<<.NET应用架构设计>>

图书基本信息

书名：<<.NET应用架构设计>>

13位ISBN编号：9787111365365

10位ISBN编号：7111365364

出版时间：2011-12-31

出版时间：机械工业出版社华章公司

作者：汪洋

<<.NET应用架构设计>>

内容概要

国内首本讲解企业级.net应用架构与设计的专著，由国内经验丰富的.net专家和架构师亲自执笔，权威性毋庸置疑。
本书的重点不在于架构与设计的理论，而是从实践的角度出发，结合大量示例和一个完整的项目案例循序渐进地讲解了.net应用架构与设计的方法、流程、原则、模式和最佳实践，实践指导意义极强。
本书在写作方式上打破了传统的知识灌输方式，而是用逐步演进的方式去引导和启发读者的抽象思维和宏观思想，从而让读者快速掌握架构与设计的精髓。

《.net应用架构设计：原则、模式与实践》一共分为三个部分：第一部分首先介绍了企业应用架构与设计的流程和核心概念，然后讲解了应用架构中常用的设计模式和设计原则，以及常用的几种设计方法；第二部分的主题是架构与设计的方法和最佳实践，既对架构分层的相关知识进行了详细的阐述，又用大量实战案例对业务层、服务层、数据访问层、数据存储层、显示层的原理和设计进行了深入的剖析；第三部分以一个真实的项目案例（已上线）演示了企业级应用的架构与设计的流程和方法，旨在帮助读者将前面所学的知识融会贯通，从而真正达到能动手实践的目的。

作者简介

汪洋(Richard Wang)，资深架构师和.NET技术专家。
一直战斗在一线，实践经验极其丰富。
现就职于惠普公司，担任技术架构师和信息分析师。
同时他还是上海益思研发管理咨询有限公司首席架构师、软件咨询组副组长。
曾在世界500强的在华日企担任架构师，负责项目业务框架和平台设计，曾带领创业团队与国内外企业进行项目合作。
51CTO、博客园、ITPUB、China Unix等IT社区的知名博客和技术专家。
CSDN和IrifoQ的特约技术作家。
2011年曾多次应邀在架构师大会、亚太软件研发管理峰会和中国软件开发大会等技术盛会上担任演讲嘉宾，深受与会者欢迎。
此外，他还在为国内多家软件企业开展软件架构与性能优化方面的培训。

<<.NET应用架构设计>>

书籍目录

前言
第一部分　架构与设计的原则和模式
第1章　架构与设计的流程和核心概念
1.1　正确认识软件架构
1.1.1　什么是架构
1.1.2　架构师的职责
1.1.3　架构设计
1.1.4　架构设计的优点
1.2　正确理解设计的含义
1.2.1　设计的重要性
1.2.2　架构和设计的关系
1.3　架构设计中的重要概念
1.3.1　tier和layer
1.3.2　架构与框架
1.3.3　架构与模式
1.4　本章小结
第2章　模式、设计原则和方法
2.1　设计原则与软件设计
2.1.1　设计原则简述
2.1.2　设计原则实战
2.2　设计模式
2.2.1　设计模式简介
2.2.2　正确使用设计模式
2.2.3　设计模式实战
2.3　企业应用架构模式
2.3.1　什么是企业架构
2.3.2　什么是企业级开发
2.3.3　再议架构设计和模式
2.3.4　企业架构模式介绍
2.4　企业架构模式实战
2.5　设计方法
2.5.1　测试驱动开发
2.5.2　领域驱动开发
2.5.3　行为驱动开发
2.6　本章小结
第二部分　架构与设计的方法和最佳实践
第3章　架构分层
3.1　反模式项目实例
3.2　分层设计
3.2.1　正确理解分层
3.2.2　常见的分层架构设计
3.2.3　n-tier架构
3.2.4　n-tier架构需要考虑的因素
3.3　从重构到分层
3.3.1　业务层设计实战

3.3.2　数据访问层设计实战
3.3.3　服务层设计实战
3.3.4　显示层设计实战
3.4　本章小结
第4章　业务层设计
4.1　业务层组织模式剖析
4.1.1　transaction script模式与实战
4.1.2　active record模式与实战
4.1.3　domain model模式与实战
4.1.4　anemic domain model模式与实战
4.1.5　业务层组织模式比较
4.2　业务层常用设计模式解析及实战
4.2.1　工厂方法模式
4.2.2　装饰者模式
4.2.3　模板方法模式
4.2.4　状态模式
4.2.5　策略模式
4.2.6　模板方法模式、状态模式、策略模式的比较
4.3　业务层常用的企业架构模式及实战
4.4　模式联合实战
4.4.1　需求规格模式
4.4.2　组合模式
4.5　业务层常用的设计原则及实战
4.6　本章小结
第5章　服务层设计
5.1　服务层详解
5.1.1　服务层的由来
5.1.2　服务层的职责是什么
5.1.3　服务层的必要性
5.2　服务层常用设计模式解析
5.2.1　外观模式
5.2.2　远程外观模式
5.2.3　数据传输对象模式
5.3　soa介绍
5.3.1　soa用途
5.3.2　soa原则
5.3.3　服务设计原则
5.3.4　服务解惑
5.3.5　服务操作设计原则
5.3.6　服务粒度介绍
5.4　服务层常用消息模式解析
5.4.1　文档消息模式与请求-响应模式
5.4.2　预约保留模式
5.4.3　等幂模式
5.5　soa实战——wcf实现
5.5.1　系统分析
5.5.2　业务层实战

5.5.3　数据层实战
5.5.4　服务层实战
5.5.5　代理层实战
5.5.6　显示层实战
5.6　本章小结
第6章　数据访问层设计
6.1　数据访问层简述
6.2　数据访问层的设计策略
6.2.1　仓储模式
6.2.2　数据访问对象模式
6.3　数据访问层常用模式与原则解析
6.3.1　工作单元
6.3.2　标识映射
6.3.3　延迟加载
6.3.4　数据并发控制
6.3.5　查询对象
6.4　orm对象关系映射
6.4.1　nhibernate介绍
6.4.2　entity framework介绍
6.5　企业级领域驱动设计项目实战
6.5.1　业务层的实现
6.5.2　服务层的实现
6.5.3　数据访问层的实现
6.5.4　显示层的实现
6.6　本章小结
第7章　数据存储层设计
7.1　合理选择数据存储方案
7.1.1　数据存储的三种方式
7.1.2　选择合理的数据存储方案
7.2　数据库架构设计
7.2.1　分布数据
7.2.2　数据拆分
7.2.3　读写分离
7.2.4　数据缓存
7.3　数据库设计
7.3.1　数据库计划
7.3.2　逻辑数据库设计
7.3.3　物理数据库设计
7.4　sql server数据库性能瓶颈分析与解决方案
7.4.1　缺失索引的瓶颈分析与解决方案
7.4.2　昂贵查询的瓶颈分析与解决方案
7.4.3　数据库碎片的瓶颈分析与解决方案
7.5　本章小结
第8章　显示层设计
8.1　mvp模式的原理与实战
8.1.1　mvp模式实战（asp.net实现）
8.1.2　mvp模式经验谈

8.1.3　mvp模式之高级话题
8.2　前端控制器模式的原理与实战
8.2.1　前端控制器模式介绍
8.2.2　深入浅出命令模式
8.2.3　前端控制器模式实战
8.3　mvc模式的原理与实战
8.3.1　深入浅出mvc模式
8.3.2　mvc模式之高级话题
8.3.3　mvc模式实战——对asp.net mvc源码进行分析
8.4　pm模式的原理与实战
8.4.1　pm模式的解析
8.4.2　pm模式实战
8.5　mvvm模式的原理与实战
8.5.1　mvvm模式介绍
8.5.2　mvvm模式深度剖析
8.5.3　mvvm模式高级话题
8.6　本章小结
第三部分　.net应用的架构与设计实战
第9章　it创业产品互推平台的项目背景与功能介绍
9.1　it创业产品互推平台背景
9.2　sns功能介绍
9.2.1　用户管理
9.2.2　个人信息管理
9.2.3　软件展示功能
9.2.4　好友功能
9.2.5　站内信息功能
9.2.6　多媒体文件管理
9.2.7　博客
9.2.8　用户群
9.2.9　微博
9.2.10　共享功能
9.2.11　论坛
9.2.12　活动
9.2.13　投票
9.2.14　页面布局定制
9.2.15　评级
9.3　本章小结
第10章　it创业产品互推平台架构设计
10.1　分层设计
10.1.1　逻辑分层的总体设计
10.1.2　通用功能设计
10.1.3　业务层设计
10.1.4　数据持久层设计
10.1.5　显示层设计
10.2　部署设计
10.2.1　单服务器部署
10.2.2　多服务器部署

10.3　文件存储的设计
10.4　基础类库的设计
10.4.1　缓存
10.4.2　配置读取
10.4.3　邮件发送
10.4.4　日志记录
10.4.5　辅助类的实现
10.5　本章小结
第11章　it创业产品互推平台用户管理
11.1　用户管理功能分析
11.1.1　用户注册
11.1.2　用户登录
11.1.3　找回密码
11.1.4　激活账户
11.1.5　删除用户
11.1.6　更新用户
11.1.7　查询用户
11.2　用户管理接口的定义
11.2.1　服务层实现定义
11.2.2　业务模型定义
11.2.3　数据访问层的接口定义
11.3　用户管理的实现
11.3.1　服务层实现
11.3.2　业务层实现
11.3.3　数据访问层实现
11.3.4　显示层实现
11.4　本章小结
附录a

章节摘录

版权页：插图：1.界面层界面层通常指的是用户界面或表现层。
此处我使用了一个比较通用的术语——界面，因为该架构不仅支持没有任何用户的面向服务的应用程序，也支持WPF、Web窗体、Windows窗体，以及其他有用户的应用程序类型。
可能会有读者疑惑，为什么我把界面层和界面控制层分开来介绍（一般把界面层和界面控制层综合在一起，统称为“显示层”）？
当然，从智能客户端的角度来看，界面和对界面的控制是相同的，因为它们就是用户能够与系统交互的图形化用户界面（GUI）窗体。
但是从Web的角度来看，它们的区别就比较明显了。
通常浏览器只为用户提供了一个界面，用来显示数据并收集用户的输入内容。
在这种情况下，所有真正的交互逻辑，也就是用来控制界面、生成输出或解释用户输入的代码，都运行在Web服务器（或者大型机）上而不是客户端计算机上。
当然，现在的浏览器上可能会运行Ajax或Silverlight。
不过，它们必须被视为分离的应用程序，因为它们只是与运行在服务器上的应用程序交互。
所以即便在浏览器上有运行代码，应用程序的界面代码还是运行在Web服务器上的。
现在界面技术的种类越来越多，并且每种技术都会带来一些新的相对不兼容的技术，我们必须为适应它们而做一些工作，这就需要我们自己动手创建了。
同时，我们应该专注于简化架构中的其他层，因为相对而言，其他层的设计更加有规则可以遵循，并且会带来巨大的效益。
2.界面控制层 既然前面已经提出了界面层和界面控制层之间的区别，那么界面控制层的目的就比较清楚了。
该层包括以下几种逻辑：决定用户应该看到什么，对路径进行导航，以及解释用户的输入。
在WPF或Windows窗体的应用程序中，这些逻辑指代窗体后台的代码。
在Web窗体应用程序中，这些逻辑不仅仅指代窗体后台的代码，还可以包含在服务器端控件中的代码。
在很多应用程序中，界面控制代码非常复杂。
对于初学者来说，它必须以非线性的方式对用户的请求做出响应。
（因为很难预料用户会怎样点击控件，或者进入或离开窗体和页面。
）界面控制代码还必须与业务层的逻辑交互，以验证用户输入、处理用户的请求，或者做任何其他与业务相关的动作。

媒体关注与评论

对于程序员来说，架构师是他们职业生涯中的一个重要追求目标。
由于架构师不仅需要非常高的综合素质，而且还需要丰富的实战经验，因此行业内真正的架构师少之又少。
对于企业而言，一位优秀的架构师就好比疆场上指挥百万大军的将军，运筹帷幄，决胜千里之外，一个应用是否具有高可用性、高扩展性和良好的性能，架构9币起着决定性作用。
本书内容是作者在大量企业级，NET项目中总结出来的宝贵经验，以实战为导向，系统地讲解了企业级应用架构与设计的流程、方法、模式、原则与最佳实践，适合希望成为优秀架构师的读者系统学习。
强烈推荐！
——51CTO本书是作者多年架构实战经验的结晶。
实践性强，通过大量的实际项目案例详细讲解了.NET应用的架构与设计的方法；内容全面，从架构设计的核心概念和原理，到架构设计的模式与最佳实践，全部包含其中：阅读体验好，语言朴实，幽默风趣，能让你在轻松的阅读中领悟到架构设计的精髓。
——China Unix社区＆ITpub社区联袂推荐有多少项目，因为设计的糟糕，做到最后，让人苦不堪言。
为什么会这样？
缺少合理的设计原则和必要的实践。
有多少项目，因为架构的缺陷，导致整个项目最后陷入“泥沼”。
为什么会这样？
缺少周密的思考和对技术的深度把握。
又有多少项目，因为不切实际的假设和过度的设计，致使开发进入“死亡行军”，让人叫苦不迭。
为什么会这样？
缺少经验的积累和失败的教训。
在这些“悲剧”的背后，我们不应该无休止地悲伤；在这些失败的项目面前，我们需要更深刻地反思。
在这本书中，作者通过自己的实战经验给出了这些问题的答案，既有可供借鉴的成功经验，也有需要大家引以为戒的失败教训，值得仔细品读。
——马伟微软ASP.NET MVP，畅销书《ASP.NET 4权威指南》作者

<<.NET应用架构设计>>

编辑推荐

<<.NET应用架构设计>>

版权说明