

ASP.NET MVC 4

实战

ASP.NET
MVC 4
IN ACTION

[美] Jeffrey Palermo
[美] Jimmy Bogard
[美] Eric Hexter 著
[美] Matthew Hinze
[英] Jeremy Skinner
徐燕萍 李萍 林逸 译



ASP.NET MVC 4 实战

ASP.NET
MVC 4
IN ACTION

[美] Jeffrey Palermo
[美] Jimmy Bogard
[美] Eric Hexter 著
[美] Matthew Hinze
[英] Jeremy Skinner
徐燕萍 李萍 林逸 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

ASP.NET MVC 4实战 / (美) 巴勒莫 (Palermo, J.)
等著 ; 徐燕萍, 李萍, 林逸译. -- 北京 : 人民邮电出版社, 2014. 5

ISBN 978-7-115-34368-0

I. ①A… II. ①巴… ②徐… ③李… ④林… III. ①
网页制作工具—程序设计 IV. ①TP393.092

中国版本图书馆CIP数据核字(2014)第019108号

版 权 声 明

Original English language edition, entitled *ASP.NET MVC 4 in Action* by Jeffrey Palermo, Jimmy Bogard, Eric Hexter, Matthew Hinze, and Jeremy Skinner, published by Manning Publications Co., 209 Bruce Park Avenue, Greenwich, CT 06830. Copyright ©2012 by Manning Publications Co.

Simplified Chinese-language edition copyright ©2014 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Manning Publications Co.授权人民邮电出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

◆ 著 [美] Jeffrey Palermo
[美] Jimmy Bogard
[美] Eric Hexter
[美] Matthew Hinze
[英] Jeremy Skinner

译 徐燕萍 李萍 林逸

责任编辑 张涛

责任印制 彭志环 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京昌平百善印刷厂印刷

◆ 开本: 800 × 1000 1/16

印张: 23

字数: 565千字

印数: 1-3000册

2014年5月第1版

2014年5月北京第1次印刷

著作权合同登记号 图字: 01-2012-7403号

定价 69.00元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315



内 容 提 要

本书讲解深入到 ASP.NET MVC 开发内部，可作为在 .NET 平台上开发 Web 应用程序的程序员的一本很好的指南。

全书分为 3 部分，共 24 章。

第 1 部分（第 1~4 章）介绍 ASP.NET MVC 框架，通过创建一个简单的留言簿应用程序，解释该框架的核心概念。第 2 部分（第 5~15 章）涉及几个在使用 ASP.NET MVC 时需要理解的重要的主题，包括验证、Ajax、安全性和路由，并用一些具体示例以阐述这些概念。此外，这部分还讲解了一些编程技巧，例如，如何构建视图专用模型和轻量控制器。最后讲解了如何在 MVC 应用程序中利用第三方的 NHibernate 项目进行数据访问。第 3 部分（第 16~24 章）探讨了几个高级主题，主要侧重于框架的可扩展点，包括如何扩展控制器、如何使用依赖性注入，以及如何扩展区域以使得它们易于可重新分发。第 23 章和第 24 章涉及移动布局和 Web API，这些是以前的 ASP.NET MVC 版本不能向后兼容的特性。

本书适合希望采用 ASP.NET MVC 技术从事 Web 应用程序开发的人员阅读，也可作为大专院校教授 ASP.NET MVC 开发技术的教材，也可供从事软件设计与开发的相关技术人员参考。

序

自我撰写本书的第一版前言之后，ASP.NET MVC 已经发生了很多事情。2011 年，微软两次发布了 ASP.NET MVC。第一次包含了对其框架的许多重要改善，但最值得关注的改善是新的 Razor 视图引擎。Razor 通过新型的整洁语法，消除了编写视图过程中所有让人感到棘手的语法冗余。更重要的是，它十分灵活且可用于 ASP.NET 之外。

第二次，他们交付了 ASP.NET MVC (ASP.NET MVC 第二版——译者注)，这一版本是 ASP.NET MVC 的工具更新，ASP.NET MVC 运行时没有任何变化。事实上，它是完全相同的运行时安装程序。但使用工具变了！这一版本的 ASP.NET MVC 包含了基于实体框架的 Code First 模型的支架。这提供了一组实体上进行简单 CRUD 操作接口所需要的全部代码。此版本也包括了 NuGet 1.0 RTM (RTM, 生产商版本——译者注)。NuGet 是一个包管理器，它能够很方便地找到并安装可以安装到 Visual Studio 项目中的包库 (NuGet 形式的包)。事实上，ASP.NET MVC 中的许多第三方库都是以这种形式交付的。这种办法的好处是，即使在项目创建之后，也可以在新版本交付时，很容易地为第三方依赖性找到并安装更新。

微软最近部署了用 ASP.NET MVC 编写的 NuGet Gallery (NuGet 走廊，展示 NuGet 包产品的一个应用程序——译者注)。这可能是我花费最近 4 年时间设计而成的框架来建立的第一个真正的 Web 应用程序。这么说肯定是有偏向的，但我相信这是事实：使用 ASP.NET MVC 进行工作是一种快乐。我当然发现了一些难点，但不管怎么说，这是一种良好的经历。在本书指导你深入 ASP.NET MVC 的内部工作，并用它建立一些 Web 应用程序时，我希望你也会有同样的感受。

Phil Haack
GitHub 公司

第二版序

如果你比较幸运，你可能会看到一个正在酝酿中的历史。对我而言，这种时刻是 2007 年 10 月。我坐在充满材料的会议室地板上，热切地看着微软 Scott Guthrie（微软软件开发部的公司副总裁——译者注）揭开了后来成为微软 ASP MVC 框架预览版的面纱。这表明，这一天是永久改变开发人员——包括许多其他方面的开发人员——生活的日子。

直接受这次会议影响的人群中包括本书——《ASP.NET MVC 4 实战》的作者。你手上所拿的这本书，是数百小时实际体验和实验的成果，是微软 ASP.NET MVC 框架最新版的最好材料。

在这本《ASP.NET MVC 4 实战》中，你将从 ASP.NET MVC 框架的专家那里学到所有主题：路由、控制器、控制器工厂、视图引擎、输入构建器、验证，以及区域等方面的知识。最后，你会发现全书遍布是宝藏：可以立即使用的提示与技巧。我可以确信的事情是，随着日复一日地使用 ASP.NET MVC 框架，本书将变成可发现无限价值的、泛黄、卷曲并做满标记的资源。

Rod Paddock

— Dash Point Software 公司总裁，

Code Magazine 主编

第一版序

ASP.NET MVC 1.0 最终版是在 2009 年 3 月的 Mix 09 会议期间发布的（Mix 会议是微软每年举行的 Web 开发者和设计者年会——译者注）。在最终版亮相之前，产品团队已经用完整的源代码发布了多个公开预览版，以努力提升微软产品的开放与社区参与性。

这么做是为什么？

透明度与社区参与是崇高的目标，但不一定是一个项目的最终目标。我们真正的目标是伟大的产品。我愿意把 ASP.NET MVC 看成是，证明透明度和社区参与是取得这一目标的重要手段的一种尝试。

在 ASP.NET MVC 预览版 2 发布之后，我们从开发人员那里收到了大量反馈，反映难以用 ASP.NET MVC 编写单元测试。Jeffery Palermo，作为《ASP.NET MVC 实战》的主笔，是在此期间提供反馈最多的人员之一。我们采纳了这些反馈，并通过引入动作结果的概念实现了一个主要 API 的修订，这是一个比原先要好得多的设计。社区参与帮助我们建立了一个更好的产品。

ASP.NET MVC 侧重于诸如关注分离这样的坚实原理，以提供一个高度可扩展且可测试的框架。虽然在你认为合适时，可以修改源代码，但框架的意愿是对扩展开放而不需要修改源代码。框架的任何一个部分都可以用你选择的其他东西来替换。不喜欢视图引擎吗？试试 Spark 视图引擎。不喜欢实例化控制器的方式吗？挂接你自己的依赖性注入容器。

ASP.NET MVC 也包括很好的工具，如 Add View（添加视图）对话框，它使用代码生成来快速创建基于模型对象的视图。最好的部分是 ASP.NET MVC 中的所有代码生成特性都依靠 T4 模板，因而是完全可定制的。

通过本书，Jeffrey 会分享所有这些特性，并且演示如何将它们组装在一起去构建很棒的应用程序。我希望你喜欢这本书，并分享他构建 Web 应用程序的激情。请记住，本书不仅邀请你了解 ASP.NET MVC，而且邀请你加入社团，并影响 ASP.NET MVC 的未来。

编程快乐！

Phil Haack

——前微软 ASP.NET MVC 团队高级项目经理

前 言

我的编程生涯始于 20 世纪 90 年代中期，当时是当地学区的一位 Web 开发人员。那时，Web 就是 HTTP。Netscape Navigator（网景浏览器）让使用 Internet 用户数量大量增长，因为在当时，它比其他任何浏览器都更先进。Netscape Navigator 3.0 版（1996 年）和 3.04 版（1997 年）帮助全世界的家庭和行业展开了 Internet 应用。而且，再没有比购物更常见的用途了！随着电子商务的来临，Internet 随着资本家的淘金热爆炸式地发展起来了。

我便是在这一公共部门开始进行 Web 开发的。在那里，我们利用了社区网的第一个线路，这是让校区毕业生与以前的其他同学通信用的。我使用 IDC（Internet Database Connector，Internet 数据库连接程序）和 HTX（HTML Extension Template，HTML 扩展模板）开始了在微软平台上的职业生涯。Internet Information Services（IIS，Internet 信息服务）2.0 版给我们提供了倚仗 ODBC 数据源的神奇的灵活性。这是我第一次使用“代码块”或 `<% %>` 界定符。IDC/HTX 后来让位于 Active Server Pages（ASP，活动服务器页面），我依然能够回想起 ASP 2.0 到 ASP 3.0 的重大改版，以及引入让人敬畏的 COM+ 集成时的情形。我曾一路涉足过 CGI、Perl、Java 和 C++，但我还是坚持选择了微软平台。

ASP 3.0 经历了随 Windows 95 一起发布的 IE 4 与 Netscape 为占领浏览器市场份额而进行的战争。编写能够很好地运行于这两种浏览器的 Web 应用程序是残酷的。IE 5.0 以它专有的 Web 扩展性，例如，能够用 `datafld=" ”` 属性对表格进行动态绑定的 XML 数据孤岛，开拓了 Intranet 应用程序的视野。（浏览器的）客户端脚本是专用的，而许多公司授权给 IE，只是为了获取高级功能。IE 5 和 IE 5.5 是 .com 繁荣时期代表微软一方的主要浏览器。随 Windows XP 发布了 IE 6，它毫不费力地占领了 Web 浏览器的主要市场。

在此期间，构建业务系统的大多数 Web 开发人员不得不对使用哪一种浏览器做出选择。对于公众网站（Internet），有许多要写两次代码的页面。对于企业内部网（Intranet），应用程序可能只会在 IE 中工作。ASP 3.0 让程序员对 HTTP、HTML、GET 和 POST 谓词等有了密切接触。CSS 仍处于幼年时期，而 `<blink>` 标签正在淡出流行。我记得曾做过一个粗糙的框架，用以处理同一个 ASP 脚本的多个请求路径。

我从一开始就采用 ASP.NET 1.0，并将部分 ASP 3.0 网站转换成了 Web Form。还记得用 GridLayout（网格布局）作为默认形式，CSS 绝对定位无处不在的那些日子吗（曾经有一段时间，几乎所有的网页都使用表格形式的布局，这里的 GridLayout 便是指当时的情形——译者注）？很显然，Web Form 1.0 是 VB 6 开发人员转到 .NET，并进入 Web 领域的桥梁。ASP.NET 1.1 摒弃了 GridLayout，并强迫开发人员理解 HTML，以及流布局的工作原理。当 IE 是“首选”浏览器时，下层渲染（Downlevel rendering，完整的名称应当是 Downlevel-hidden conditional comments（下层隐藏条件注释），这是指较早的 IE 时期，在 HTML 标记中使用注释标记控制浏览器显示的情形，如 `<!-- [if IE 6] --> ... <!-- [endif] -->` 这样的注释标记，这一标记对中的内容便属于 downlevel（下层），它对 IE 是生效的（读者不必对此加以理会，因为这些东西现在已经不再使用了——译者注）。而对其他浏览器都是下层隐藏的。当 Firefox 的市场份额开始攀升，并满足标准兼容标记时，这种格局开始被打破。我们终于知道，需要声明 DOCTYPE，

而且不再需要下层渲染，于是我们关闭了这一特性。

我成了一名 ASP.NET 专家，并且是 .NET 2.0 beta 版测试期间的一位活跃的博主。我了解 ASP.NET 1.1 到 2.0 的每一个特性和每一个重大变更，而且帮助我的团队采用了 2.0。在 ASP.NET 2.0 时代，我开始关注 Martin Fowler 关于 Model-View-Presenter（模型-视图-呈现器）的文章。我实现了这种模式，将逻辑从已经很臃肿的后台代码文件中剥离出来。2005 年，Java 开发人员很享受有多种用于 Web 的 MVC 框架可供选择。而我却正在奋力从 Web Form 转向模型-视图-呈现器，并提议测试驱动开发。

2006 年，随着工作变动，我直接跳转到 WinForm 的智能客户端开发。依然是沉闷的后台代码模型，而且需要管理一个开发团队，于是我实现了以 WinForm 类作为视图的“模型-视图-控制器”模式。这终于让人得以放松。UI 开发是无缝的，而控制器是域模型到 UI 的自然边界。2007 年，我又开始了 Web 开发，并再次用 Web Form 勉强实现了“模型-视图-呈现器”。回想过去，我真希望我从一开始就采用了 MonoRail——这是另一个 .NET 的“模型-视图-控制器”框架。

2007 年 3 月，Scott Guthrie (@scottgu) 创建了后来成为 ASP.NET MVC 框架的原型。Guthrie 收到了很多客户来信，都是关于使用 Web Form 的困难，他们需要一种更简单、更灵活的方式来编写 Web 应用程序。在 2007 年的 MVP（微软认证的最有价值专家）峰会上，Guthrie 找到了一个微软 MVP 小组。这个小组由我、Darrell Norton、Scott Bellware 和 Jeremy Miller 等人组成，验证了他的原型构想，并给出了最终被编入框架的一些初步实现。

2007 年 10 月，当 Scott Guthrie 在 AltNetConf Open Spaces 会议上演示 ASP.NET MVC 的工作原型与构想时，我立刻意识到，这就是我一直希望拥有的。作为一名长期的 Web 开发人员，我理解 HTTP 和 HTML，而且我认为，这是 ASP.NET 1.0 就应该做到的。从 ASP 3.0 到 ASP.NET MVC 的过渡非常顺利。当说服 Guthrie 给了我一个他的原型副本时，我可以肯定地说，是我做出了第一个产品形式的 ASP.NET MVC 应用程序。我修订了我的注册网站 <http://www.partywithpalermo.com>，并于 2007 年 11 月在 DashPoint 上的一个 Rod Paddock 服务器上发布了它。

当 Manning 出版社约我写一本 ASP.NET MVC 的书时，我已经是这一主题的活跃博客人，并已经在 CoDe 杂志上发表了一篇关于该框架的文章。Ben Sheirman、Jimmy Bogard 和我在这本《ASP.NET MVC 实战》上花了一年多的时间，我们非常高兴地看到了它的出版，并深受开发者社区的好评。微软继续发布了下一版本 ASP.NET MVC 的升级预览版。对于本书的第二版，我们的作者队伍加入了两位新成员：Eric Hexter 和 Matthew Hinze。我们 5 人利用在这一领域耕耘所获得的框架知识，以及编写第一版时所得到的经验，于 2009 年年底开始编写《ASP.NET MVC 4 实战》。

对于《ASP.NET MVC 4 实战》，Ben 转入 iOS 开发，而 Jeremy Skinner 加入了作者团队，并带来了他作为 MvcContrib 委员会委员和 ASP 权威人士的渊博知识。这一版本发布时，正是 HTML 5、CSS 3 和 jQuery 流行之时。ASP.NET MVC 进一步集成了 jQuery，并提供了对 JavaScript 的更多支持——JavaScript 正越来越多地被 Node.js 和 Backbone.js 之类的其他框架所采用。

与开发者部（Developer Division，微软的一个开发部门，专门从事于开发人员工具等方面的开发——译者注）以前的项目相比，微软对 ASP.NET MVC 的发布周期是前所未有的。该项目至少每季度都会在 CodePlex 上发布其源代码和其他全部内容。该框架也是使用测试驱动开发这种软件构建技术开发而成的。源代码下载中包含了完整的单元测试内容，ASP.NET MVC 也是在 Apache 开源许可协议下发布的。如果你选择这么做（同意遵守 Apache 开源许可协议——译者注），现在可以直接向 ASP.NET

开发团队提交下载请求。

ASP.NET MVC 以 Web 同样的工作方式进行工作，这是一种自然的吻合。尽管微软是最后一个为其开发平台提供“模型-视图-控制器”框架的，但该框架是一个强有力的竞争者。它的设计首先侧重于核心抽象理念。它也有利于社区扩展。事实上，在第一个社区技术预览版（Community Technology Preview, CTP）发布的同一周，Eric Hexter 和我就推出了 MvcContrib 开源项目—最先提供了与 ASP.NET MVC 框架的集成扩展。MvcContrib 随后被 CodePlex 基金会接纳为第一个社区项目——CodePlex 是一个推动开源合作贡献的组织。

在我负责咨询活动的 Headspring 公司中，ASP.NET MVC 是一个频繁使用的工具。2009 年，对于整个 .NET 业界，我曾预言，至 2011 年，ASP.NET MVC 将被视为 ASP.NET 开发的标准。现在，2011 年已过去，该预言也成真了。每天都有新的开发人员加入 .NET 平台，而且，对于 Web 开发人员而言，ASP.NET MVC 非常容易上手。由于降低了复杂性，采用该平台的障碍减少了，而由于其简单性，它能够扩展到满足某些最复杂企业系统的需求。

与此同时，该框架也给我们的客户端项目带来了直接的好处。将该框架实际运用于客户端项目，会明显有助于提高本书所含信息的质量——因为本书是基于实际工作经验编写的。我们看到了成功，也发现了一些不能正常工作的东西。从读者的需求出发，在本书中提供了这些经验教训，并希望在你编写了第一个应用程序之后，这本书仍能与你相伴。

虽然其他平台从模型-视图-控制器框架中得益已经有很多年了，但对于许多 .NET 开发人员来说，MVC 模式还是陌生的。本书将解释使用该框架的方法和时机，以及该模式和那些颇受赞誉模式背后的理论和原理。我们希望本书将有助于向你传授一项十分易学的必备技术。

Jeffrey Palermo

致 谢

我们要感谢 Scott Guthrie，他在 .NET 开发中看到了对这一框架的需求。没有他的原型、想象力和领导，这一成果仍然不会出现在 .NET 框架中。还要感谢 ASP.NET MVC 项目经理 Phil Haack，及他所领导的 Microsoft 的核心 ASP.NET MVC 团队。ASP.NET MVC 1 团队的其他关键成员有 Eilon Lipton（首席开发）、Levi Broderick（开发人员）、Jacques Eloff（开发人员）、Carl Dacosta（QA）和 Federico Silva Armas（首席 QA）等。还要感谢微软支持团队的其他很多工作人员，他们负责 ASP.NET MVC 框架的包装、文档和分发。与其他产品相比，该框架虽然很小，但微软此举却转变了整个 .NET 软件业的思维模式。

任何大型出版物都需要大批人员的巨大努力，本书也不例外。5 位作者完成本书写作，每个人都是多个在建项目的顾问。这本第 3 版书籍，从 ASP.NET MVC 的第一个预览版开始，花费两年半的时间。这种工作环境需要 Manning 出版社工作人员的大力支持。我们感谢他们自始至终的耐心和支持。我们还要特别感谢采编 Michael Stephens，他看到了此项技术将成为先进技术书籍的潜力，并批准发行了整个项目的创意共享原始文件。Michael 最初在 2007 年看到了本书的需求，并就编写第一版事宜与我进行了接洽。

我们真诚地感谢 Phil Haack 和 Rod Paddock，他们审阅了手稿，并写了精彩的前言。独立技术评审 Javier Lozano 也非常出色，没有他提供的信息，这本书不会像我们期望得这么好。

本书还得益于那些在百忙中抽出时间阅读部分手稿，并提供反馈的技术评审者：Alonso Robles、Anne Epstein、Brandon Barry、Cedric Yao、Chris Missal、David Brown、Deran Schilling、Dustin Wells、Eric Sollenberger、Glenn Burnside、JT McCormick、Justin Pope、Katie Barbaro、Kelly Schaub、Kevin Hurwitz、Kurt Schindler、MahendraMavani、Mary Chauvin、Nolan Egly、Patrick Lioi、Pedro Reys、Rebecca Heath、SharonCichelli、Steve Donie、Tim Thomas、Roger Wright、Andrew Siemer、Dhiren Sham、JonasBandi、Tetsuo Torigai 和 Gaston Verelst 等。

Jeffrey Palermo

首先，我要感谢我美丽的妻子 Liana，在本书写作过程中的耐心和支持。在本书第一版写作之初，Liana 生下了我们的第一个孩子 Gwyneth Rose，接着是第二个孩子 Xander。还要感谢我的父母 Peter 和 RosemaryPalermo，从小就给我灌输对书籍和学习的热爱。我必须提到我的德克萨斯 A&M 大学教授 Mike Hnatt，通过他的编程课程、业务训练和持续的友谊，对我的工作影响深远。最后，要感谢 Dustin Wells 和 Kevin Hurwitz。我们在一起建立了 Headspring 咨询公司，使我们能够进行深入的研究和实践，并孕育了使用 ASP.NET MVC 的先进方法。

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。我还要感谢那些通过书籍、文章、博客、代码、演示和事件给社区反馈的人们。我也想感谢所有来到我面前的大师，好

心分享使他人可以学习和成长的智慧。最后，感谢我的父母和家人多年来对我努力的支持和指导。

Eric Hexter

首先，我想对我美丽聪明的妻子 Chriss 说声谢谢。没有她，我就没有完成这一项工作的动力和灵感。她是一个很好的母亲和妻子。我还要感谢我可爱的女儿 Emerson、Elliott 和 Everlee，她们使我的生活如此与众不同。我的家人自始至终都在帮助我，感谢他们当我在上大学时，为我买了第一台计算机。我想这一切都是值得的！感谢爸爸、妈妈和 Gordon。我还要感谢大学教授 Bob Williams 博士，是他鼓励我努力进入软件行业。

Matthew Hinze

我要感谢我的父亲 Rick Hinze，他给了我无尽的支持和友谊。他让我进入这个行业。我还要感谢我的妻子 Sarah，她让我脱颖而出。

Jeremy Skinner

非常感谢我的父母 Paul 和 Nina，他们支持并鼓励我完成了这个项目。我还要感谢我的第一个老板 David Woodward，他给了我进入这个行业的机会。

关于本书

ASP.NET MVC 框架在过去几年已经发展了很长时间。

ASP.NET 框架最初是 Scott Guthrie 在 2007 年年初设想的，随着 2007 年后期的原型演示，并聘请 Phil Haack 担任高级项目经理，该设想终于变成了现实。在接下来的一年中发布了该框架的几个公共预览版，随后于 2009 年年初发布了最终的 ASP.NET MVC 1.0 版。

那时，.NET 社区的许多 Web 开发人员都感到十分沮丧，因为其他平台都有很好的 MVC 框架可用（如 Ruby on Rails），这些框架可以提供轻量级的、整洁且简单的方式构建 Web 应用程序，而 ASP.NET Web Form 正在失宠。开发人员努力让它完成最初没有想让它做的事。对许多具有 Web 背景的开发人员来说，页面生命周期的复杂性和伪状态化模型都是很陌生的概念。

ASP.NET MVC 的目的是要解决这个问题，通过将自身定位为 Web Form 的一个替代平台，实现在 .NET 平台上开发 Web 应用程序。ASP.NET MVC 的灵感来自于其他框架，如 Rails、MonoRail 等。它为 .NET 开发人员构建 Web 应用程序提供了一种更整洁的方式。

随着 2010 年 3 月 ASP.NET MVC 2 的发布，它增加了对第一版中缺失的几个重要特性的支持（如验证和模板辅助器）。现在，通过 ASP.NET MVC 4，微软已经建立了坚实的平台，并引入了几个主要的新变化——作为渲染 HTML 的默认机制、新的 Razor 视图引擎替代了 Web Form ASPX 引擎，另外，框架也包含了许多随 .NET 4 引入的新特性。

ASP.NET MVC 对于有多种软件背景的人员来说，熟悉 Visual Studio 开发经验是最好不过的。对于以 .NET 1.0 或更晚版本开始软件开发的人来说，这是一个思维上的根本性转移，因为他们是将 Web Form 视为“规范的”Web 开发而成长起来的。

本书首先介绍 ASP.NET MVC。如果之前从未用过 ASP.NET MVC，或者有过使用以前版本的经历，但想看看版本 3 中有哪些新内容，这些内容是有益的。此后，我们将深入到对现代 .NET 的 Web 开发非常重要的核心概念，包括使用 Ajax、整洁的 URL、依赖性注入，以及验证等。

本书的目的是能长久地待在你的书架上。API 会有发展，但是使用 MVC 框架，以及构建 URL、测试和应用程序层次的方式，其背后的原理是持久不变的。我们希望本书不仅深入到 ASP.NET MVC 开发内部，还能作为在 .NET 平台上长期开发 Web 应用程序的一本很好的指南。

学习路线图

本书分为 3 部分，共 24 章：

- 第 1 部分（第 1～4 章）介绍 ASP.NET MVC 框架，通过创建一个简单的留言簿应用程序，解释该框架的核心概念。
- 第 2 部分（第 5～15 章）涉及几个在使用 ASP.NET MVC 时需要理解的重要的主题，包括验证（第 6 章）、Ajax（第 7 章）、安全性（第 8 章）和路由（第 9 章）。本书这部分提供了一些具体示例以阐述这些概念。此外，这部分还涉及编程的技巧，例如，如何构建视图专用模

型（第 5 章），以及轻量控制器（第 12 章）。该部分还讲解了如何在 MVC 应用程序中利用第三方的 NHibernate 项目进行数据访问。

- 第 3 部分（第 16~24 章）讲解了高级主题，主要侧重于框架的可扩展点，包括如何扩展控制器（第 16 章），如何使用依赖性注入（第 18 章），以及如何扩展区域以使得它们易于可重新分发（第 19 章）。然后，第 23 章和第 24 章涉及移动布局和 Web API，这些是以前的 ASP.NET MVC 版本不能向后兼容的特性（意即，是以前版本没有的特性——译者注）。

本书的读者

本书主要是为使用 ASP.NET 的初、中、高级开发人员编写的。本书的第 1 部分主要照顾以前从未用过 ASP.NET MVC，或者有过旧版体验并寻求升级的开发人员。

本书的第 2、3 部分适合各种水平的开发人员，他们希望扩展 ASP.NET MVC 知识，以便将其用于实际应用程序。此外，这些章节也有益于应用程序架构师和团队领导，他们必须选择其团队所用的技术。本书假设你已经熟悉 Web 开发概念（如 HTTP、HTML、CSS 及 JavaScript 等），并且具有 C# 语言的编程经验。

源代码约定与下载

所有清单或正文中的源代码都以等宽字体表示，以区别于普通文本。许多清单中都带有代码注释，以强调重要概念。在有些情况下，数字式标号与清单下面的解释关联在一起。

本书实例的源代码可以在出版商的网站 <http://www.manning.com/ASP.NETMVC4inAction> 上得到。

作者在线

本书的读者可以免费访问 Manning 出版公司的 Web 论坛。在那里，你可以对本书进行评论，提问技术问题，并得到作者或其他用户的帮助。为了访问论坛并进行订阅，可以访问 <http://www.manning.com/ASP.NETMVC4inAction>。

该页面提供了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等方面的信息。Manning 对读者的承诺是为读者提供一个场所，在此，个别读者之间，以及读者与作者之间，可以进行有意义的对话。但对作者的参与并不承担任何义务，作者对本书论坛的贡献是自愿的（也是无报酬的）。我们建议你尝试问一些有挑战性的问题，以免他们失去兴趣！

关于作者



Jeffrey Palermo 是两个孩子 (Gwyneth Rose 和 Xander) 的爸爸, 也是一个幸运的丈夫。在空余时间, 他喜欢弹奏吉他和学习有关业务。在业务领域, 他是以 Austin (奥斯汀, 得克萨斯州府) 为基地的软件咨询公司 Headspring 的总裁兼 COO (首席运营官)。Jeffrey 领导了 Headspring 咨询业务的成长——从一个小型的开发公司成长为一个数百万美元的软件定制公司。Jeffrey 一直在寻求一种更好的构建软件方式——他能够在其中使用新的技术而又不失过去的优势。他倡导适度、简单的办法。Jeffrey 自 2006 年起便被微软认定为 MVP。他在很多业界会议, 如 VSLive、DevTeach、微软 MVP 峰会, 各种 ALT.NET 会议, 以及 Microsoft Tech Ed 上做过演讲或起过推动作用。他也曾作为 INETA 演讲者协会 (INETA Speakers' Bureau) 的成员, 为用户组做过演讲。他毕业于 Texas A&M 大学, 目前正在 Jack Welch 管理学院学习 MBA 课程。

Jeffrey 和 Eric Hexter 一起创立了 MvcContrib 开源项目。如今, 该项目可以在微软 CodePlex 基金会的主页上找到。

Jimmy Bogard 是 Headspring 的一位技术架构师。他是一位具有 6 年专业开发经验的 Agile 软件开发人员。他为许多客户提供从概念到产品的系统解决方案, 这些方案涵盖了小型产品到财富百强的企业级电子商务应用。他还是一位 MCAD (Microsoft Certified Application Developer, 微软认证的应用程序开发专家), 也是 .NET 社区的一位活跃成员, 领导着开源项目, 提供技术帮助。



目前, Jimmy 是 NBehave 项目 (一个 .NET 的行为驱动开发框架) 和 AutoMapper (一个基于约定的“对象-对象”映射器) 的首席开发人员, 他是 Austin Domain-Driven Design Book Club (Austin 领域驱动设计图书俱乐部) 的主持人。Jimmy 是 ASPInsiders 和 C# Insiders 小组的成员, 并于 2009 年获得 ASP.NET 的“MVP 奖”。



Eric Hexter 从事专业软件开发 15 年以上, 内容涉及咨询、产品开发、企业 IT、高级品牌的 Web 网站和电子商务。他是 Agile 项目管理和软件工程实践的积极倡导者。

Eric 在 Austin 开发社区非常活跃。他是 Austin 的 .NET 用户组 (Austin.NET Users Group) 的主管。自 2007 年起, 他就举办了 Austin 代码营 (Austin Code Camp), 这是一个为期一天的开发人员会议。除了 Austin 的 .NET 用户组的任职以外, Eric 还曾担任过以下职位: 南得克萨斯州 INETA 成员导师、ASP-Insider (ASP 权威人士)、微软 ASP.NET 的 MVP, 以及 MVC 虚拟用户组社区的创始人。Eric 和 Javier Lazano 共同创立了 MVC 社区, 这是一个专注于 ASP.NET MVC 的虚拟用户组。Eric 还与人合作创立了 MVCCConf, 这是一个为期一天的大型在线

ASP.NET MVC 会议 (mvcconf.com)。他在 Los Techies (一个社区主导型的技术博客社区) 上开设了博客, 并在得克萨斯和美国各地的技术会议上为用户组发表演讲。

Eric 受到很多人羡慕, 他有一位美丽的妻子(Chriss)和 3 个可爱的女儿(Emerso、Elliott 和 Everlee)。他用尽可能多的时间与家人在一起。

Matt Hinze 是一位来自得克萨斯奥斯汀的程序员和软件设计人员。自 2005 年起, 他已成功地向软件开发人员开设了多门技术课程。Matt 是一位 ASP 权威人士和微软 C# 的 MVP。他就职于 Round Rock 一家大型电子商务公司 Sogeti, 并与他的妻子和儿子生活在奥斯汀北郊。



Jeremy Skinner 是一位英国籍软件开发人员, 主要擅长 ASP.NET MVC 和 C# 的 Web 应用程序开发。他参与了多个开源项目, 包括 MvcContrib 和 FluentValidation, 还是微软 ASPInsiders 小组的成员。可以在 <http://www.jeremyskinner.co.uk> 上看到他的博客。

关于译者

徐燕萍：盐城工学院教师，硕士，印度 NIIT 双语教学认证教师，长期从事计算机专业的教学与科研工作。主要研究领域：计算机网络与分布式系统、面向对象方法与技术、Web 数据库应用开发。承担了本书第 1~12 章的文字翻译工作。

李 萍：盐城工学院教师，硕士，印度 NIIT 双语教学认证教师，长期从事计算机专业的教学与科研工作。主要研究领域：大型 Web 数据库应用与设计、计算机应用与开发。承担了本书第 13~24 章的文字翻译工作。

林 逸：盐城工学院教师，硕士，德国 Karlsruhe（卡尔斯鲁厄）大学访问学者，印度 NIIT 双语教学认证教师，长期从事网络运行与管理及计算机专业的教学与科研工作。主要研究领域：计算机网络与安全、软件体系架构与应用系统开发。承担了全部译文的修订、审阅与最终定稿工作。