

<<ASP.NET MVC 1.0高级编程>>

图书基本信息

书名：<<ASP.NET MVC 1.0高级编程>>

13位ISBN编号：9787302222439

10位ISBN编号：7302222436

出版时间：2010-3

出版时间：清华大学出版社

作者：(美)康纳利 等著, 冯飞 译

页数：422

字数：663000

译者：冯飞

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>

前言

为什么这世上还需要另一个Web架构？

这个问题可能一直就在读者的脑子里盘旋，也可能是当有人看到本书放在书架上面时会有这一疑问。

在过去的几年里，本书的几个作者也多次思考这一问题。

确实，现在市面上有很多架构在行业上使用，而且充满了各种噱头。

简而言之，很容易让人产生怀疑。

就像本书的作者一样，在较深入地研究了最新最好的Web架构之后，他们都不约而同地意识到在过去的10年里这一行业已经走得很远。

早在1997年，Rob就开始使用Classic ASP来为Web编程，曾一度沉浸于刺激和兴奋中。

随着.NET的出现，他记得他曾经跑出自己的办公室，让所有人停下手头的工作，然后向他们解释到：世界绕着他的轴倾斜了。

作者对于ASENETMVC也有相同的感受。

这不是因为它“与众不同”，而是因为它向开发人员提供了“按照自己的方式工作”的机会。

不喜欢平台提供的视图？

那么修改它。

ASENET MVC架构几乎每个部分都是“可互换的”——就好比，如果嫌鞋子小的话，就换一双。

不喜欢领带？

为什么不用领结呢？

一切尽在掌握中。

ASENETMVC是一个有着大量约定的Web架构。

如果遵循这些约定，您的工作会变得很简单；但是如果不需要这些约定，那么您可以迅速脱离该架构，按照自己想要的方式来完成工作。

虽然本书探讨的是关于ASENETMVC所带来的“独特”体验，但是更为重要的是，读者将会学到通过自己的方式来拓展ASENETMVC的更实用的方法——然后，读者可能会希望与其他人共享这些技巧。

由于其可扩充性以及关注于“按自己的方式工作”，所以本书的作者很高兴地接受了这个Web架构，并希望读者与他们同行。

内容概要

ASP.NET MVC架构是以一些核心原则(可扩展性、可测试性和关注点分离)为指导从头设计而成的。该架构添加了各种约定以帮助开发人员走向成功，并且提供了适合于Web工作方式的流线型开发体验。

对于想剥离抽象层并接近事物实质的开发人员来说，ASP.NET MVC就很合适。对于特别关心架构如何组合在一起的开发人员来说，ASP.NET MVC也极具扩展性，几乎允许对其任何部分进行定制，甚至可按个人喜好整体替换。

本书由ASP.NET小组成员编写，Scott Guthrie首先提供了一个说明如何构建应用程序的完整演示，然后深入介绍MVC的基本概念和历史，并快速过渡到讨论ASP.NET MVC模式如何实现这些概念。在学习本书的过程中，您将理解在从传统的ASP.NET Web Forms转移到ASP.NET MVC时所需的思维转换以及一旦转移后所获得的收益。

作者简介

Rob Conery, 在Microsoft的ASP.NET小组中工作。他是SubSonic的创建者, 也是Commerce Starter Kit (一个免费的.NET开放源代码的电子商务平台) 的主要设计师。

他和他的妻子以及两个) L (Maddy和Ruby) 住在夏威夷的考艾岛。

Scott Guthrie, 是Microsoft的.NET Developer Division的副总裁。他所率领的开发小组主要负责交付Microsoft Visual Studio开发工具以及用于构建客户端和Web应用程序的Microsoft.NET架构技术。

作为.NET项目的主创人员, Guthrie自1999年以来就一直在 VisualStudio和.NET架构的设计和开发中扮演着非常重要的角色。

Guthrie还负责Microsoft的web服务器平台和开发工具小组的工作。

最近, 他还促成了Silverlight的开发, silverlight是一款跨浏览器, 是用来交付下一代媒体经验以及web上丰富的Internet应用程序的跨平台插件。

现在, Guthrie直接管理的开发小组有很多, 这些小组分别负责构建公共语言运行库 (CLR)

、ASENET、Silverlight、Windows Presentation Foundation (WPF)、IIS、CommerceServer以及用于Web、客户端和Silverlight开发的Visual Studio Tools。

Guthrie毕业于杜克大学的计算机专业。

Phil Haack, 是ASP.NET小组的高级编程经理, 主要负责ASENET MVC项目。

在加入Microsoft之前, Phil Haack曾是一个代码搜索引擎的产品经理、一家网络游戏公司的开发经理, 也曾是流行西班牙语电视网络的高级设计师。

作为一名编码的狂热者, Phil Haack喜欢制作软件。

他不仅仅喜欢编写软件, 而且热衷于在他的博客 ([http: haacked.com](http://haacked.com)) 上撰写关于软件和软件管理的内容。

在闲暇时间, Phil Haack向各种开放源代码的项目投稿, 他是Subtext博客引擎项目的创建人, 这个项目当然也正在使用ASENET MVC来重写。

书籍目录

第1章 NerdDinner第2章 MVC与ASP.NET第3章 ASP.NET MVC第4章 路由和URL第5章 控制器第6章 视图第7章 AJAX第8章 过滤器第9章 保护应用程序的安全第10章 使用ASP.NET MVC进行测试驱动开发第11章 可测试的设计模式第12章 将Web Forms和MVC相结合

章节摘录

在开始深入讨论测试驱动开发（Test Driven Development，TDD）之前，本书作者将先澄清一件事情。

ASENET MVC不只是提供给那些实践测试驱动开发的人使用的。

因此，如果您不打算实践TDD且计划跳过本章，那么请稍等片刻。

本书并没有试图劝服读者转为使用TDD或为其布道（虽然Phil可能在街角遇到您的时候会这样劝服您），但是这里提供了一个机会来解释为什么Microsoft努力让该架构对TDD迷们很友好，即使您不喜欢TDD，但是这一努力确实会对您有益。

那么，当讨论ASENET MVC时，为什么都集中在TDD上？

为了理解这一答案，可以从历史的角度来理解，这将是很有帮助的。

在一开始设计ASP.NET时，开发人员没有时刻牢记TDD的思想。

当时，TDD还处于其最初的发展阶段，没有像今天这样广为采用。

因此，因为ASENET的很多领域都与其他子系统紧密耦合在一起，所以它们向那些为利用这些领域的开发人员编写自动化单元测试的人带来了挑战。

牢记可测试性而设计出来的架构要比只能够编写单元测试带来更多的好处。

因为编写正确的单元测试通常需要隔离架构的片断并交换架构用于双重测试（完全处于掌控下的接口的虚假实现）的其他依赖性，所以这种架构作为其可测试的副产品，极具可扩充性。

事实证明，就可测试性而言，TDD专业人士具有很高的需求。

TDD是一种代码设计的动作，它生成单元测试，因此需要内在的架构是天生可测试的。

虽然处理不可测试领域的技术也存在，但是如果一个架构要求太多这样的工作区，则将导致与TDD专业人士之间产生很多摩擦。

在这种情况下，这种摩擦往往让人们感到很不愉快。

如果ASENET MVC架构可以将这种摩擦降至最低，那么这不仅是TDD专业人士将从中受益，而且所有需要使用和扩充架构的开发人员都将从中受益。

这就是为什么会产生所有围绕TDD和ASENET MVC的忙乱。

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>