做修改。这肯定比从零开始容易得多，也更高效。本书和 “Crystal Clear” (Cockburn, 写作中) 中就有这样的案例。

在互联网上也可以找到关于DSDM和敏捷开发的资源。本书后面的参考文献中列举了一些具体站点和主题。初学者可以从以下一些网站着手：

- [www.DSDM.org](http://www.DSDM.org) 是DSDM联盟的主站点，其中有最新的新闻和其他资源的链接。
- [www.AgileAlliance.org](http://www.AgileAlliance.org) 是非赢利性组织 AgileAlliance 的站点，其中有该组织的活动和很多敏捷开发讨论组的链接。
- [www.Alistair.Cockburn.us/crystal](http://www.Alistair.Cockburn.us/crystal) 收录了越来越多的文章、工作样本和敏捷过程，也有一个敏捷开发的讨论区。

# 目 录

## 第一部分 框 架

|                                           |    |
|-------------------------------------------|----|
| 第 1 章 DSDM 过程概述 .....                     | 2  |
| 1.1 引言 .....                              | 2  |
| 1.2 可行性研究 .....                           | 4  |
| 1.3 业务研究 .....                            | 4  |
| 1.4 功能建模阶段（迭代式）.....                      | 5  |
| 1.5 设计编程阶段 .....                          | 7  |
| 1.6 实施阶段 .....                            | 7  |
| 1.7 项目后期 .....                            | 8  |
| 1.8 要点回顾 .....                            | 8  |
| 第 2 章 基本原则 .....                          | 10 |
| 2.1 原则 1：用户必须持续参与 .....                   | 10 |
| 2.2 原则 2：必须授予 DSDM 团队制定决策的权力 .....        | 11 |
| 2.3 原则 3：注重产品的经常交付 .....                  | 12 |
| 2.4 原则 4：满足业务用途是接受交付品的主要依据 .....          | 12 |
| 2.5 原则 5：迭代和增量式开发对得到正确的业务解决方案是必不可少的 ..... | 13 |
| 2.6 原则 6：开发过程中的所有变化可逆 .....               | 14 |
| 2.7 原则 7：在高层次上制定需求的基线 .....               | 14 |
| 2.8 原则 8：测试自始至终贯穿于开发周期之中 .....            | 14 |
| 2.9 原则 9：所有项目涉众间的通力合作是不可或缺的 .....         | 15 |
| 2.10 要点回顾 .....                           | 15 |
| 第 3 章 实践中的 DSDM .....                     | 17 |
| 3.1 何时使用 DSDM .....                       | 17 |
| 3.2 迭代与增量交付的现实 .....                      | 20 |
| 3.3 分析和设计的技术 .....                        | 20 |
| 3.4 要点回顾 .....                            | 22 |

|                             |           |
|-----------------------------|-----------|
| <b>第4章 时间与功能 .....</b>      | <b>24</b> |
| 4.1 蛇吞象 .....               | 24        |
| 4.2 时光盒 .....               | 25        |
| 4.3 MoSCoW 法则 .....         | 26        |
| 4.4 时光盒中活动的控制 .....         | 27        |
| 4.5 是否使用时光盒法 .....          | 32        |
| 4.6 最坏的情况 .....             | 32        |
| 4.7 要点回顾 .....              | 33        |
| <b>第5章 协同工作 .....</b>       | <b>34</b> |
| 5.1 全面改变的机会 .....           | 34        |
| 5.2 项目中的角色 .....            | 34        |
| 5.3 项目结构 .....              | 35        |
| 5.4 要点回顾 .....              | 36        |
| <b>第6章 现实中的敏捷项目经理 .....</b> | <b>38</b> |
| 6.1 有什么不同 .....             | 38        |
| 6.2 DSDM 项目的计划 .....        | 38        |
| 6.3 风险管理 .....              | 41        |
| 6.4 进度监控 .....              | 41        |
| 6.5 工作量 .....               | 42        |
| 6.6 要点回顾 .....              | 43        |
| <b>第7章 对软件组织的影响 .....</b>   | <b>44</b> |
| 7.1 制定决策 .....              | 44        |
| 7.2 用户参与 .....              | 45        |
| 7.3 更好的沟通 .....             | 46        |
| 7.4 专题研讨会 .....             | 47        |
| 7.5 培训用户 .....              | 48        |
| 7.6 要点回顾 .....              | 49        |
| <b>第8章 质量问题 .....</b>       | <b>50</b> |
| 8.1 “足够好”的软件 .....          | 50        |
| 8.2 质量设计 .....              | 50        |
| 8.3 测试 .....                | 51        |
| 8.4 DSDM 和 TickIT .....     | 52        |

|               |                            |           |
|---------------|----------------------------|-----------|
| 8.5           | 旧瓶装新酒 .....                | 53        |
| 8.6           | 能力成熟度模型 .....              | 54        |
| 8.7           | 要点回顾 .....                 | 55        |
| <b>第 9 章</b>  | <b>建立原型不是在浪费时间 .....</b>   | <b>57</b> |
| 9.1           | 架起语言的桥梁 .....              | 57        |
| 9.2           | 可是用户总是不断改变主意 .....         | 57        |
| 9.3           | 原型的种类 .....                | 58        |
| 9.4           | 得到有效的反馈 .....              | 59        |
| 9.5           | 保持控制 .....                 | 60        |
| 9.6           | 要点回顾 .....                 | 60        |
| <b>第 10 章</b> | <b>敏捷的职业标准 .....</b>       | <b>62</b> |
| 10.1          | “不再是快而粗劣” .....            | 62        |
| 10.2          | 技术和人性 .....                | 62        |
| 10.3          | 自我管理 .....                 | 63        |
| 10.4          | 要点回顾 .....                 | 64        |
| <b>第 11 章</b> | <b>DSDM 环境中的极限编程 .....</b> | <b>65</b> |
| 11.1          | 竞争的敏捷方法 .....              | 65        |
| 11.2          | XP 与 DSDM 共舞 .....         | 65        |
| 11.3          | XP 的独特要求 .....             | 66        |
| 11.4          | 要点回顾 .....                 | 67        |
| <b>第 12 章</b> | <b>技术支持 .....</b>          | <b>68</b> |
| 12.1          | 技术支持的必要性 .....             | 68        |
| 12.2          | DSDM 支持环境 .....            | 68        |
| 12.3          | 测试工具 .....                 | 70        |
| 12.4          | 配置管理工具 .....               | 70        |
| 12.5          | 有效地使用工具 .....              | 70        |
| 12.6          | 要点回顾 .....                 | 71        |
| <b>第 13 章</b> | <b>保持系统的活力 .....</b>       | <b>72</b> |
| 13.1          | 时光盒法 .....                 | 72        |
| 13.2          | 设置优先级 .....                | 73        |
| 13.3          | 变更的交付 .....                | 73        |
| 13.4          | 要点回顾 .....                 | 73        |

## 第二部分 案例研究

|                                 |    |
|---------------------------------|----|
| 第 14 章 DSDM 在英航电子商务中的实施 .....   | 76 |
| 14.1 简介 .....                   | 76 |
| 14.2 创建 eBA .....               | 76 |
| 14.3 引入 DSDM .....              | 77 |
| 14.4 eBA 中的角色 .....             | 77 |
| 14.5 奖励方案 .....                 | 78 |
| 14.6 环境 .....                   | 79 |
| 14.7 项目实例 .....                 | 80 |
| 第 15 章 DSDM 与消除合约上的分歧 .....     | 82 |
| 15.1 简介 .....                   | 82 |
| 15.2 项目的生命周期 .....              | 82 |
| 15.3 项目 .....                   | 83 |
| 15.4 方法 .....                   | 84 |
| 15.5 个性与角色 .....                | 85 |
| 15.6 功能建模迭代和设计编程迭代 .....        | 85 |
| 15.7 利益 .....                   | 87 |
| 15.8 经验教训 .....                 | 87 |
| 第 16 章 DSDM 在非 IT 项目中的运用 .....  | 89 |
| 16.1 引言 .....                   | 89 |
| 16.2 项目环境 .....                 | 89 |
| 16.3 项目 .....                   | 90 |
| 16.4 维护 / 项目之后 .....            | 92 |
| 16.5 功能建模迭代讨论会 .....            | 92 |
| 16.6 结论 .....                   | 93 |
| 第 17 章 面向对象的 DSDM 项目 .....      | 95 |
| 17.1 引言 .....                   | 95 |
| 17.2 OO/UML 与 DSDM 的理论兼容性 ..... | 95 |
| 17.3 项目动机 .....                 | 96 |
| 17.4 可行性研究 .....                | 96 |
| 17.5 业务研究 .....                 | 97 |
| 17.6 功能建模迭代 .....               | 98 |

|               |                                 |            |
|---------------|---------------------------------|------------|
| 17.7          | 设计和编程迭代 .....                   | 99         |
| 17.8          | 实施阶段 .....                      | 99         |
| 17.9          | 存在的问题 .....                     | 100        |
| 17.10         | 结论 .....                        | 101        |
| <b>第 18 章</b> | <b>DSDM 如何降低离岸工作的风险 .....</b>   | <b>103</b> |
| 18.1          | 引言 .....                        | 103        |
| 18.2          | 管理海外项目的挑战 .....                 | 103        |
| 18.3          | 为什么采用一体化模式 .....                | 104        |
| 18.4          | DSDM 与资源一体化相结合所具有的好处 .....      | 104        |
| 18.5          | 向异地转移 DSDM .....                | 105        |
| 18.6          | 异地 DSDM 生命周期概述 .....            | 106        |
| 18.7          | 为适应异地工作增设的 DSDM 新角色 .....       | 108        |
| 18.8          | 结论 .....                        | 108        |
| <b>第 19 章</b> | <b>迷途羔羊 .....</b>               | <b>109</b> |
| 19.1          | 技术协调人失踪了 .....                  | 109        |
| 19.2          | 失去用户代表的案例 .....                 | 109        |
| 19.3          | 貌合神离的时光盒 .....                  | 110        |
| 19.4          | 谁需要测试 .....                     | 110        |
| 19.5          | 我们无法与用户交谈 .....                 | 111        |
| <b>第 20 章</b> | <b>量化的 DSDM 项目 .....</b>        | <b>112</b> |
| 20.1          | 走进 DSDM .....                   | 112        |
| 20.2          | 项目日记 .....                      | 117        |
| 20.3          | 项目数据 .....                      | 118        |
| 20.4          | 结论与经验教训 .....                   | 123        |
| 20.5          | 项目复审会的结果 .....                  | 124        |
| <b>第 21 章</b> | <b>从 DSDM 作坊到 DSDM 工厂 .....</b> | <b>127</b> |
| 21.1          | 引言 .....                        | 127        |
| 21.2          | 项目情况 .....                      | 127        |
| 21.3          | DSDM 工具箱 .....                  | 131        |
| 21.4          | 后续工作 .....                      | 134        |
| <b>第 22 章</b> | <b>过程改进中的 DSDM .....</b>        | <b>135</b> |
| 22.1          | 背景 .....                        | 135        |