

<<构筑敏捷的开发团队>>

图书基本信息

书名：<<构筑敏捷的开发团队>>

13位ISBN编号：9787121106491

10位ISBN编号：7121106493

出版时间：2010-5

出版时间：电子工业出版社

作者：高阳 等编著

页数：523

字数：767000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<构筑敏捷的开发团队>>

前言

随着软件开发日趋国际化，对软件的质量要求和管理要求也随之提高，微软顺应应用程序生命周期管理在业界逐渐被接受和认可的趋势，发布了Visual Studio 2010。它是微软新一代的开发工具和平台，致力于引领下一代平台发展，增强开发人员团队的敏捷思想管理与协作能力，提高IT技术人员的工作效率。

利用各种工具合理化地辅助每个关键环节进行管理（ALM）是Visual Studio 2010的重要特性，它还支持统一建模语言（UML）和特定领域语言（DSL），能够为各类.NET技术人员提供合适的工具，而这也只是微软更大模型平台的一部分。

读者们需要注意的是，微软从2010版本以后，已经不再用VSTS进行命名了，而是采用Visual Studio 2010（包括开发的IDE、架构设计的UML和周期管理TFS等）来统一命名。

本书缘起 在微软Visual Studio 2010工具中，真正的主角已经不再是作为编码工具的VS IDE，而是ALM（Application Lifecycle Management）的核心TFS（Team Foundation Server），这是微软Visual Studio系列产品革命性的进步。

由于在其之前的版本（VSTS2005/2008）存在安装配置复杂、硬件环境要求高等问题，再加上没有很强大的建模架构设计工具和软件开发敏捷过程模型思想进行指导，所以在微软.NET方向工作的技术圈子中没有得到大规模的有效普及。

如今，微软Visual Studio 2010的一系列产品测试开发过程（CTP Beta1 Beta2 RC RTM）功能更加成熟，已经具有安装简单快捷、配置简单、具有功能和易用性均强的建模设计工具，以及具有bug管理协作能力的测试管理工具（Microsoft Test Manager 2010）等特点。

这种以团队开发、协作管理、架构建模设计和软件开发过程模型（Scrum、XP等）合为一体的开发工具是软件工程发展的趋势，但是很多.NET编程开发人员却对之了解甚少。

作为一名IT职业经理人和软件开发人员，我们一直留意微软Visual Studio的相关图书，但市面上却很少有图书对其功能与实践进行详细的介绍。

我理想中的书应该是一本实践性很强的书，是真正能够帮助从事.NET相关工作人员的图书。

如果这本书能实现我的目标，则对大家的技术实践与领悟软件过程思想会有不小的帮助和提升。

我们在大量研究和实践的基础上，还查阅了大量相关资料，加强研究，力求融会贯通，其间还得到了微软相关部门的大力支持，费时8个月，终于写成这本书。

本书的主要目的是帮助从事微软相关技术工作和软件开发工作的人员，使其能够迅速熟悉Visual Studio 2010的新功能特点与应用，并可以使用Visual Studio 2010构建自己的开发与测试环境。

在本书里，我们将对.NET开发人员、架构设计人员和测试等人员在工作中所遇到的问题、难点进行分析，并使用Visual Studio 2010加以解决。

相信这些问题的解决会给开发人员带来不少收获，并有助于读者拓展视野，更快地掌握Visual Studio 2010主要特性功能的应用及项目过程的管理技巧，从而起到抛砖引玉的作用。

<<构筑敏捷的开发团队>>

内容概要

本书介绍了敏捷开发团队依托Visual Studio 2010、采用Scrum / XP等敏捷过程模型方法从各个方面组织开发其项目的过程，读后使人茅塞顿开。

本书从开发到管理的各方面对Visual Studio 2010的新功能进行了详细的介绍，从软件生命周期(SDLC)的角度对其分析、设计、开发、测试等工具列举了较详细的实践示例，并覆盖了使用Visual Studio 2010进行增量构建和集成、版本控制、架构建模设计(探索分析建模)、敏捷驱动开发、测试管理和报表管理等功能的实践内容，从而让读者深入了解Visual Studio 2010应用的精髓，是学习Visual Studio 2010实践的优秀参考书。

本书适合于.NET技术爱好者及初、中级开发、管理和测试等人员参考学习，特别有助于.NET技术人员学习软件项目的管理和应用，也可作为相关培训机构的参考教材，还可以作为高等院校相关专业师生的参考书。

<<构筑敏捷的开发团队>>

作者简介

高阳，微软最有价值专家（ASP.NET MVP），复旦大学软件硕士，曾任职于微软ATC、MCS开发经理，北京中科领航开发部经理，大展信息科技（北京）有限公司技术总监和北京某公司CTO等职务。有10多年左右的管理与其他程序设计经验和7年左右的.NET应用程序开发经验，对于微软.NET企业应用开发与架构设计有较为深入的理论见识与实践经验。

著有《我也能做CTO之程序员职业规划》《.NET软件设计新思维——像搭积木一样搭建软件》《深入体验Microsoft Office SharePoint Server实践开发》《Windows Powershell 2.0应用编程最佳实践》等图书。

<<构筑敏捷的开发团队>>

书籍目录

- 第1章 VS2010实践与软件开发 1.1 软件开发概述 1.1.1 软件开发过程 1.1.2 软件开发问题
 1.1.3 VS 2010中的软件开发 1.1.4 项目成功因素与VS 1.2 VS 2010功能概述 1.2.1 VS 2010的功能模块 1.2.2 Visual Studio产品介绍 1.2.3 Team Foundation平台 1.3 Visual Studio 2010的新特性
 1.3.1 改进软件开发生命周期 1.3.2 保证软件的健壮性和可靠性 1.3.3 集成性 1.3.4 Visual Studi 2010新功能 1.4 VS 2010演示与实战 1.4.1 VS 2010演示 1.4.2 VS实战 1.5 总结 第2章 TFS 2010安装场景 2.1 Team Foundation的体系结构 2.1.1 Team Foundation的逻辑结构 2.1.2 Team Foundation安装安全 2.1.3 Team Foundation安装的必备组件 2.2 单服务器安装 2.2.1 单服务器安装环境 2.2.2 单服务器应用场景 2.2.3 单服务器安装与配置 2.3 多服务器安装 2.3.1 多服务器安装环境 2.3.2 多服务器应用场景 2.3.3 多服务器的安装步骤 2.4 从TFS 2008 SP1升级 2.4.1 升级安装准备 2.4.2 升级安装应用场景 2.4.3 升级安装步骤 2.5 安装实验室管理 2.5.1 实验室管理安装环境 2.5.2 实验室管理应用场景 2.5.3 实验室管理配置 2.6 卸载及安装常见问题 2.6.1 卸载TFS 2.6.2 安装与配置问题 2.7 总结 第3章 MSF开发模式与流程 3.1 MSF体系概述 3.1.1 MSF简介 3.1.2 MSF组成原理 3.1.3 MSF与Visual Studio 2010的结合 3.2 MSF for Agile 5.0 (敏捷) 开发模式与流程 3.2.1 理解敏捷 (Agile) 3.2.2 基于Scrum的MSF敏捷解决方案 3.2.3 VS 2010敏捷开发的应用 3.3 MSF for CMMI改进开发模式与流程 3.3.1 MSF for CMMI简介 3.3.2 VS 2010中对于MSF For CMMI支持的新特性 3.4 总结 第4章 VS 2010团队协作管理 4.1 项目团队协作 4.1.1 团队模型及角色 4.1.2 团队成员利用VS 2010实现协同办公 4.1.3 通过Web方式访问团队项目 4.2 团队门户搭建 4.2.1 团队门户功能概要 4.2.2 团队门户搭建 4.3 TFS中的项目管理 4.3.1 项目过程模板 4.3.2 项目配置管理 4.4 总结 第5章 源码功能控制管理 5.1 TFS源码管理 5.1.1 源码管理使用场景及常用工具 5.1.2 Visual Studio 2010实现源码管理 5.1.3 VSS源码迁移到TFS 5.2 TFS源码管理设置 5.2.1 角色管理 5.2.2 安全管理 5.2.3 定义签入签出策略 5.3 源代码管理的高级功能 5.3.1 标签管理 5.3.2 分支管理 5.3.3 合并管理 5.4 总结 第6章 版本控制管理 6.1 TF构建概述 6.1.1 Team Foundation Build基础 6.1.2 Team Foundation Build架构 6.1.3 Microson Build引擎 6.1.4 Team Foundation Build的拓扑与安全性 6.2 构建TFS流程 6.2.1 同步源代码 6.2.2 编译应用程序 6.2.3 单元测试和集成测试 6.2.4 单元测试和集成测试 6.2.5 代码分析 6.2.6 发布构建 6.2.7 生成验证测试 (BVT) 6.2.8 发布构建报告 6.3 签入策略及签入文件 6.3.1 签入策略 6.3.2 签入文件 6.4 代码版本审查及搁置 6.4.1 代码版本审查 6.4.2 搁置的概念及用途 6.4.3 搁置的应用 6.5 确定文件冲突、合并及撤销 6.5.1 确定文件冲突及解决 6.5.2 合并 6.5.3 撤销 6.6 总结 第7章 Visual Studio 2010建模应用管理 7.1 Visual Studio 2010建模应用基础 7.1.1 软件系统建模 7.1.2 架构代码分析工具 7.1.3 有向图标记语言 (DGML) 7.1.4 依赖图 (Dependency Graph) 7.2 模型浏览器 (Model Explorer) 7.2.1 UML模型浏览器 (UML Model Explorer) 7.2.2 架构浏览器 (Architecture Explorer) 7.2.3 规格浏览器 (Spec Explorer) 7.3 开发软件设计模型 (Developing Models for Software Design) 7.3.1 架构过程概述 7.3.2 UML活动图 (UML Activity Diagrams) 7.3.3 UML类图 (UML Class Diagrams) 7.3.4 UML组件图 (UML Component Diagrams) 7.3.5 UML顺序图 (UML Sequence Diagrams) 7.3.6 UML用例图 (UML Use Case Diagrams) 7.3.7 层图表 (Layer Diagram) 7.4 开发与过程模型扩展 7.4.1 模型项目 (Modeling Project) 7.4.2 模型与工作项集成 7.4.3 用户需求模型 (Modeling User Requirements) 7.4.4 模型和图表扩展 (Extending Models and Diagrams) 7.5 总结 第8章 Visual Studio 2010测试管理 8.1 Visual Studio 2010测试体系结构 8.1.1 Visual Studio 2010测试概述 8.1.2 Visual Studio 2010测试框架 8.1.3 微软测试管理框架 (Microsoft Test Manager) 8.2 微软测试管理 (Microsoft Test Manager) 8.2.1 Visual Studio 2010与测试应用程序生命周期 (Lifecycle) 8.2.2 测试中心 (Testing Center) 8.2.3 实验室中心 (Lab Center) 8.2.4 网络实验室的配置与管理 8.3 Visual Studio 2010自动化测试 (Automated Tests) 8.3.1 自动化UI测试 (Coded UI Test) 8.3.2 单元测试 (UnitTest) 8.3.3 负载测试 (LoadTest) 8.4 Visual Studio 2010测试工具功能扩展 8.4.1 管理查询文件夹和安全 8.4.2 测试影响分析 8.4.3 门控

<<构筑敏捷的开发团队>>

式签入 8.4.4 滚动生成 8.4.5 微软测试用例迁移工具 8.5 总结 第9章 TFS报表管理与定制 9.1
报表分析 (TFS过程模板内建报表说明) 9.1.1 TFS内建报表概述 9.1.2 TFS数据仓库 9.1.3
Agile模板的常见报表 9.2 报表定制 (SQL Reporting Services定制项目报表) 9.2.1 TFS下的SQL
Reporting Services架构及报表生命周期 9.2.2 创建自定义报表 9.2.3 最佳实践:使用报表设计器
创建报表 9.3 报表管理 9.3.1 报表管理器 9.3.2 报表管理方法 9.3.3 管理报表服务器内容
9.4 报表的安全性设置 9.4.1 报表的权限设置 9.4.2 在TFS中添加团队成员 9.5 使用Excel生成
项目报表 9.5.1 在Excel中创建新表 9.5.2 在Excel中修改报表 9.5.3 在项目门户中管理Excel报
表 9.6 总结

<<构筑敏捷的开发团队>>

章节摘录

通过将SDLC工具集成到UI级别的表面，VS提高了团队工作效率并增强了项目的可预见性，VS的集成性主要体现在用户界面集成、数据集成、过程集成。

(1) 用户界面集成 工具间的无缝集成是增强生产效率的关键。

VS提供跨整个SDLC工具套件的、一致的用户体验。

对于开发人员而言，可以在他们当前的开发环境中使用某些活动（例如：单元测试、工作项跟踪、代码剖析以及代码分析）。

(2) 数据集成 通过使用一个公共的跨工具集的数据仓库，VS解决了在大多数SDLC工具中建立数据仓库这一问题，并启动了一个聚合的项目状态视图。

团队根据能够收集到的规则来管理项目。

今天，数据主要限于缺陷跟踪。

VS中集成的数据将开创一条新路——通过平衡利用贯穿于SDLC的大量、多样的数据来管理项目规则。

VS会收集精确的数据——不仅限于缺陷跟踪，而且包括测试结果、代码涵盖、代码生成、任务进度等这些贯穿于团队常规工作流程的数据。

该数据以某些标准的报告方式呈现，客户和微软内部团队已在报告中建立了项目管理的关键规则。

此外，团队还能够创建自定义报告，仅当在项目的范围内查看数据时，团队才能够精确地报告项目的状态。

(3) 过程集成 在VS中，工具行为由项目开始时选择的过程决定。

通过将过程与工具相集成，VS帮助确保在项目各阶段之间或各种项目角色之间避免丢失任何内容。

通过自动处理团队成员之间的工作调整，可以提高团队效率。

公司通过使过程标准化，能够纠正贯穿于过程改进中的系统问题，而无须增加团队的工作量。

此外，过程集成并不增加工作量，且通常能够降低与SDLC中所采纳过程相关的开销。

<<构筑敏捷的开发团队>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>