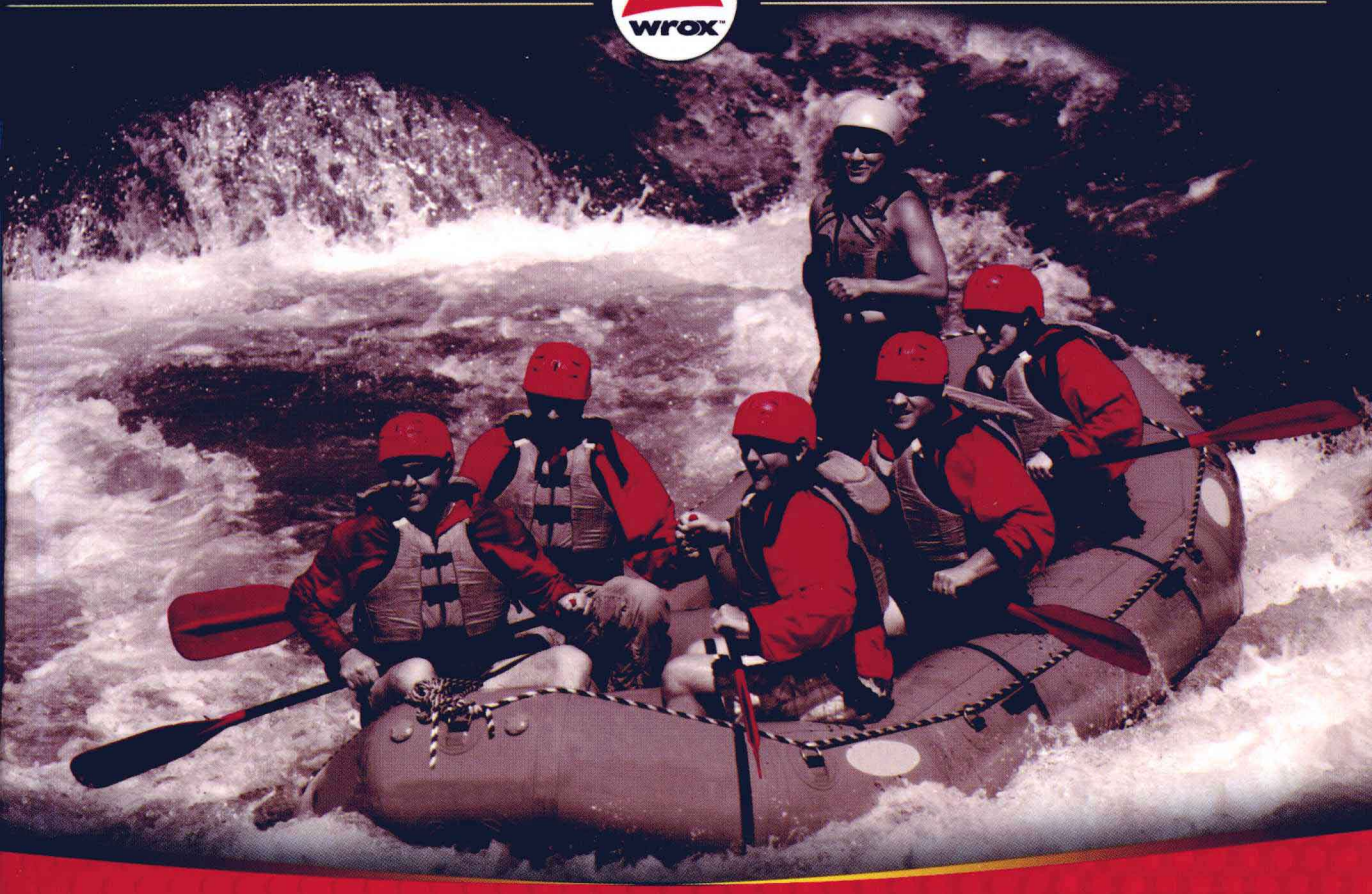
Join the discussion @ p2p.wrox.com

Wrox Programmer to Programmer™



Professional Application Lifecycle Management with Visual Studio 2010

Visual Studio 2010
软件生命周期管理高级教程

(美) Mickey Gousset 等著
Brian Keller
窦朝晖 司倩然 译

清华大学出版社

Visual Studio 2010 软件生命周期管理高级教程

(美) Mickey Gousset 等著
Brian Keller
窦朝晖 司倩然 译

清华大学出版社
北 京

Mickey Gousset，Brian Keller，et al.
Professional Application Lifecycle Management with Visual Studio 2010
EISBN：978-0-470-48426-5

北京市版权局著作权合同登记号　图字：01-2010-5685

图书在版编目(CIP)数据

Visual Studio 2010 软件生命周期管理高级教程/ (美) 古塞(Gousset, M.)，(美) 凯勒(Keller, B.)　等著；窦朝晖，司倩然　译. —北京：清华大学出版社，2011.8

书名原文：Professional Application Lifecycle Management with Visual Studio 2010

ISBN 978-7-302-25550-5

Ⅰ.V… Ⅱ.①古… ②凯… ③窦… ④司… Ⅲ.计算机网络—程序设计 Ⅳ.TP393

中国版本图书馆 CIP 数据核字(2011)第 082608 号

责任编辑：王　军　韩宏志
装帧设计：孔祥丰
责任校对：胡雁翎
责任印制：何　芊
出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦 A 座
http://www.tup.com.cn　　**邮　　编**：100084
社　总　机：010-62770175　　**邮　　购**：010-62786544
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
印 装 者：清华大学印刷厂
经　　销：全国新华书店
开　　本：185×260　**印　张**：35.25　**字　数**：902 千字
版　　次：2011 年 8 月第 1 版　　**印　　次**：2011 年 8 月第1 次印刷
印　　数：1～4000
定　　价：78.00 元

产品编号：038462-01

作者简介

Mickey Gousset 是 Infront Consulting Group 公司(主要从事 Microsoft System Center 产品系列咨询业务)的高级技术开发人员。他已经连续 5 年荣获 Microsoft Team System MVP，不仅获得了 Team Foundation Server 和 SCOM 2007 专业开发人员认证，而且与 Jean-Luc David 和 Erik Gunvaldson 合著了 *Professional Team Foundation Server*(Indianapolis: Wiley, 2006)一书。Gousset 开通了一个专门研讨 Visual Studio Team System 和 Visual Studio 2010 的社区网站 Team System Rocks! (http://www.teamsystemrocks.com)，在这里撰写了很多关于 Visual Studio 和 Team Foundation Server 的博客文章。他还与他人合办了广受欢迎的 Team Foundation Server 播客 Radio TFS (http://www.radiotfs.com)。他曾在各种用户团体、编程夏令营和重要会议(包括 Microsoft Tech Ed Developer——North America 2008 和 2009)发表过有关 Visual Studio 和 Team Foundation Server 的演讲。Mickey 的业余爱好十分广泛，包括玩 Xbox Live(玩家标识为 HereBDragons)以及参加地方社团剧院的演出。有时也会与爱妻 Amye 及两个吉娃娃 Lucy 和 Linus 依偎在沙发上共度闲暇时光。

Brian Keller 是 Microsoft 的高级技术大使，他的专业方向是 Visual Studio 和应用程序生命周期管理。Keller 自 2002 年起便与 Microsoft 结缘，出席在世界各地举办的会议，包括 TechEd、Professional Developers Conference(PDC)和 MIX。Keller 还是 MSDN 的 Channel 9 网站的常客，经常与他人共同主持热门节目 This Week on Channel 9。Keller 在工作之余最喜欢户外运动，如攀岩、背包旅行、滑雪和冲浪等。

Ajoy Krishnamoorthy 担任 Microsoft 模式与实践组的高级产品经理，主要负责规划“模式与实践”投资领域和商业策略。Krishnamoorthy 此前曾担任 Microsoft Visual Studio Team System 的高级产品经理。他拥有十多年的咨询经验，曾做过开发人员、架构师和技术项目经理。Krishnamoorthy 在网络和杂志上发表过多篇文章，并与他人合著过几本介绍 ASP.NET 的书籍。他的博客是 http://blogs.msdn. com/ajoy。Krishnamoorthy 获得了俄亥俄州立大学的 MBA。工作之余他会与家人团聚，或者与好友玩棋牌、看体育比赛或练习打手鼓。

Martin Woodward 目前担任 Microsoft Visual Studio Team Foundation Server Cross-Platform Tools Team 的计划经理。Woodward 在加盟 Microsoft 之前就已被评选为年度 Team System MVP。Woodward 曾多次在国际会议上发表有关 Team Foundation Server 的演讲。Woodward 通过总结 5 年多来在各种规模的公司中使用 Team Foundation Server 的实际经验，对该产品的内部原理有了深刻独到的见解，同时也乐于与大家交流心得。Woodward 的博客是 http://www.woodwardweb.com。

前　言

在 1999 年 6 月，Microsoft 开始重新评估如何将 Visual Studio 作为软件开发过程的一部分。Visual Studio 以代码为焦点进行快速应用程序开发的特点，使它的效率变得非常高， Microsoft 就是通过这些功能继续满足程序员的需求的。但如何将程序员组成一个团队来进行工作，Microsoft 在这方面却做得不多。对于软件架构师而言，他们又是如何与编程团队协同工作的呢？对于测试人员、项目管理人员呢？

许多团队开始使用第三方的、内部的和开发商提供的工具构建自己的解决方案，以解决诸如版本控制、bug 跟踪和团队沟通等问题。但使用这些混杂的工具构建解决方案以及对其进行维护时需要很高的技巧，并且很难集成。Microsoft 设计了一个能满足整个软件开发团队需求的集成工具集，以求解决这些问题。因此 Visual Studio Team System 便应运而生，并第一次与 Visual Studio 2005 产品系列一起发布。

在 Microsoft 内部，对于曾经进行的一些非常复杂的软件项目，多年来一直在采用一些工具和技术进行管理。Team System 就是在这些工具和技术的基础上开发出来的。Team System 不仅引起了程序员的兴趣，而且引起了开发团队所有成员(架构师、应用程序开发人员、数据库开发人员、测试人员和项目管理人员)的兴趣。Team System 旨在满足整个软件开发生命周期的要求，因此更多地被称为应用程序生命周期管理工具。

三年之后，这个版本发展为了 Visual Studio 2008 Team System，这一版本包含了更多可供项目团队所有成员使用的工具和功能。

0.1　名称变化

细心的读者会发现，在本书的书名中没有出现 Team System 一词。并且，除了刚读到的简单回顾之外，在本书其他任何地方将再也看不到 Team System 一词。这究竟是什么原因？

Microsoft 通过一些调研发现，如果使用两个 Visual Studio 商标名称，客户就会搞不清产品之间的差异。对 Visual Studio 的定位是供开发人员使用的基本工具，而对 Visual Studio Team System 的定位是供软件开发团队使用的一组工具。但是，几乎所有专业的开发人员都是与团队协同工作的，因此，Team System 这一术语就在一定程度上失去了意义。基于这一原因，Microsoft 决定去掉 Team System 这一名称，并将所有功能都合并到统一的 Visual Studio 品牌家族中。

这一变化还有一些更微妙的原因。然而无论名称长短，在 2010 产品系列中，Team System 这一名词将不复存在，但是其中仍将包含其产品和技术，而且这些产品和技术将比以往更加完善。

0.2 Visual Studio 2010 产品系列

表 0-1 列出了 Visual Studio 2010 新的产品系列。

表 0-1

产品名称	说明
Microsoft Visual Studio 2010 Ultimate with MSDN	针对软件团队的应用程序生命周期管理工具全集，可帮助开发团队保证从设计到部署的质量
Microsoft Visual Studio 2010 Premium with MSDN	这是一个完全工具集，可帮助开发人员交付可伸缩的高质量应用程序
Microsoft Visual Studio 2010 Professional with MSDN	这是针对基本开发任务的主要工具，可帮助开发人员轻松地实现自己的想法
Microsoft Visual Studio Test Professional 2010 with MSDN	这是针对一般的手动测试人员的主要工具。这类测试人员需要定义和管理测试用例、执行测试并记录文件 bug。Test Professional 产品包含了 Miscrosoft Test Manager，Test Manager 的相关内容将在第 14 章进行介绍
Microsoft Visual Studio Team Foundation Server 2010	这是一个服务器组件，用于团队开发、版本控制、工作条目跟踪、自动生成和报表
Microsoft Visual Studio Lab Management 2010	这是支持虚拟实验室的工具，和其他 Visual Studio 工具配合使用时，可以使开发人员和测试人员更好地进行协作

Visual Studio 2010 Premium 包含 Visual Studio 2010 Professional 的所有功能，而 Visual Studio 2010 Ultimate 又包含 Visual Studio 2010 Premium 的所有功能。Visual Studio 2010 Ultimate 还包含 Visual Studio Test Professional 2010 的所有功能。

表 0-2 列出了 Visual Studio 2010 各个版本所包含的功能。

表 0-2

Visual Studio 2010 版本	包含功能
Microsoft Visual Studio 2010 Ultimate	IntelliTrace
	统一建模语言(UML)
	Architecture Explorer
	逻辑类设计器
	测试用例管理
	手动测试
	测试记录与回放
	分层图
	Web 性能测试
	负载测试

(续表)

Visual Studio 2010 版本	包 含 功 能
Microsoft Visual Studio 2010 Premium	编码 UI 测试(Coded user interface testing)
	性能分析
	代码覆盖
	数据库更改管理
	数据库单元测试
	测试影响分析
	静态代码分析
	代码度量指标
	数据库部署
	测试数据生成
Microsoft Visual Studio 2010 Professional	Silverlight 开发
	Web 开发
	Windows Presentation Foundation(WPF)开发
	多核开发
	云开发
	Windows 窗体开发
	Office 开发
	可定制的 IDE

本书将主要介绍 Visual Studio 2010 Premium 和 Visual Studio 2010 Ultimate 所包含的功能。

0.3 现代软件开发所面临的挑战

无论开发团队的规模如何，软件开发人员都会面临一些共同的挑战。在商业上需要追求产出投入比的最大化——因此，软件开发的时间必须最短，且没有错误。

软件开发人员所面临的挑战包括：

- **集成问题**——软件开发团队所用的大多数工具通常来自第三方的厂商。在使用这些工具进行集成时所面临的主要挑战在于，很多时候要求将数据复制到多个系统中。每一个应用程序都有一个学习曲线，而且将信息从一个应用程序转换到另一个(不兼容的)应用程序是很令人沮丧和费时的。
- **地域分散的团队**——很多开发和管理工具并没有考虑开发团队分布在不同地方的情况。因此，要得到很准确的报告是非常困难的，而且对沟通和协作工具的支持很差。结果就导致需求与规范不能正确映射，从而导致项目延期和引入错误。全球化的团队要求将统一的设计、过程和软件配置管理集成到一个包中。但是能够满足所有这些需求的软件包可以说是凤毛麟角，即使有，价格也非常昂贵。

- **角色划分**——专业化对开发团队来说是一个巨大的问题。专家可以假定其他部门能够意识到在状态报告中不出现的信息，但从总体上看，这对项目可能会产生巨大的影响。不同部门人员之间的沟通是普遍存在的巨大挑战。
- **报告质量差**——这是角色划分问题的一个分支。很多时候，每个团队必须手动生成报告，从而导致效率低下。现在还没有任何有效的工具能将不同来源的数据收集在一起。因此，项目经理在进行有效决策时就会缺乏必要的数据。
- **缺乏过程指导**——特定的编程风格不是很容易实现。如果对代码的更改是非周期性的，则会带来非常严重的问题，需要花费数小时甚至数天的额外工作才能解决。如今的软件依赖程度非常高，遗憾的是，大多数工具不包含或没有实施过程指导，这将会导致过程和工具的不匹配。
- **不重视测试**——由于周期越来越短，加之缺乏测试，因此，在过程后期必然会产生代码缺陷。软件开发工具厂商几乎完全忽视了手动测试人员的需求，从而导致开发人员和测试人员之间的合作很差，结果是需要经常来回返工，既浪费精力，又不能排除软件缺陷。
- **沟通问题**——大多数公司使用了各种各样的沟通方法(如 Email、即时消息、备忘录、便签)给团队成员发送信息。然而如果不小心，则很容易丢失一条纸质消息，或删除一条重要的 Email 消息。由于没有可供管理团队进行沟通的集中统一的系统，因此，就需要定期召开例会以跟踪团队的工作进展，这是很费时的，同时还需要很多手动沟通过程(如发送 Email，以及剪贴报告)。其根本原因在于工具和项目经理之间没有沟通渠道。

有些公司引入一些方法学和实践经验以简化和组织软件设计过程，但必须在这些方法学之间进行权衡。目的就是为了使设计过程可预测；因为在一个可预测的环境中，这些方法学可以使项目保持正常运行。但另一方面，这些方法学会给设计过程带来额外的工作(如生成报告)。如果开发人员花费太多时间来做这些工作，那么将会降低效率，从而导致公司缺乏竞争力。

0.4 进入 Visual Studio 2010

应用程序生命周期管理是在生命期所有阶段对软件开发项目进行管理的一种理念。建立在 Visual Studio 2005 Team System 和 Visual Studio 2008 Team System 基础上的 Visual Studio 2010 的生命周期管理功能，其设计初衷就是为了减轻或消除这些挑战。

Visual Studio 2010 的生命周期管理功能背后有 3 个基本原则：效率、集成和可扩展性。

效率是通过下列方法提高的：

- **协作**——Team Foundation Server 集中了所有的团队协作。通过 Team Foundation Server 2010 可以对 bug、需求、任务、测试用例、源代码和应用程序生成进行统一管理。所有的报表功能也集成其中，从而使项目经理可以很容易地监控项目的整个过程，而不必关心度量标准的来源。
- **复杂度管理**——软件开发项目比以前更复杂，并且复杂度仍然在逐年增加。Team Foundation Server 主要通过跟踪整个软件开发过程来帮助管理软件开发项目的复杂度，从而保证整个开发团队能看到项目在任意一个时刻的状态和工作流程。另外，Visual

Studio 2010 Ultimate 所提供的架构工具可以基于现有代码进行可视化的逆向工程，这将有助于降低应用程序的复杂性。

通过下列方法增强了集成：

- **集成工具**——这些工具使部门之间的沟通变得更容易。更重要的是，这些工具消除了信息闭塞。有了 Visual Studio 2010 产品系列之后，集成不再是事后考虑的问题，而是工具集首先要考虑的核心问题。
- **可视化**——Visual Studio 2010 和 Team Foundation Server 增强了项目的可视性。项目经理可以很容易地浏览与项目相关的度量标准，而且可以通过识别模式和趋势事先指出存在的问题。

可扩展性是通过下列方法实现的：

- Team Foundation Core Servives API——该平台的很多地方对开发人员而言是透明的，从而提供了很多可扩展的机会，可以创建很多能够与 Team Foundation Server 进行集成的定制工具。
- IDE——Visual Studio 2010 集成开发环境本身就是可扩展的，允许第三方和最终用户利用其他工具的功能添加任何东西，甚至可以将一个新的语言编译器添加到该开发环境中。

0.5　应用程序生命周期管理

为了更好地说明 Visual Studio 2010 是如何帮助我们进行生命周期管理的，本书将研究 eMockSoft 公司(一个虚构的软件开发公司)的一个典型场景。eMockSoft 最近与一个分销商签署了一份发布其产品目录的合作协议。分销商要求一个安全的 Web 服务，能在内部和外部合作组织之间传递库存和定价信息。

下面将研究如何使用应用程序生命周期管理和 Visual Studio 2010 工具实现这一场景。

0.5.1　需求

项目管理人员与发起人面谈从而获得该项目的需求。需求会向开发团队说明项目发起人希望软件具有什么样的功能。项目管理人员可以使用所选择的工具(如 Visual Studio、Excel 或 Microsoft Project)将这些需求存储在 Team Foundation Server 中。Team Foundation Server 能够生成一个基于 SharePoint 的团队项目门户，项目发起人可以通过这个门户对需求进行确认和跟踪。这个团队项目门户还可以将存储在 Team Foundation Server 中的报告和其他内容展现在一个基于 Web 的 SharePoint 站点上。

此时，基础结构架构师就可以开始进行系统设计了。

0.5.2　系统设计和建模

根据客户的规范，基础结构架构师可以使用 Visual Studio 2010 中新的 UML 工具定义外部 Web 服务。同时，项目管理人员可以对设计的进度，包括生成的图进行跟踪。项目管理人员可以根据规范将任务向下分解为将要分配给团队中各个开发人员的任务(也存储在 Team Foundation Server 中)。

0.5.3 代码生成

开发人员接受所分配的工作，对架构师所设计的 UML 图进行审查，并对规范进行检查——该应用程序要求使用 Web Services Enhancements(WSE) 3.0 实现一个安全的 Web 服务。之后，开发人员编写必要的代码，并使用 Visual Studio 2010 提供的静态代码分析和测试工具进行一些初步的测试。在工作过程中，开发人员可以随时将代码和测试签入到 Team Foundation Server 2010 中。

0.5.4 测试

测试人员通过监控夜间进行的项目生成和自动测试来检查开发团队的进度。利用 Visual Studio Lab Management 2010，每一个夜间生成都会触发虚拟机的自动生成过程，这样，每天早晨测试人员就可以用这些自动生成的东西开始进行测试。测试人员每天都会使用 Visual Studio Test Professional 2010 设计、管理和执行一组手动测试用例，并向开发团队提交项目中存在的潜在 bug。测试人员可以将 bug 记录在 Team Foundation Server 中，而 Team Foundation Server 会将其分配给开发人员进行修改。

所有 bug 都存储在 Team Foundation Server 中，并且会提供给团队成员和项目发起人，从而使他们可以对该项目的进度有一个全面的了解。

0.5.5 小结

以上只是一个简单的示例，仅说明了使用 Visual Studio 2010 进行应用程序生命周期管理的几种方法。本书中还有其他一些示例，这些示例可以帮助团队成为有机统一的整体，并开发出更好的软件。

0.6 本书读者对象

本书主要针对的是商业或企业软件开发领域的专业团队——即针对的是中高级用户。如果您属于下列人员，则可能会发现本书的价值所在：

- 希望了解 Visual Studio 2010 系列产品是如何帮助自己完成工作的开发人员、测试人员或架构师
- 软件开发项目的管理人员

本书并非为没有一点基础的初学者而编写。本书的重点在于介绍工具的实际应用、代码示例和常见的应用场景。本书的组织形式使它既可以作为入门指南，又可以作为建模、设计、测试和协调企业各层次解决方案的参考书。

Visual Studio 2010 适用于任何规模的软件开发团队。因此，无论您的开发团队有 5 个成员还是 2000 个成员，本书所提供的有关 Visual Studio 2010 和生命周期管理的知识都会对您大有裨益。与大多数 Wrox 图书不同，本书针对的是软件开发组织的所有角色——架构师、开发人员、测试人员、项目经理和管理人员——而不仅仅是开发人员。

0.7　本书主要内容

本书全面介绍了 Visual Studio 2010 的应用程序生命周期管理功能。本书根据软件开发团队的不同角色分为如下 5 个主要部分：

- 第 I 部分：架构师
- 第 II 部分：开发人员
- 第III部分：测试人员
- 第IV部分：Team Foundation Server
- 第 V 部分：项目/过程管理

第 I 部分：架构师

这一部分主要介绍的是 Visual Studio 2010 中与架构师角色相关的工具。在对架构的相关概念进行简单介绍之后，将对所有可用的新的 UML 工具进行详细介绍，包括用例图、活动图、顺序图、类图和组件图。然后将学习 Architecture Explorer (架构浏览器)，以及如何使用 Architecture Explorer 了解应用程序的架构。最后还将对层次图进行讨论。

第 II 部分：开发人员

这一部分将介绍使用 Visual Studio 2010 创建应用程序的开发人员最感兴趣的所有主题。包括单元测试、重构、静态代码分析和代码覆盖等内容都将详细涉及。此外，还将介绍处理开发、测试和数据库应用程序部署的功能，以及在新功能模块 IntelliTrace 中所用到的高级应用程序调试技术。

第III部分：测试人员

Visual Studio 2010 为测试人员提供了大量工具，这一部分将对所有这些工具进行详细的介绍。首先，研究 Web 性能和负载测试。然后介绍新的手动测试功能，以及用户界面自动测试的功能。在该部分的最后将对 Visual Studio 2010 的实验室管理功能进行讨论，这是一个新功能，它可以利用虚拟机自动构建能够执行测试的测试环境。

第IV部分：Team Foundation Server

这一部分将介绍 Team Foundation Server 所提供的功能。首先介绍 Team Foundation Server 2010 的新架构，然后详细介绍版本控制系统以及一些使用 Team Foundation Server 进行分支与合并的最佳实例。最后详细介绍自动生成过程 Team Foundation Build 的一些新变化。

第 V 部分：项目/过程管理

本书最后一部分将介绍 Visual Studio 2010 和 Team Foundation Server 的项目和过程管理功能。该部分不仅会介绍产品中附带的新过程模板以及新的需求总表与产能规划功能。还将介绍 Team Foundation Server 中附带的报告。最后会介绍一些常用的过程模板定制。

0.8 本书约定

为了帮助您从文本得到最重要的信息，记住所发生的事情，本书将使用若干约定。

像这样的框中的信息是重要的、不能遗忘的，是与周围文本密切相关的。

注记、技巧、提示和小窍门将在此类框中列出。

文中风格：

当前讨论的补充说明将采用这种格式。

以两种不同方式给出代码：

- 大多数示例代码使用等宽字体。
- 少部分代码采用加粗格式，表示当前上下文中十分重要的代码。

0.9 如何下载本书的示例代码

当读者学习本书中的示例时，可以手工输入所有的代码，也可以使用本书附带的源代码文件。本书使用的所有源代码都可以从本书合作站点 http://www.wrox.com 下载。登录到站点 http://www.wrox.com/，使用 Search 工具或使用书名列表就可以找到本书。接着单击本书细目页面上的 Download Code 链接，就可以获得所有的源代码。

许多图书的书名都很相似，所以通过 ISBN 查找本书是最简单的，本书的 ISBN 是 978-0-470-48426-5。

在下载了代码后，只需用自己喜欢的解压缩软件对它进行解压缩即可。另外，也可以进入 www.wrox.com/dynamic/books/download.aspx 上的 Wrox 代码下载主页，查看本书和其他 Wrox 图书的所有代码。

0.10 勘误表

尽管我们已经尽了各种努力来保证文章或代码中不出现错误，但是错误总是难免的，如果您在本书中找到了错误，例如拼写错误或代码错误，请告诉我们，我们将非常感激。通过勘误

表，可以让其他读者避免受挫，当然，这还有助于提供更高质量的信息。

要在网站上找到本书的勘误表，可以登录 http://www.wrox.com，通过 Search 工具或书名列表查找本书，然后在本书的细目页面上，单击 Book Errata 链接。在这个页面上可以查看到 Wrox 编辑已提交和粘贴的所有勘误项。完整的图书列表还包括每本书的勘误表，网址是 www.wrox.com/misc-pages/booklist.shtml。

包括勘误表链接的完整图书列表可从 www.wrox.com/misc-pages/booklist.shtml 获得。

如果在勘误表中没找到您自己发现的错误，请给 fwkbook@tup.tsinghua.edu.cn 发电子邮件，我们会检查您的信息，如果是正确的，我们将在本书的后续版本中采用。

0.11　P2P.WROX.COM

P2P 邮件列表是为作者和读者之间的讨论而建立的。读者可以在 p2p.wrox.com 上加入 P2P 论坛。该论坛是一个基于 Web 的系统，用于传送与 Wrox 图书相关的信息和相关技术，与其他读者和技术用户交流。该论坛提供了订阅功能，当论坛上有新贴子时，会给您发送您选择的主题。Wrox 作者、编辑和其他业界专家和读者都会在这个论坛上进行讨论。

在 http://p2p.wrox.com 上有许多不同的论坛，帮助读者阅读本书，在读者开发自己的应用程序时，也可以从这个论坛中获益。要加入这个论坛，需执行下面的步骤：

(1) 进入 p2p.wrox.com，单击 Register 链接。

(2) 阅读其内容，单击 Agree 按钮。

(3) 提供加入论坛所需的信息及愿意提供的可选信息，单击 Submit 按钮。

(4) 随后就可以收到一封电子邮件，其中的信息描述了如何验证账户，完成加入过程。

不加入 P2P 也可以阅读论坛上的信息，但只有加入论坛后，才能发送自己的信息。

加入论坛后，就可以发送新信息，回应其他用户的贴子。可以随时在 Web 上阅读信息。如果希望某个论坛给自己发送新信息，可以在论坛列表中单击该论坛对应的 Subscribe to this Forum 图标。

对于如何使用 Wrox P2P 的更多信息，可阅读 P2P FAQ，了解论坛软件的工作原理，以及许多针对 P2P 和 Wrox 图书的常见问题解答。要阅读 FAQ，可以单击任意 P2P 页面上的 FAQ 链接。

目　录