

<<领域驱动设计C# 2008实现>>

图书基本信息

书名：<<领域驱动设计C# 2008实现>>

13位ISBN编号：9787302221913

10位ISBN编号：730222191X

出版时间：2010-3

出版时间：清华大学

作者：麦卡锡

页数：374

译者：UML China

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>

## <<领域驱动设计C# 2008实现>>

### 前言

在阅读了Eric Evans的书Domain-Driven Design , Tackling Complexity in the Heart Of Software之后, 我设计软件系统的方式完全改变了。

之前, 我常常以一种侧重于以数据为中心设计软件对象模型, 我没有真正聚焦于如何把行为和数据结合到对象中。

我对这种新的思考方式感到如此震惊, 因此我开始试图找到能够揭示Eric书中概念的代码示例。

我通常用Google搜索我的领域驱动设计(DDD)问题的答案, 并且确实发现有些东西对我有帮助, 但我依然渴求更多关于这个主题的知识。

我不得不搜索.NET中的DDD答案, 因为Eric的书是与技术无关的。

本书的要点是架构概念。

里面到处有Java和Smalltalk的代码示例, 但不是关于架构概念的。

然后, 就有了Jimmy Nilsson的书Applying Domain-Driven Design and Patterns, 此时我开始看到可以用于连接DDD概念的更多模式。

Jimmy把来自Martin Fowler的优秀书籍Patterns of Enterprise Application Architecture的一些概念结合起来, 并说明它们如何有助于得到好的DDD设计原则。

Jimmy也做了大量的工作, 在他的书中提供了大量优秀的.NET代码示例, 并引导读者体验了几种按照DDD实现操作的方法。

就在我刚看完Jimmy的书后, 我开始订阅Yahoo!Groups上的DDD RSS Group feed, 这也给了我很多帮助。

在DDD小组中, 我发现的一件事情是人们一直在寻找一个能够揭示DDD原则的.NET相关应用。

在阅读了这些帖子之后, 我决定撰写这本书, 把我懂得的关于如何使用DDD技术构建.NET应用的知识分享给开发人员社群。

我猜我可能感觉有点内疚, 因为我读了小组里如此多其他人的帖子, 但我只是偶尔发帖。

现在, 我编写了一本书代替发帖!可能这将促使我更多地融入这个小组。

我写这本书的主要目的是吸取Eric、Martin和Jimmy书中的思想和模式, 构建一个真实的端到端.NET应用。

我确实希望说明我如何使用DDD原则在.NET中构建一个领域模型的某些想法, 但是, 我并非只是构建老的.NET应用, 我也要尝试一些Microsoft在构建应用方面的最新技术, 如Visual Studio 2008和.NET3.5框架。

**本书读者对象** 本书针对希望精通面向对象的设计技巧和学习DDD的有经验的.NET开发人员。

如果您不在这个级别, 也是可以的, 但我建议您至少有一些编写.NET代码或者Java代码的经验。

如果您之前没有编写过任何.NET代码, 则本书可能稍微有点难。

我也推荐您阅读之前提到的Eric Evans、Jimmy Nilsson和Martin Fowler的书籍。

您不一定要读, 但我强烈推荐阅读, 因为这有助于您更好地理解本书的许多设计和模式。

因为本书的每一章都建立在前一章的基础上, 所以我建议您按顺序阅读。

## <<领域驱动设计C# 2008实现>>

### 内容概要

这是第一本也是唯一一本关于使用C#实现领域驱动设计的技术书籍，本书介绍了构建实际应用系统的全过程。

本书的项目基于一个现有公司的实际系统，重点介绍了领域驱动设计和构建支持领域驱动设计的框架。

每一章都按照清晰的层次结构分成特定的模块，介绍了如何选择最佳的解决方案，然后采取相应的设计来解决问题。

所有章节构建了一个完整的项目，在创建系统代码的过程中您可以学到重要的实践技能。

本书主要内容 何时、为什么以及如何使用领域驱动设计 如何设计和构建初始的领域模型 如何实现“持久化无关” 如何为领域模型构建仓储框架 将TDD应用到领域模型的技能 如何应用模型—视图—视图模型模式 如何构建客户端会员管理系统 如何将客户系统和服务器同步 本书读者对象 本书面向有经验的C#和.NET开发人员，以进一步提升读者编写高性能和可扩展的应用程序的技能。

## 作者简介

Tim McCarthy是一名自由职业的咨询师，他采用最新的Microsoft平台和技术设计和构建分层的Web和智能客户系统。

Tim是一个解决方案架构方面的Microsoft MVP，他精通各种各样的Microsoft技术，不仅局限于以下技术：.NET Framework(ASP.NET / 智能客户NSTO / 工作流 / Web服务、Windows Presentation Foundation)、SQL Server、Active Directory、MS Exchange开发、UDDI、SharePoint和面向服务架构(SOA)应用。

Tim不仅是一个项目技术主管 / 成员，而且在一些财富500强公司担任技术咨询。

## &lt;&lt;领域驱动设计C# 2008实现&gt;&gt;

## 书籍目录

第1章 介绍项目：SmartGA系统	1.1 问题	1.2 设计	1.2.1 可靠性和可得性	1.2.2 伸缩性
	1.2.3 可维护性	1.2.4 富客户应用功能	1.2.5 离线可得	1.2.6 Web访问
	1.2.7 智能安装和自动更新功能	1.2.8 附加客户设备支持	1.3 解决方案	1.3.1 满足可靠性、可得性、伸缩性、离线可得和附加客户设备支持需求
	1.3.2 满足可维护性需求	1.3.3 满足富客户应用功能需求	1.3.4 满足Web访问需求	1.3.5 满足智能安装和自动更新功能需求
	1.4 总结	第2章 设计分层架构	2.1 问题	2.2 设计
	2.2.1 设计Visual Studio解决方案	2.2.2 设计基础设施层	2.3 解决方案	2.3.1 实现Visual Studio解决方案
	2.3.2 实现架构层	2.4 总结	第3章 管理项目	3.1 问题
	3.2 设计	3.2.1 设计领域模型	3.2.2 定义项目聚合	3.2.3 定义聚合边界
	3.2.4 设计仓储	3.2.5 编写单元测试	3.3 解决方案	3.3.1 Project类
	3.3.2 实现仓储	3.3.3 实现服务类	3.3.4 实现项目信息视图模型	3.3.5 实现项目信息视图
	3.4 总结	第4章 公司和联系人	4.1 问题	4.2 设计
	4.2.1 设计领域模型	4.2.2 定义Company聚合和Contact聚合	4.2.3 定义聚合边界	4.2.4 设计仓储
	4.2.5 编写单元测试	4.3 解决方案	4.3.1 Company类	4.3.2 Contact类
	4.3.3 ProjectContact类	4.3.4 实现仓储	4.3.5 实现服务类	4.3.6 Company视图模型
	4.3.7 Company视图	4.3.8 项目联系人视图模型	4.3.9 项目联系人视图	4.4 总结
	第5章 递交传送	5.1 问题	5.2 设计	5.2.1 设计领域模型
	5.2.2 定义Submittal聚合	5.2.3 定义聚合边界	5.2.4 设计仓储	5.2.5 编写单元测试
	5.3 解决方案	5.3.1 实现Submittal仓储	5.3.2 实现Submittal服务类	5.3.3 Submittal视图模型
	5.3.4 Submittal视图	5.4 总结	第6章 信息请求	6.1 问题
	6.2 设计	6.2.1 设计领域模型	6.2.2 定义RFI聚合	6.2.3 定义聚合边界
	6.2.4 设计仓储	6.2.5 编写单元测试	6.3 解决方案	6.3.1 实现RFI仓储
	6.3.2 实现RFI服务类	6.3.3 RFI视图模型类	6.3.4 RFI视图	6.4 总结
	第7章 提案请求	7.1 问题	7.2 设计	7.2.1 设计领域模型
	7.2.2 设计提案请求聚合	7.2.3 定义聚合边界	7.2.4 设计仓储	7.2.5 编写单元测试
	7.3 解决方案	7.3.1 提案请求类的私有字段和构造器	7.3.2 ProposalRequest属性	7.3.3 Validate方法
	7.3.4 实现提案请求仓储	7.3.5 实现提案请求服务类	7.3.6 提案请求视图模型类	7.3.7 提案请求视图
	7.4 总结	第8章 更改单	第9章 施工更改指令	第10章 和服务器同步
	第11章 客户会员管理系统			

<<领域驱动设计C# 2008实现>>

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>