

MANNING

# ASP.NET MVC 实战

## ASP.NET MVC IN ACTION

Jeffrey Palermo  
〔美〕 Ben Scheirman 著  
Jimmy Boggard

姚 军 译



人民邮电出版社  
POSTS & TELECOM PRESS

 MANNING

# ASP.NET MVC 实战

ASP.NET  
MVC  
IN ACTION

Jeffrey Palermo  
美) Ben Scheirman 著  
Jimmy Boggard

姚 军 译

人 民 邮 电 出 版 社  
北 京

## 图书在版编目 (C I P) 数据

ASP.NET MVC 实战 / (美) 巴勒莫 (Palermo, J.),  
(美) 史凯尔曼 (Scheirman, B.), (美) 博加德  
(Boggard, J.) 著; 姚军译. — 北京: 人民邮电出版社,  
2010. 12  
ISBN 978-7-115-23775-0

I. ①A… II. ①巴… ②史… ③博… ④姚… III. ①  
主页制作—程序设计 IV. ①TP393.092

中国版本图书馆CIP数据核字(2010)第177552号

## 版权声明

*Original English language edition published by Manning publications Co., 209 Bruce Park Avenue, Greenwich, Connecticut 06830. Copyright (c) 2007 by Manning Publications Co.. Simplified Chinese-language edition copyright (c) 2010 by Posts & Telecom Press. All rights reserved.*

本书中文简体字版由美国Manning出版公司授权人民邮电出版社出版。未经出版者书面许可,对本书任何部分不得以任何方式复制或抄袭。

版权所有,侵权必究。

## ASP.NET MVC 实战

- 
- ◆ 著 [美] Jeffrey Palermo Ben Scheirman Jimmy Boggard  
译 姚 军  
责任编辑 汪 振
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京艺辉印刷有限公司印刷
  - ◆ 开本: 800×1000 1/16  
印张: 21.5  
字数: 492千字 2010年12月第1版  
印数: 1-3500册 2010年12月北京第1次印刷

著作权合同登记号 图字: 01-2009-3791号

ISBN 978-7-115-23775-0

定价: 49.00元

读者服务热线: (010)67132705 印装质量热线: (010)67129223  
反盗版热线: (010)67171154

# 内 容 提 要

MVC作为一种流行的应用程序开发框架已经存在多年。现在，Microsoft公司也加入了这个社区，ASP.NET MVC框架为应用程序开发人员提供利用成熟开发模式，开发易于理解、易于测试和易于维护的应用程序的新利器。本书通过一个已经投产的代码营网站实例，介绍了这个框架的方方面面，贯穿本书始终的是MVC框架的特性及其重要的设计理念（关注点分离、依赖注入等），一个完整的开发过程（包括建立模型、编码和测试）为开发人员提供了熟悉这一新框架的极好示范，同时揭示了与成熟的MVC框架（如Ruby on rails等）的比较以及互相借鉴之处。

本书适合于对MVC框架感兴趣的ASP.NET开发人员，其中的许多设计原则和最佳实践则可以适合于各种开发人员。

# 序

ASP.NET MVC 1.0的最终版本在2009年3月份的Mix09会议期间发布了,没有人惊讶于它的内容——这是件好事。在这个最终版本登场之前,产品团队已经发布了带有完整源代码的多个公开预演版本,以努力提升Microsoft产品的开放性和社区参与度。

我们为什么这么做?

透明和社区参与是高尚的目标,但是它们并不是一个项目的最终目标。我们真正追求的是一件伟大的产品。我愿意将ASP.NET MVC看作一个展示透明和社区参与是达到这一目标的主要方法的试验。

在ASP.NET MVC的Preview 2版本发布之后,我们从开发人员那里得到反馈——使用ASP.NET MVC编写单元测试很困难。本书的首席作者Jeffrey Palermo在这一时期是提供反馈的主要人物之一。我们采纳了这一意见,并且通过引入动作结果的概念实施了一个主要的API更改,这比我们之前的设计要好得多。社区参与帮助我们创建一个更好的产品。

ASP.NET MVC聚焦于固定的原则,如用于提供非常可扩展和可测试的框架的关注度分离。虽然在你觉得合适时可以修改源代码,但是该框架用于开放给不需要修改源代码的扩展。该框架的任何部分都可以交换为你选择的其他东西。不喜欢视图引擎?尝试Spark视图引擎。不喜欢我们实例化控制器的方式?使用你自己的依赖注入容器。

ASP.NET MVC还包含了很棒的工具,如Add View(添加视图)对话框,它使用代码生成快速地创建基于模型对象的视图。最好的部分是ASP.NET MVC中的代码生成特性依赖T4模板从而可以完全定制。

在这本书中,Jeffrey将分享所有这些特性,并且展示将它们组合到一起创建一个了不起的应用程序的方法。我希望你享受这本书并且分享他创建Web应用程序的激情。记住,本书不仅是邀请你学习ASP.NET MVC,还邀请你加入社区并且影响ASP.NET MVC的未来。快乐地编程吧!

PHIL HAACK

微软公司

ASP.NET MVC团队

高级程序经理



# 作者序

我的职业生涯从20世纪90年代中期作为早期Web开发人员开始。Web就是HTTP。Netscape Navigator帮助了具有Internet modem的家庭数量的增长，因为它比同时代的其他软件都要先进。Netscape Navigator 3.0（1996）和3.04（1997）帮助全世界的家庭和公司将Internet用于日常事务。没有比购物更常用的任务了！随着电子商务的出现，Internet在资本的作用下爆炸性地增长了。

我在公共部门开始了Web开发，在那里我们通过使学区的毕业生能够与之前的同学协作建立了第一个社交网络。我使用IDC（Internet数据库连接）和HTX（HTML扩展模板）在Microsoft平台上开始了职业生涯。Internet信息服务（IIS）2.0给了我们使用ODBC数据源的神奇灵活性。这是我第一次使用“代码金块”，或者<% %>定界符。IDC/HTX为活动服务器页面（ASP）让路，我仍然能回忆起那时的变化——从ASP 2.0到ASP 3.0以及令人生畏的COM+集成。我涉猎了CGI、Perl、Java和C++，但是坚持使用Microsoft平台。从局外人的观点观察Visual Basic的蓬勃发展，我用小的实用程序摸到了窍门。

Active Server Pages 3.0的创作人员看到了与Windows 95一起发布的Internet Explorer 4和Netscape竞争市场份额的浏览器大战。编写与这两个浏览器都能很好地协作的Web应用程序很困难。IE 5.0用像XML数据岛和更好的脚本能力开启了Intranet应用程序的新视野。Windows XP带有IE 6，它毫不费力地占有了Web浏览器的大部分市场。ASP 3.0使编程人员密切地接触到HTTP、HTML以及GET和POST动词。我记得拔除粗糙的框架以处理来自相同ASP脚本的多个请求路径的情景。

在ASP 3.0被广泛采用的同时，Struts掀起了Java Web应用世界的一场风暴。Struts可能是最著名的Java MVC框架，尽管现在已经有了许多流行的JVM框架。使用ASP 3.0，我没有注意到我对应的Java产品已经吸取了教训，但是我感受到了在一个单独的ASP脚本中集中大量任务的痛苦。

我从一开始就使用了ASP.NET 1.0并且将我的一些ASP 3.0网站转换为Web Form。记得GridLayout是默认工具，到处都是CSS绝对定位的时候吗？很明显，Web Forms 1.0是VB 6开发人员转到.NET并接触到Web的桥梁。回传和按钮点击句柄同我没什么关系，但是我的那些对VB 6经验丰富的同伴们对此感到很亲切。ASP.NET 1.1抛弃了GridLayout并强迫开发人员理解HTML以及流布局的工作方法。当Internet Explorer是“首选”浏览器时，downlevel呈现是很棒的，而在其他浏览器中都降级了。这种模式在Firefox提升了市场份额并且提出遵循标准的标示时被打破了。

我成为了一个ASP.NET专家并且在.NET 2.0 beta测试期间频繁地在博客上提出意见。我知道从ASP.NET 1.1到ASP.NET 2.0的每个特性和每个重要的更改，并且帮助我的团队采用2.0。在ASP.NET 2.0的时代，我开始跟从Martin Fowler和他的模型—视图—表现著作。我实现了的模式将逻辑从变得浮肿的代码隐藏文件中剥离的模式。在2005年，Java开发人员享受用于Web的多种

MVC框架的一个选择，而我正奋力地将Web Forms转换为模型—视图—表现模式和测试驱动的开发建议。这令人疲惫不堪，但是还有什么替代的方案？

在2006年，由于工作的变化，我转向了软件管理和WinForms客户开发。因为熟悉了代码隐藏模式的沉闷，并且需要管理一个开发团队，我以WinForm类为视图实现了模型—视图—控制器模式。这令人耳目一新。UI开发是无缝的，并且控制器成为了领域模型到UI的自然边界。在2007年，我回到了Web开发并且勉强地再次使用Web Forms来实现模型—视图—表现器。回想过去，我希望自己已经采用了另一个用于.NET的模型—视图—控制器框架。

在2007年2月，Scott Guthrie (ScottGu) 创建了ASP.NET MVC框架的一个原型。他从许多客户那里听到了关于Web Forms的困难，以及对更简单、更灵活的编写Web应用程序方法的需求。在2007 MVP高层会议上，Scott从一个Microsoft MVP小组那里得到了信息。Darrell Norton、Scott Bellware、Jeremy Miller和我验证了他的原型的构想，并且给出了一些最终进入该框架代码的原始信息。

当Scott Guthrie在2007年8月于美国德克萨斯州Austin召开的AltNetConf开放空间会议上将一个ASP.NET MVC的工作原型和构想介绍给听众时，我立刻知道了这是我自始至终希望的。作为一个长期的Web开发人员，我理解HTTP和HTML，而且相信这是ASP.NET 1.0应该做到的。从ASP 3.0到ASP.NET MVC的迁移是非常顺利的。我可以自称投产了第一个ASP.NET MVC应用程序，因为我说服Scott给了我一个他的原型的副本，并且改进了我的www.partywithpalermo.com注册网站，2007年11月在DashPoint上的一个Rod Paddock服务器上启用它。

Microsoft在ASP.NET MVC发行周期中所做的一切是开发部门中史无前例的项目。该项目至少每季度在CodePlex网站上发布一次，包括源代码。它也使用测试驱动开发作为软件构造技术来进行开发。源代码下载中包含了完整的单元测试覆盖，ASP.NET MVC 1.0在MS-PL和OSI认可的开源许可下发行。

ASP.NET MVC以Web的工作方式进行工作。它天生适合这个工作，尽管Microsoft公司是最后一个为其开发平台提供模型—视图—控制器框架的，但是这个框架很强大。它的设计首先聚焦于核心抽象理念。这有助于社区对其进行扩展。实际上，在第一个社区技术预览 (CTP) 版本发行的同一周，Eric Hexter和我就启动了MvcContrib开源项目，提供了与ASP.NET MVC框架集成的第一个扩展。

在本书出版的时候，ASP.NET MVC框架在Headspring Systems已经是被频繁使用的工具了，这减轻了我的顾问工作量。对于整个.NET业界，我预测ASP.NET MVC将在2011年被认定为ASP.NET开发的规范。

每天都有新的开发人员来到.NET平台，对于Web开发人员来说，ASP.NET MVC易于采用和学习。因为复杂度的降低，所以采用该平台的障碍减少了；因为简单，所以它可以发展到符合最复杂企业系统的要求。

当Manning出版公司邀请我编写一本关于ASP.NET MVC的书籍时，我已经是这个主题上的活跃博客作者，并且已经在CoDe杂志上发表了一篇关于这个框架的文章。尽管如此，我知道编写一本书籍将是巨大的挑战。本书的编写过程超过了一年，我因为它的出版而兴奋。在这个项目中，我从Ben和Jimmy那里学到了很多，而且通过编写关于这个框架的书籍，我学到了更多关于它的

知识。这些知识对于我们的客户项目有着直接的好处。

我们希望这本书在你编写了自己的第一个应用程序之后仍然伴随着你。在1.0版本发行之后就出版一本书是个挑战，因为在一种技术曝光之后人们会发现很多情况。在客户项目上立刻使用它绝对能够帮助提升包含在本书中的信息的质量，因为这来自于第一手经验。

虽然其他的平台已经从模型—视图—控制器框架中得益多年，但是对于许多.NET开发人员来说MVC模式仍然是陌生的。本书解释使用该框架的方法和时机，还有在该模式背后的理论和原则，以及得到了好评的模式。我们希望本书将能启发你理解一种不可或缺而易学的技术。

JEFFREY PALERMO



# 致 谢

我们要感谢Scott Guthrie发现了这一框架在.NET空间中的需求。没有它的原型、构想和领导，这一框架将仍然不能出现在.NET框架之中。我们还要感谢Phil Haack（ASP.NET MVC程序经理）领导的Microsoft ASP.NET MVC团队。ASP.NET MVC团队的其他关键成员是Eilon Lipton（首席开发人员）、Levi Broderick（开发人员）、Jacques Eloff（开发人员）、Carl Dacosta（QA）和Federico Silva Armas（首席QA）。我们还要感谢大量支持Microsoft的工作于包装、文档和交付ASP.NET MVC框架的人们。尽管这个框架相对于其他来说很小，但是Microsoft的这一步改变了软件界的.NET部分的惯性思维。

参予本书写作的3位作者，都是多个项目的顾问，Dave Verwer还帮助本项目的启动并且草拟了一章。从ASP.NET MVC框架的第一次社区技术预演开始，本书历时一年半。这一活动得到了Manning出版公司人员的大力支持。我们感谢他们自始至终的耐心和支持。我们要特别感谢采编Michael Stephens和编辑Tom Cirtin的指导。Michael看到了本书的需求并且与我接洽。Tom在我们三人的第一本出版物编写期间一直耐心地帮助我们。

我们的独立技术评审非常出色。他们在此期间为每一章提供了建议和独特的观点，没有这些信息，这本书不会像我们希望的那么好。我们真诚地感谢Phil Haack审阅手稿并且编写了精彩的前言。同样应该感谢Freedom Dumlao，他辛勤地审阅每一章以确保信息以最好的方式提供给目标读者。Jeremy Skinner也是对这个项目的恩赐。他一再测试了书中每个代码清单和代码实例以及本书所带的许多Visual Studio项目。他对细节的注意，加上他在ASP.NET MVC和MvcContrib上的丰富经验，对本书是极大的贡献。

Manning邀请了以下评审在开发的不同阶段阅读手稿并提出他们的建议：Mark Monster、Andrew Siemer、Benjamin Day、Frank Wang、Derek Jackson、Tim Binkley-Jones、Marc Gravell、Alessandro Gallo、Josh Heyer、Peter Johnson、Jeremy Anderson和Alex Thissen。

本书还从志愿阅读部分手稿并提供反馈的外部技术评审那里获益：Rod Paddock、Craig Shoemaker、Hamilton Verissimo、Matt Hinze、Kevin Hurwitz、Blake Caraway、Nick Becker、Mahendra Mavani、Eric Anderson、Rafael Torres、Eric Hexter、Tom Jaeschke、Matt Hawley和Sebastien Lambla。

在本书付印之前，许多人通过参加Manning的早期访问程序（MEAP）订购了PDF版本。我们也要感谢这些读者在早期以及对整个手稿部分的评论和合作，特别是Eric Kinateder、Ben Mills、Peter Kellner、Jeff P.、Orlando Agostinho、Liam McLennan、Ronald Wildenberg、Max Fraser、Gudmundur.Hreidarsson、Kyle Szklenski、Philippe Vialatte、Lars Zeb、Marc Gravell、Cody Skidmore、Mark Fowler、Joey Beninghove、Shadi Mari、Simone Chiaretta、Jay Smith、Jeff Kwak和Mohammad

Azam。

**JEFFREY PALERMO**

我要感谢我美丽的妻子Liana在本书写作过程中的支持和耐心。Liana在本书开始写作之前刚刚生下我们的女儿Gwyneth Rose，花费更多的时间和我成长中的家庭在一起的动力驱使我完成本书。还要感谢我的父母Peter和Rosemary Palermo从小灌输给我的对书籍和学习的热爱。

**BEN SCHEIRMAN**

我首先要感谢我了不起的妻子Silvia。她对我的持续支持和鼓励促成了本书的写作。我还要感谢我在大学的导师Venkat Subramaniam。由于他的指导，我发现自己在软件开发上的激情并且坚持学习和挑战极限。他是我职业生涯的灵感源泉。最后我要感谢我出色的孩子们Andréa、Noah和Ethan（最近还有Isaac和Isabella），他们在父亲深夜还在敲击键盘时表现出了巨大的宽容和鼓励。

**JIMMY BOGARD**

感谢我的妻子Sara，没有她的爱、支持和宽容，我对这本书不会有任何贡献。我还要感谢我的家庭在这些年来对我这个奇怪的小书虫的容忍。最后，感谢我的高中计算机科学教师Scotty Johnson，他向我展示了这种技术所能带来的真正的激情。

# 前 言

ASP.NET MVC框架是Scott Guthrie在2007年初的一个构想。随着2007年后期展示了一个原型以及对Phil Haack为部件团队高级编程经理的关键任命，Scott使这一构想变成现实。每当.NET社区因为其他平台有类似Tapestry、Rails等杰出的MVC框架而感到沮丧时，Web Forms随着开发人员努力使其完成它在2001年推出时无法想象的任务而逐渐失宠。Castle MonoRail是很强大的框架并且继续具有领先性，但是更广阔的.NET业界需要从Web Forms中有所改变。Phil Haack以他在Microsoft之外和在开源社区的经验，很快领导ASP.NET MVC框架团队推出了令.NET社区欢欣鼓舞的成功的1.0版本。

ASP.NET MVC从其他MVC框架（如Struts、WebWork、Tapestry、Rails和MonoRail）的经验教训中得益。它还出现在C#开始抛弃其完全静态类型的根基的时候。.NET 3.5引入的这一预言改进在ASP.NET MVC中得到完全的利用，使其大大优于前面出现的框架，以及与当前支持的Java语法绑定的所有Java框架。

对于有多种软件背景的人来说，ASP.NET MVC对于Visual Studio开发体验是个很好的补充。对于以.NET 1.0或更晚版本开始软件生涯的人们来说，这是对思路的根本改变，因为他们是同以Web Forms为“规范”的Web开发一起成长起来的。

本书试图从超出文档和ASP.NET MVC网站（<http://www.asp.net/mvc/>）上的联机丛书的位置开始。如果你刚刚开始使用ASP.NET，那么你将希望阅读一些介绍ASP.NET管道和服务器的运行时的旧书。因为ASP.NET MVC建立在ASP.NET之上，理解基础知识是很重要的。如果你是一位在职的ASP.NET开发人员，那么你将发现本书不会埋没你的才能。这是一本以告诉你原理和方法为目标的快速进修书籍。

因为ASP.NET MVC是一种新技术，你可以预期要好几本书才能覆盖这一主题。这不是一个静止的框架。从2009年3月份发布以来，已经出版了几本书，但是社区正在寻求使用这一框架的新的更好的方法。最新的思路发展到MvcContrib项目，随着新材料的不断补充，频繁地发行公开版本。因为这一动态，本书处处都使用MvcContrib来介绍ASP.NET MVC。作者们都积极地使用这一框架进行开发，而MvcContrib在每个应用程序中起着关键的作用。本书的目标是在你的书架上占据一个长久的位置。API将会演变，但是使用一个MVC框架的原则以及构建URL、测试和应用程序层次的原则会更加持久。因此，我们希望本书不仅作为进入ASP.NET MVC开发的入门读物，而且作为在.NET平台上开发长寿的Web应用程序的一个指南。

我们希望本书的来到正是时候，因为在本书的编写过程中一直留意着ASP.NET MVC 2.0线路图的前景。由于线路图计划已经发布，而且V2的第一个CTP已经可用，本书中的技术在目前有用

而且也与ASP.NET MVC V2相关，这是一个捷径。我们希望本书能够帮助你在新的ASP.NET版本上开始创建许多可维护的长寿命应用程序。

## 本书的读者

本书是为高级的ASP.NET开发人员所编写的。作者都是公司、本地社区和业界的资深专家和领导者。3位作者都被Microsoft公司授予Microsoft最有价值专家（MVP）奖。因此，我们认为编写一本针对软件团队的高级成员的书籍是合适的。市场上充斥着入门书籍以及对连接文档和教程进行重新格式化的书籍，我们试图编写一本可能不适合于一些初学者，但同时能够挑战高级开发人员和架构师的书籍。不管你是否熟悉其他MVC框架，本书都将比你阅读一本技术书籍更进一步提高你的知识水平。

本书具有一个在<http://CodeCampServer.com>上实际投产的参考实现。CodeCampServer由作者和许多其他开源捐献者开发。CodeCampServer是一个ASP.NET MVC应用程序，针对用户组网站以及.NET用户组会议网站（常被称为代码集中营）。这个代码库使用Onion架构、领域驱动设计、测试驱动开发和控制反转（inversion of control）开发。本书中揭示的技术在这个项目中得以实现。本书中许多代码实例是对CodeCampServer部件的详细研究。虽然本书出版之后这个项目将会继续发展，但是它和本书的文本中体现的原则是永存的，并且可以移植到该技术的其他版本上。

因为在任何像CodeCampServer这样的实际项目中，你使用许多特定的程序库，我们也不必羞于使用这些程序。我们感到为了简单而避开其他程序库也会使读者难以应用阅读时所得到的知识。因此，我们使用了流行的程序库，比如MvcContrib、NAnt、NUnit、StructureMap、Windsor、Castle、RhinoMocks、Log4Net、NHibernate、Tarantino、AutoMapper、Iesi.Collections等。因为实际的项目是程序库的大杂烩，所以我们感到在这种实际的设置中学习ASP.NET MVC最为合适。在必要的时候我们已经小心地将所感兴趣的方面分开。我们始终将数据访问从领域模型和表现层中分离出来，并且从视图中分离表现模型；你将不会看到像从一个UI控制器直接执行查询这样的简单实例。除了在最无价值的应用程序中如<http://PartyWithPalermo.com>（一个三页的网站）之外，这都是一个坏习惯。真正的应用程序有许多屏幕，UI中的嵌入数据访问和其他逻辑使代码库的维护代价变得非常高。

我们已经尽可能地强调我们所期望的与实例相关的ASP.NET知识，但是如果你发现自己不知道什么是HTTP模块，可能就需要阅读早期的ASP.NET书籍。

## 路线图

第1章将默认项目模板拆散，让读者直接投入代码之中。在路由入门之后，主题通过简单的控制器和视图转向初始的可维护性。该章接着介绍了测试控制器的基础知识。

第2章转入模型。它不仅介绍应用程序的领域模型，还介绍根据不同的用法对不同类型模型（如表现模型）的需求。因为作者考虑使用一个表现模型（一般被称为视图模型），这对于有价值的系统的可维护性是必不可少的，所以它立刻投入使用。

第3章详细介绍控制器。控制器既可以很简单也可以很复杂，文中对这两种情况都做了介绍。

本章研究表单值和查询字符串值的使用，并且介绍模型绑定，这是目前为止ASP.NET最需要的抽象之一。第3章在概述所有可用的内建扩展点之后结束。

第4章进一步深入到视图中。在概述了默认视图引擎中的关键抽象之后，读者被引导到重要的概念（如布局、部分视图）之上，并且建立自己的验证和HTML助手。

第5章前所未有地深入到路由中。虽然大部分项目将不需要这么多高级路由，但是我们全面地研究这个主题。我们不仅介绍制作URL的方法还介绍其原因。从设计URL schema到添加动态路由，本章是你将会需要的最灵活的路由综合指南。

第6章研究许多定制和扩展ASP.NET MVC框架的方法。从自定义路由句柄开始，再转向创建控制器工厂的时机、原因和方法。在控制器工厂小节中介绍两个控制反转容器：**Windsor**和**StructureMap**。因为大部分有价值的应用程序希望使用自定义控制器工厂，这一部分被全面地介绍。接下来本章转到用动作调用和过滤器扩展控制器的各种方法。在一个自定义视图引擎和使用新的T4模板之后，读者将对可用的扩展点有一个完整的概念。

第7章交流更复杂的站点架构的伸缩方法。首先是解决大控制器动作的问题以及将多个错误的任务从控制器中移除的方法。视图助手也被更详细地介绍，还有将大的视图分段为一些内聚的更小一些的视图的技术。本章还介绍了动作过滤器所遇到的更大的问题。

第8章提供在一个ASP.NET MVC中利用现有的ASP.NET特性的方法。文中介绍了使用现有的服务器控件的方法，然后转向缓冲，包括输出缓冲和HttpContext.Items提供的请求级缓冲的使用。接下来是跟踪、健康监控、网站地图、个人化、本地化、linq、Cookie、会话状态以及配置。因为ASP.NET MVC是ASP.NET的一个附件而不是替换，本章确保读者理解现有的功能适用的场所。

第9章是早期访问程序中最流行的章节，因为在2009年中期，AJAX是个热点。我们首先放弃对AJAX的偏见，然后从简单的HTML替换开始概述其最常见的用法和技术。本章介绍了用控制器实现一个REST API，以及一些用于AJAX的第三程序库和控件。本章还概述了使控制器动作自动支持AJAX的方法。

第10章介绍宿主环境和部署。虽然不像AJAX的主题那么有吸引力，但是这是理解如何将这个框架上建立的应用程序部署到IIS 5/6/7/7.5的关键。所有版本都详细地介绍，还包括扩展、通配符映射、URL重写的隐含适用。在介绍XCOPY部署之后，本章提供了管理生产和开发环境设置的技术。本章以与CodeCampServer相似的一个自动部署实例结束。

第11章研究MonoRail和Ruby on Rails作为ASP.NET MVC的比较和基准。从MonoRail开始并介绍验证、ActiveRecord数据访问和视图引擎选择。Rails以“Rails方式”——ActiveRecord和ActionPack紧随其后。本章的目的是让读者熟悉相互竞争的MVC框架，因为好的思路来自各个方面。

第12章使用有争议的标题“最佳实践”。我们概述了这些方法支持的环境，概述了控制器、视图、路由和测试的最佳实践。每个主题都有源于这种应用类型中实际使用ASP.NET MVC的非常自信的建议。

第13章提供4个可以很容易地在你的项目中实施的综合性诀窍。这是较大的一个章节，从一个自动完成的文本框使用jQuery开始，然后转向实现自动客户端数据验证的方法。使用NHibernate的数据访问是下一个诀窍，并为从控制器调用NHibernate支持的数据访问提供一个完

整的垂直切片实现。本章概述了映射和配置NHibernate的方法以及一些基本的映射和查询技术。本章以为不喜欢Web Form的标记格式的人所实现的一个完整的Spark视图引擎结束。

## 代码惯例和下载

清单或者文本中的所有源代码都以固定宽度字体表示，以区别于普通文本。在许多清单中有代码注释以提示重要的概念。在一些情况下，清单后的解释带有着重号。

本书实例的源代码可以在出版商的网站<http://www.manning.com/ASP.NETMVCinAction>上得到。

## 作者在线

购买本书可以免费访问Manning出版公司的Web论坛，在那里你可以为本书作出评论、询问技术问题并且从作者和其他用户那里得到帮助。为了访问这个论坛并且进行订阅，可以在你的浏览器中访问<http://www.manning.com/ASP.NETMVCinAction>。

这个页面提供关于注册后如何进入论坛，论坛所能提供的帮助以及论坛的行为准则方面的信息。Manning承诺给读者的是提供单独的读者之间和读者与作者之间进行有意义交谈的场所。这并不承诺作者参加这种聚会的具体次数，对本书论坛的贡献也是自愿的（不付费）。我们建议你尝试提出一些有挑战性的问题，避免他们失去兴趣！

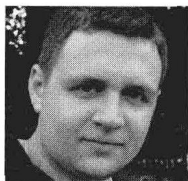
只要本书在销售中，作者在线论坛以及过去讨论的存档将可以从出版商的网站上访问。

# 关于作者



Jeffrey Palermo是Headspring Systems的CTO。Jeffrey擅长于敏捷管理并且帮助许多公司提升他们软件团队的生产力。他作为AgileAustin成员和Austin .NET用户组主管效力于AgileAustin软件社区。Jeffrey在五年前已被Microsoft授予解决方案架构方面的“Microsoft最有价值专家”(MVP)，并且参与ASPInsiders小组，这是为ASP.NET团队未来版本提供建议的场所。他还被认证为MCSD.NET和ScrumMaster。Jeffrey曾经在业界会议(如VSLive、DevTeach、the Microsoft MVP Summit)、各种ALT.NET会议和Microsoft Tech Ed上发表演讲。他还作为INETA演讲协会的成员为国内的用户组发表演讲。他的网站是headspringsystems.com和jeffreypalermo.com。他是Texas A&M大学的毕业生、雄鹰童子军奖章获得者。Jeffrey是CodeCampServer开源项目的创始人和MvcContrib项目的合作创始人。

他还负责召集“与Palermo聚会”(Party with Palermo)活动，这一活动一般安排在以微软技术为主题的行业大会的前夕。Party with Palermo开始于2005年6月，已经越来越流行，参与人数也越来越多。每次活动一般都有数百人参与，现场提供免费的食物饮料和抽奖。你可以在<http://partywithpalermo.com>看到过去和即将开始的聚会，该网站从2007年10月之后运行在ASP.NET MVC上。



Ben Scheirman是一位擅长于.NET的软件开发人员。他曾经广泛地在各种平台和语言上进行Web开发工作。Ben是一位Microsoft MVP、Microsoft ASP Insider和认证的ScrumMaster。在不进行编程工作的时候，Ben享受着演讲和博客，与妻子和5个出色的孩子在一起，或者摆弄他的最新小发明。Ben是Texas州Houston的Sogeti首席顾问。他的博客在<http://flux88.com>上。

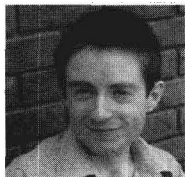


Jimmy Bogard是Headspring Systems的首席顾问。他是具备6年专业开发经验的敏捷软件开发人员。他为许多客户提供从概念到生产系统的解决方案。Jimmy所提供的解决方案从小型产品到财富百强公司的企业级电子商务应用。他还是一位Microsoft认证应用开发人员(MCAD)，.NET社区的一位积极成员，领导开源项目，提供技术演讲并且推进技术图书俱乐部。目前，Jimmy是NBehave项目(一个用于.NET的行为驱动开发框架)、AutoMapper(基于约定的对象—对象映射程序)的首席开发人员和Austin领域驱动设计图书及俱乐部的推进者。Jimmy



是ASPInsiders小组成员，并且在2009年获得ASP.NET“Microsoft最有价值专家”(MVP)奖。

## 关于技术审校者



Jeremy Skinner生活在英国，是一名软件开发人员。他的大部分工作涉及使用ASP.NET和C#编写Web应用程序。他参与多个开源项目，包括MvcContrib、Fluent Validation和Fluent Linq to Sql。

Jeremy审阅了本书的每个段落、每个图片和代码实例，他对本书的贡献是无价的。他找出并且改正了许多错误，没有他，本书就不能成为一本好书。他自己有能力成为一名作家，我们期望他在将来能贡献一本书。Jeremy对于ASP.NET MVC框架和流行的第三方框架（如Castle）的经验使他成为一名强有力的评审。他的博客在<http://www.jeremyskinner.co.uk/>上。

Freedom Dumlao是一位主要工作于.NET的软件工程师。他的博客在<http://weblogs.asp.net/FreedomDumlao/>上。Freedom审阅了本书各章的草稿，并且提出了关键的改进建议。他的观点对本书的质量很有价值。

## 关于封面插图

本书的封面插图名为“L’Habitant de Versailles”，意为凡尔赛城的居民。今天，凡尔赛是巴黎的郊区，人口超过9万人，但是在过去，它非常著名，这是因为它在17和18世纪的很多年里都是法国的都城，该城围绕凡尔赛宫发展起来。

这张插图取自Sylvain Maréchal 19世纪在法国出版的四卷地区服装习俗纲要。每张插图都是手工绘制和上色的。Maréchal的丰富收藏提醒我们仅仅200年前世界上的城市和地区在文化上的鲜明区别。由于互相之间的隔绝，人们说着不同的方言和语言。在街上和乡下，很容易从穿着上认出他们所居住的地方以及所从事的职业。

服装的代号在那之后变化了，当时丰富的地区差异已经淡化。现在很难区分出不同大陆的居民，更不用说不同的城市和地区。可能我们在更多样化的个人生活中有着文化上的差异——对于更多样和快速变化的技术生活来说确实如此。

在难于区别不同的计算机书籍的这个时代，Manning用Maréchal的图画所描述的两个世纪之前丰富的地区生活差异作为封面，来赞美计算机行业的独创性和进取心。