

Pro ASP.NET 4 in C# 2010 Fourth Edition

ASP.NET 4

高级程序设计

(第4版)

Matthew MacDonald

[美] Adam Freeman

Mario Szpuszta

博思工作室 译

- ASP.NET经典著作之屠龙刀
- 全面、深入、贴近实战，世界顶尖专家倾力打造
- 国内MVP担纲翻译，名著名译，相得益彰

TURING 图灵程序设计丛书

Pro ASP.NET 4 in C# 2010 **Fourth Edition**

ASP.NET 4 高级程序设计 (第4版)

[美] Matthew MacDonald Adam Freeman 著
Mario Szpuszta
博思工作室 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

ASP.NET 4高级程序设计 : 第4版 / (美) 麦克唐纳 (MacDonald, M.), (美) 弗里曼 (Freeman, A.), (美) 兹普兹塔 (Szpuszta, M.) 著; 博思工作室译. -- 北京: 人民邮电出版社, 2011. 6

(图灵程序设计丛书)

书名原文: Pro ASP.NET 4 in C# 2010, Fourth Edition

ISBN 978-7-115-25329-3

I. ①A… II. ①麦… ②弗… ③兹… ④博… 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(2011)第086729号

内 容 提 要

本书是 ASP.NET 领域的鸿篇巨制, 全面讲解了 ASP.NET 4 的各种特性及其背后的工作原理, 并给出了许多针对如何构建复杂、可扩展的网站从实践中得出的建议。本书还深入讲述了其他 ASP.NET 图书遗漏的高级主题, 如自定义控件的创建、图像处理、加密等。此外, 本书专门提供了两章的内容来教你如何何用 Ajax 技术制作快速响应的页面, 以及如何使用微软的 ASP.NET AJAX 平台。另外, 还专门介绍了 ASP.NET 4 新增的功能, 如 MVC 和动态数据等。

本书适合各层次的 ASP.NET 程序员阅读。

图灵程序设计丛书

ASP.NET 4高级程序设计 (第4版)

-
- ◆ 著 [美] Matthew MacDonald Adam Freeman
Mario Szpuszta
译 博思工作室
责任编辑 王军花
执行编辑 毛倩倩
- ◆ 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 电子邮件 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: 72.75
字数: 2049 千字 2011年6月第1版
印数: 1-3 000册 2011年6月北京第1次印刷
著作权合同登记号 图字: 01-2010-6603号

ISBN 978-7-115-25329-3

定价: 148.00元

读者服务热线: (010)51095186转604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Original English language edition, entitled *Pro ASP.NET 4 in C# 2010, Fourth Edition* by Matthew MacDonald, Adam Freeman and Mario Szpuszta, published by Apress, 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705 USA.

Copyright © 2010 by Matthew MacDonald, Adam Freeman and Mario Szpuszta. Simplified Chinese-language edition copyright © 2011 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Apress 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。

引 言

.NET 最初出现时，它引入了不少新技术。当时它带来了编写 Web 应用程序的全新方法（ASP.NET）、连接数据库（ADO.NET）的全新方式、全新的类型安全语言（C#和 VB.NET）以及托管的运行时（CLR）。这些新技术中特别值得注意的是 Windows Forms，它是构建 Windows 应用程序的类库。

毫无疑问，你已经知道 ASP.NET 是微软创建服务器端 Web 应用程序的新一代技术。它构建在 Microsoft .NET Framework 之上，.NET Framework 聚合了紧密相关的多种新技术，彻底改变了从数据库访问到分布式应用程序的一切。而 ASP.NET 是 .NET Framework 中最重要的部件之一，通过它我们可以开发出高性能的 Web 应用程序。

开发人员很容易对 ASP.NET 技术感兴趣。毫不夸张地说，ASP.NET 是针对 Web 开发的最为完整的技术平台。ASP.NET 技术大大超过了它的前辈 ASP——ASP 是一个粗制滥造的快捷工具，用于将动态内容插入到普通 Web 页面。相比而言，ASP.NET 是一个非常成熟的平台，我们可以用它来开发广泛适用且高效的 Web 应用程序。

本书将讲述掌握 ASP.NET 4 所需的相关知识。如果你曾使用前一个版本的 ASP.NET 编过程序，那么可以将重点放在学习 ASP.NET 的新特性上，例如第 32 章的 ASP.NET MVC、第 33 章的 ASP.NET 动态数据（ASP.NET Dynamic Data）和第 34 章的 Silverlight。如果你从未接触过 ASP.NET 技术，会发现本书是一本很好的指南，将引导你逐步学习所有基础知识和隐藏在 ASP.NET 技术背后的工作原理。阅读本书的唯一前提是，具备良好的 C#语言基础并了解 .NET 的基本知识。如果你用过其他语言，比如 Java 或者 C++，但是没有接触过 C#，先看介绍 .NET 基础知识较多的其他图书会更好一些，比如 Andrew Troelsen 所著的《C#与 .NET 4 高级程序设计（第 5 版）》^①。

本书内容

下面是本书的内容提要。

第一部分 第 1 章简要介绍了整个 ASP.NET 平台和 .NET Framework，并大致说明了 ASP.NET 4 所做的改动。第 2 章详细介绍了开发利器 Visual Studio 2008 的使用方法。第 3 章至第 6 章讲解了 ASP.NET 基础结构的重要知识，比如 Web 页面模型、应用程序配置以及状态管理。在学习这些核心概念的时候，还可以了解 ASP.NET 处理请求和管理 Web 应用程序生命周期的一些底层知识。该部分还讲解了如何扩展 ASP.NET 架构。

第二部分 这部分解决所有软件开发的一个核心问题——访问及操作数据。第 7 章和第 8 章介绍 ADO.NET 应用于应用程序时的基本原理以及如何设计数据访问组件。第 9 章和第 10 章介绍 ASP.NET

^① C#领域享有盛誉的圣经级著作，已由人民邮电出版社出版。——编者注

的一整套数据绑定控件，你可以用来格式化并展现数据，而不再需要编写页面代码。第 11 章介绍如何通过高级缓存策略来获得更高的程序性能。最后，第 12 章至第 14 章从 ADO.NET 转到讲解如何使用文件、LINQ 和 XML。

第三部分 这部分将介绍管理大量 Web 页面的重要技术和特性。第 15 章介绍如何使用用户控件技术来重用用户界面。第 16 章会涉及主题（自动控制控件样式）和母版页（可供多个页面重用的布局模板）。第 17 章展示了如何使用 ASP.NET 的导航模型让访问者从一个页面转到另一个页面。最后，第 18 章讲述部署技术和 IIS Web 服务器软件。

第四部分 这部分让你了解 ASP.NET 丰富的安全特性。第 19 章简要介绍安全概念，第 20 章讲解了表单验证，第 21 章讲解新的成员资格 API。第 22 章介绍 Windows 验证，第 23 章介绍如何通过复杂的授权规则来约束已经验证的用户以及如何使用基于角色的安全特性。第 24 章介绍用户配置特性，这是一个用来存储用户特定信息的预先构建的解决方案。第 25 章介绍如何保护存储在数据库内的数据以及在 URL 中发送的加密信息。最后，第 26 章展示了如何通过设计一个自定义的成员资格提供程序来插入 ASP.NET 安全模型。

第五部分 这部分展示了用来扩展 Web 页面的高级技术。第 27 章介绍自定义控件。第 28 章介绍手工控制图片的 GDI+ 技术。第 29 章和第 30 章介绍如何在 ASP.NET Web 页面里使用 JavaScript 和 Ajax 技术以便让 Web 页面变得更加动态（通过加入文本自动完成和拖放的效果）且更具响应性（通过响应客户端事件并无缝地刷新 Web 页面）。第 31 章展示了用于构建 Web 门户的 ASP.NET 的 Web 部件的功能。

第六部分 在这一部分中，我们将学习现代 Web 开发中最精彩的创新。在第 32 章中，我们将要探索 ASP.NET MVC，它可以代替传统的 Web 表单模型，让开发人员能够完全控制 HTML 的呈现以及 URL 的结构。在第 33 章中，我们将学习 ASP.NET 动态数据，它是快速创建涉及查看和编辑数据库信息的应用程序的完美解决方案。最后，第 34 章将深入介绍 Silverlight，它是微软创建的浏览器插件，可以给各种浏览器和操作系统上的普通 Web 页面带来丰富的图形、动画、音频和视频。

读者对象

本书针对的是有一定服务器端 Web 开发知识的专业开发者。本书并没有完整描述 .NET Framework 的每一个细节，实际上，那样的一本书可能会有本书的两倍厚。本书为专业开发者提供精练的 ASP.NET 技术介绍，而这类读者基本上都不想去重复最基本的概念。在阅读本书的过程中，你要关注的是需要用来构建专业 Web 应用程序的 .NET Framework 技术，包括数据访问和 XML。使用这些功能，你就可以使用当前最好的工具来构建下一代网站了。

本书还极其实用。你不仅可以学习功能，还可以学到实用技术，从而让你的网站达到一个更高的层次。后面的章节专门介绍了很多高级功能，比如自定义控件、动态图片、高级安全、高性能数据访问等。所有这些是你构建专业 Web 应用程序所必需的东西。

为了最大限度地利用本书，你最好熟悉 C# 语法以及面向对象的概念。你不一定非要用过老版本的 ASP.NET，因为本书介绍了所有相关的基础知识。如果你很熟悉 Java 或者 C++ 但没用过 .NET，需要考虑同时阅读一些介绍 .NET 的图书，比如 Andrew Troelsen 的《C# 与 .NET 4 高级程序设计（第 5 版）》（人民邮电出版社，2011）。

学习本书的硬件和软件环境

你需要一个安装有 Visual Studio 2010 的电脑来开发和测试 Web 应用程序。虽然从理论上说，你

可以手写代码，但这是极其乏味的，并且极易出错，所以在专业环境下从来不用这种方法。此外，如果要承载 ASP.NET 网站，需要使用基于服务器的 Windows 版本，如 Windows Server 2003 或 Windows Server 2008。还需要安装 IIS (Internet Information Services, Internet 信息服务)，它是 Windows 操作系统的一个组件，将在第 18 章中介绍。

本书包含了许多示例，这些示例都使用 SQL Server 自带的样例数据库来演示数据存取、安全技术等功能。你可以使用任何版本的 SQL Server 来演示这些例子，比如 SQL Server Express 版本，它是与 Visual Studio 的某些版本一起包含在内的（并且可以在 <http://www.microsoft.com/express/database> 上免费下载）。如果你使用其他关系数据库引擎，也可以使用相同的概念，但需要修改示例代码。

相关资源

我们非常重视读者的反馈，也希望能了解你对本书的想法——哪些是你喜欢的地方，哪些是你不喜欢的地方，还有你希望我们下次改进的地方。你可以发送建议到 feedback@apress.com。请务必在邮件中说清本书的书名。

示例代码

如果打算下载与本书配套的示例代码^①，可以访问 Apress 的网站 <http://www.apress.com>，选择本书书名就可以下载 ZIP 格式的示例代码文件了。在使用这些代码之前，需要使用 WinZip 等工具解压缩下载文件。代码是按章顺序排列的，在使用代码之前，请阅读 readme.txt 文件来了解相关注意事项。

附送章节

在 Apress 网站上还有一些可下载的 PDF 格式的其他章节。这些章节由于篇幅所限未包括在书里，而且对于 ASP.NET Web 开发也不是特别重要，如下所示。

- ❑ Bonus Chapter 1. 这一章介绍如何在 ASP.NET 网站里使用资源和本地化。如果需要开发用多种语言查看的网站，则本章内容十分重要。
- ❑ Bonus Chapter 2. 这一章介绍如何为自定义控件添加设计时支持，添加后它们在 Visual Studio 环境里的行为看起来会很棒，它们会管理自身属性的序列化，并能够提供高级的设计器功能，如智能标签。

注解 这些附送章节完全引用自本书的前一个版本。这些章节的信息对 ASP.NET 4 依然适用，因为这些特性并没有发生任何变化。

勘误表

我们尽可能保证文字及代码没有错误。但是，人无完人，金无足赤，错误和疏漏在所难免。如果你在本书中发现了错误，比如拼写错误或者代码错误，我们将非常高兴得到你的反馈。你提交的勘误将使其他读者受益，同时也可以帮助我们提高出版质量。请将发现的错误发送到 support@apress.com，这些错误将被核对并发布到勘误表页面或者在本书的下一版本中修正。你可以从本书的网页上查看勘误表。

^① 与本书配套的示例代码可以在图灵网站 www.turingbook.com 本书主页上免费注册下载。——编者注

目 录

第一部分 核心概念

第 1 章 ASP.NET 简介	2
1.1 ASP.NET 的 7 大要点	2
1.1.1 要点 1: ASP.NET 是同 .NET Framework 集成在一起的	2
1.1.2 要点 2: ASP.NET 是编译执行的, 而不是解释执行的	3
1.1.3 要点 3: ASP.NET 支持多语言	4
1.1.4 要点 4: ASP.NET 运行在公共语言运行库内	6
1.1.5 要点 5: ASP.NET 是面向对象的	7
1.1.6 要点 6: ASP.NET 支持所有的浏览器	8
1.1.7 要点 7: ASP.NET 易于部署和配置	8
1.2 ASP.NET 的演变	9
1.2.1 ASP.NET 1.0 和 ASP.NET 1.1	9
1.2.2 ASP.NET 2.0	9
1.2.3 ASP.NET 3.5	10
1.2.4 ASP.NET 4	12
1.2.5 Silverlight	14
1.3 总结	15
第 2 章 Visual Studio	16
2.1 Visual Studio	16
2.1.1 网站和 Web 项目	17
2.1.2 创建无项目文件的网站	18
2.1.3 设计网页	21
2.2 Visual Studio IDE	26
2.2.1 解决方案资源管理器	28
2.2.2 文档窗口	29
2.2.3 工具箱	29
2.2.4 错误列表和任务列表	30
2.2.5 服务器资源管理器	31
2.3 代码编辑器	32
2.3.1 添加程序集引用	33
2.3.2 智能感知和大纲显示	35
2.3.3 Visual Studio 2010 改进	37
2.4 编码模型	42
2.4.1 代码隐藏文件如何与页面连接	44
2.4.2 控件标签如何与页面变量连接	44
2.4.3 事件如何与事件处理程序连接	46
2.5 Web 项目	47
2.5.1 基于项目的开发	47
2.5.2 创建 Web 项目	48
2.5.3 迁移旧版 Visual Studio 创建的网站	49
2.6 Visual Studio 调试	51
2.6.1 单步调试	52
2.6.2 变量监视	54
2.6.3 高级断点	55
2.7 Web Development Helper	56
2.8 总结	57
第 3 章 Web 窗体	58
3.1 页面处理	58
3.1.1 HTML 表单	59
3.1.2 动态用户界面	60
3.1.3 ASP.NET 事件模型	61
3.1.4 自动回送	62
3.1.5 视图状态	63
3.1.6 XHTML 兼容性	66
3.1.7 客户端控件 ID	71
3.2 Web 窗体处理阶段	73
3.2.1 页面框架初始化	74
3.2.2 用户代码初始化	74

3.2.3 验证	74	4.4.2 BulletedList 控件	120
3.2.4 事件处理	75	4.5 输入验证控件	121
3.2.5 自动数据绑定	75	4.5.1 验证控件	121
3.2.6 清除	76	4.5.2 验证流程	122
3.2.7 页面流示例	76	4.5.3 BaseValidator 类	122
3.3 作为控件容器的页面	78	4.5.4 RequiredFieldValidator 控件	124
3.3.1 显示控件树	78	4.5.5 RangeValidator 控件	124
3.3.2 页头	82	4.5.6 CompareValidator 控件	125
3.3.3 动态控件的创建	82	4.5.7 RegularExpressionValidator 控件	125
3.4 Page 类	84	4.5.8 CustomValidator 控件	127
3.4.1 Session、Application 和 Cache	84	4.5.9 ValidationSummary 控件	128
3.4.2 Request	85	4.5.10 通过编程使用验证器	129
3.4.3 Response	86	4.5.11 验证组	130
3.4.4 Server	88	4.6 富控件	132
3.4.5 User	90	4.6.1 AdRotator 控件	132
3.4.6 Trace	91	4.6.2 Calendar 控件	134
3.4.7 访问其他类中的 HTTP 上下文	96	4.7 总结	136
3.5 总结	96	第 5 章 ASP.NET 应用程序	137
第 4 章 服务器控件	97	5.1 剖析 ASP.NET 应用程序	137
4.1 服务器控件的类型	97	5.1.1 应用程序域	137
4.2 HTML 服务器控件	99	5.1.2 应用程序生命周期	138
4.2.1 HtmlControl 类	100	5.1.3 应用程序更新	139
4.2.2 HtmlContainerControl 类	100	5.1.4 应用程序目录结构	139
4.2.3 HtmlInputControl 类	100	5.2 global.asax 应用程序文件	140
4.2.4 HTML 服务器控件类	101	5.2.1 应用程序事件	141
4.2.5 设置 Style 特性和其他属性	102	5.2.2 演示应用程序事件	143
4.2.6 通过编程创建服务器控件	103	5.3 ASP.NET 配置	144
4.2.7 处理服务器端事件	104	5.3.1 machine.config 文件	144
4.3 Web 控件	107	5.3.2 web.config 文件	147
4.3.1 WebControl 基类	107	5.3.3 <system.web>设置	150
4.3.2 基本的 Web 控件类	108	5.3.4 <system.webServer>	150
4.3.3 单位	109	5.3.5 <appSettings>	150
4.3.4 枚举	110	5.3.6 <connectionStrings>	152
4.3.5 颜色	110	5.3.7 通过编程读写配置节	152
4.3.6 字体	111	5.3.8 网站管理工具 (WAT)	155
4.3.7 焦点	112	5.3.9 扩展配置文件结构	156
4.3.8 默认按钮	113	5.3.10 加密配置节	159
4.3.9 可滚动面板	114	5.4 .NET 组件	161
4.3.10 处理 Web 控件事件	114	5.4.1 创建组件	161
4.4 List 控件	116	5.4.2 通过 App_Code 目录使用组件	162
4.4.1 Selectable 列表控件	117	5.4.3 通过 Bin 目录使用组件	163

5.5 扩展 HTTP 管道	165	7.2.3 连接池	215
5.5.1 HTTP 处理程序	165	7.3 Command 类和 DataReader 类	217
5.5.2 创建自定义的 HTTP 处理程序	167	7.3.1 Command 基础	217
5.5.3 配置自定义的 HTTP 处理程序	168	7.3.2 DataReader 类	218
5.5.4 使用无须配置的 HTTP 处理程序	169	7.3.3 ExecuteReader() 方法和 DataReader	218
5.5.5 创建高级的 HTTP 处理程序	169	7.3.4 ExecuteScalar() 方法	223
5.5.6 为非 HTML 内容创建 HTTP 处理程序	171	7.3.5 ExecuteNonQuery() 方法	223
5.5.7 HTTP 模块	173	7.3.6 SQL 注入攻击	224
5.5.8 创建自定义 HTTP 模块	174	7.3.7 使用参数化命令	226
5.6 总结	176	7.3.8 调用存储过程	227
第 6 章 状态管理	177	7.4 事务	229
6.1 ASP.NET 状态管理	177	7.4.1 事务和 ASP.NET 应用程序	230
6.2 视图状态	179	7.4.2 隔离级别	234
6.2.1 视图状态示例	180	7.4.3 保存点	235
6.2.2 在视图状态中保存对象	181	7.5 提供程序无关的代码	235
6.2.3 视图状态评估	183	7.5.1 创建工厂	236
6.2.4 有选择地禁用视图状态	184	7.5.2 用工厂创建对象	237
6.2.5 视图状态安全	185	7.5.3 使用提供程序无关的代码的查询	237
6.3 在页面间传递信息	186	7.6 总结	239
6.3.1 查询字符串	187	第 8 章 数据组件和 DataSet	240
6.3.2 跨页回发	188	8.1 构建数据访问组件	240
6.4 cookie	193	8.1.1 数据包	241
6.5 会话状态	194	8.1.2 存储过程	242
6.5.1 会话架构	194	8.1.3 数据应用类	243
6.5.2 使用会话状态	196	8.1.4 测试数据库组件	248
6.5.3 配置会话状态	197	8.2 非连接的数据	249
6.5.4 会话状态安全	202	8.2.1 Web 应用程序和 DataSet	250
6.6 应用程序状态	203	8.2.2 XML 集成	251
6.7 总结	206	8.3 DataSet 类	251
第二部分 数据访问		8.4 DataAdapter 类	252
第 7 章 ADO.NET 基础	208	8.4.1 填充 DataSet	253
7.1 ADO.NET 架构	209	8.4.2 使用多个表和关系	254
7.1.1 ADO.NET 数据提供程序	209	8.4.3 查找特定行	257
7.1.2 ADO.NET 的标准化	210	8.4.4 在数据访问类里使用 DataSet	257
7.1.3 基本 ADO.NET 类	211	8.4.5 数据绑定	258
7.2 Connection 类	212	8.5 DataView 类	259
7.2.1 连接字符串	212	8.5.1 使用 DataView 排序	259
7.2.2 测试连接	214	8.5.2 使用 DataView 过滤	261
		8.5.3 使用关系的高级过滤	262

8.5.4 计算列	263	10.4.3 排序和选择	319
8.6 总结	264	10.4.4 高级排序	319
第9章 数据绑定	265	10.5 GridView 分页	320
9.1 基本的数据绑定	265	10.5.1 自动分页	320
9.1.1 单值绑定	266	10.5.2 分页和选择	322
9.1.2 其他类型的表达式	268	10.5.3 ObjectDataSource 的自定义 分页	322
9.1.3 重复值绑定	271	10.5.4 定制分页栏	324
9.2 数据源控件	277	10.6 GridView 模板	325
9.3 SqlDataSource	278	10.6.1 使用多个模板	327
9.3.1 选择记录	279	10.6.2 在 Visual Studio 中编辑模板	328
9.3.2 参数化命令	281	10.6.3 绑定到方法	328
9.3.3 错误处理	285	10.6.4 处理模板中的事件	330
9.3.4 更新记录	286	10.6.5 使用模板编辑	330
9.3.5 删除记录	289	10.6.6 模板中的客户端 ID	335
9.3.6 插入记录	289	10.7 ListView	336
9.3.7 SqlDataSource 的不足	290	10.7.1 分组	338
9.4 ObjectDataSource	291	10.7.2 分页	340
9.4.1 选择记录	291	10.8 DetailsView 和 FormView	341
9.4.2 更新记录	295	10.8.1 DetailsView	341
9.4.3 使用数据对象执行更新	296	10.8.2 FormView	344
9.5 数据源控件的限制	299	10.9 高级网格	345
9.5.1 问题	299	10.9.1 GridView 中的汇总	345
9.5.2 添加其他项	300	10.9.2 单个表中的父/子视图	347
9.5.3 使用 SqlDataSource 处理其他 选项	301	10.9.3 使用查找表编辑字段	349
9.5.4 使用 ObjectDataSource 处理其 他选项	301	10.9.4 处理来自数据库的图片	351
9.6 总结	302	10.9.5 探测并发冲突	355
第10章 富数据控件	303	10.10 总结	359
10.1 GridView	303	第11章 缓存和异步页面	360
10.2 格式化 GridView	307	11.1 理解 ASP.NET 缓存	360
10.2.1 格式化字段	307	11.2 输出缓存	361
10.2.2 样式	308	11.2.1 声明性的输出缓存	361
10.2.3 格式化特定值	311	11.2.2 缓存和查询字符串	362
10.3 GridView 行选择	313	11.2.3 使用特定查询字符串参数的 缓存	363
10.3.1 使用选择来创建主-从表单	314	11.2.4 自定义缓存控制	364
10.3.2 SelectedIndexChanged 事件	315	11.2.5 使用 HttpCachePolicy 类进 行缓存	365
10.3.3 将数据字段用作选择按钮	315	11.2.6 缓存后替换和部分页缓存	365
10.4 对 GridView 排序	316	11.2.7 缓存用户配置	367
10.4.1 使用 SqlDataSource 排序	317	11.2.8 缓存配置	368
10.4.2 使用 ObjectDataSource 排序	318		

11.2.9 输出缓存扩展	369	12.2.5 压缩	423
11.3 数据缓存	373	12.3 序列化	423
11.3.1 向缓存添加项目	373	12.4 总结	426
11.3.2 简单的缓存测试	375	第 13 章 LINQ	427
11.3.3 缓存优先级	376	13.1 LINQ 基础	427
11.3.4 使用数据源控件的缓存	376	13.1.1 延迟执行	429
11.4 缓存依赖	379	13.1.2 LINQ 是如何工作的	429
11.4.1 文件和缓存项目依赖	379	13.1.3 LINQ 表达式	430
11.4.2 聚合依赖	380	13.1.4 揭秘 LINQ 表达式	436
11.4.3 移除项目回调	381	13.2 LINQ to DataSet	439
11.4.4 理解 SQL 缓存通知	383	13.2.1 强类型的 DataSet	441
11.4.5 缓存通知的工作方式	384	13.2.2 空值	441
11.4.6 启用通知	384	13.3 LINQ to Entities	441
11.4.7 创建缓存依赖	385	13.3.1 生成数据模型	442
11.5 自定义缓存依赖	386	13.3.2 数据模型类	442
11.5.1 基本的自定义缓存依赖	386	13.3.3 实体关系	444
11.5.2 使用消息队列的自定义缓存 依赖	387	13.3.4 查询存储过程	446
11.6 异步页面	389	13.3.5 LINQ to Entities 查询揭秘	447
11.6.1 创建异步页面	390	13.4 数据库操作	452
11.6.2 在异步页面中查询数据	392	13.4.1 插入	452
11.6.3 错误处理	393	13.4.2 更新	454
11.6.4 在异步任务中使用缓存	395	13.4.3 删除	454
11.6.5 多异步任务和超时	397	13.4.4 管理并发	455
11.7 总结	399	13.4.5 处理并发冲突	455
第 12 章 文件和流	400	13.5 EntityDataSource 控件	459
12.1 使用文件系统	400	13.5.1 显示数据	459
12.1.1 Directory 类和 File 类	400	13.5.2 获取关联数据	463
12.1.2 DirectoryInfo 类和 FileInfo 类	402	13.5.3 编辑数据	464
12.1.3 DriveInfo 类	404	13.5.4 验证	464
12.1.4 使用 Attributes	405	13.6 使用 QueryExtender 控件	465
12.1.5 使用通配符过滤文件	407	13.6.1 使用 SearchExpression	466
12.1.6 获取文件的版本信息	407	13.6.2 使用 RangeExpression	467
12.1.7 Path 类	408	13.6.3 使用 PropertyExpression	467
12.1.8 文件浏览器	410	13.6.4 使用 MethodExpression	468
12.2 使用流读写文件	414	13.7 总结	468
12.2.1 文本文件	415	第 14 章 XML	470
12.2.2 二进制文件	416	14.1 什么时候使用 XML	470
12.2.3 上传文件	417	14.2 XML 简介	471
12.2.4 使文件对多用户安全	419	14.2.1 XML 的优点	472
		14.2.2 格式良好的 XML	472
		14.2.3 XML 命名空间	473

14.2.4 XML 架构	474	15.1.2 把页面转换成用户控件	522
14.3 基于流的 XML 处理	475	15.2 为用户控件添加代码	523
14.3.1 写 XML 文件	476	15.2.1 处理事件	523
14.3.2 读取 XML 文件	478	15.2.2 添加属性	524
14.4 内存中的 XML 处理	481	15.2.3 使用自定义对象	525
14.4.1 XmlDocument	482	15.2.4 添加事件	528
14.4.2 XPathNavigator	485	15.2.5 公开内部 Web 控件	530
14.4.3 XDocument	486	15.3 动态加载用户控件	531
14.5 搜索 XML 内容	491	15.4 局部页面缓存	534
14.5.1 搜索 XmlDocument	491	15.4.1 VaryByControl	535
14.5.2 使用 XPath 搜索 XmlDocument	493	15.4.2 共享缓存控件	536
14.5.3 使用 LINQ 搜索 XDocument	494	15.5 总结	537
14.6 验证 XML 内容	496	第 16 章 主题和母版页	538
14.6.1 基本架构	496	16.1 层叠样式表	538
14.6.2 验证 XmlDocument	497	16.1.1 创建样式表	538
14.6.3 使用 XDocument 进行验证	498	16.1.2 应用样式表规则	540
14.7 转换 XML 内容	499	16.2 主题	542
14.7.1 基本的样式表	499	16.2.1 主题文件夹和外观	543
14.7.2 使用 XslCompiledTransform	500	16.2.2 应用简单主题	544
14.7.3 使用 Xml 控件	501	16.2.3 处理主题冲突	545
14.7.4 使用 LINQ to XML 转换 XML	502	16.2.4 为同一控件创建多个外观	546
14.8 XML 数据绑定	504	16.2.5 含有模板和图片的外观	547
14.8.1 非层次化绑定	504	16.2.6 在主题里使用 CSS	549
14.8.2 使用 XPath	506	16.2.7 通过配置文件来应用主题	549
14.8.3 嵌套的网格	508	16.2.8 动态应用主题	550
14.8.4 使用 TreeView 的层次化绑定	509	16.3 标准化网站布局	551
14.8.5 使用 XSLT	511	16.4 母版页基础	552
14.8.6 绑定到来自其他源的 XML 内容	512	16.4.1 简单的母版页	552
14.8.7 通过 XmlDataSource 更新 XML	513	16.4.2 简单的内容页	554
14.9 XML 和 ADO.NET 数据集	513	16.4.3 默认内容	556
14.9.1 把 DataSet 转换为 XML	514	16.4.4 具有表格和 CSS 布局的母版页	556
14.9.2 把 DataSet 作为 XML 访问	515	16.4.5 母版页和相对路径	559
14.10 总结	517	16.4.6 通过配置文件应用母版页	559
		16.5 高级母版页	560
		16.5.1 和母版页类交互	560
		16.5.2 动态设置母版页	561
		16.5.3 嵌套母版页	561
		16.6 总结	563
		第 17 章 网站导航	564
		17.1 多视图页面	564
第三部分 构建 ASP.NET 网站			
第 15 章 用户控件	520		
15.1 用户控件基础	520		
15.1.1 创建简单的用户控件	521		

17.1.1	MultiView 控件	565
17.1.2	Wizard 控件	568
17.2	站点地图	575
17.2.1	定义站点地图	576
17.2.2	绑定站点地图	577
17.2.3	导航路径	578
17.2.4	显示站点地图的一部分	580
17.2.5	站点地图对象	583
17.2.6	加入自定义站点地图信息	584
17.2.7	创建自定义的 SiteMapProvider	584
17.2.8	安全调整	590
17.3	URL 映射和路由	592
17.3.1	URL 映射	592
17.3.2	URL 路由	593
17.4	TreeView 控件	594
17.4.1	TreeNode	595
17.4.2	按需填充节点	597
17.4.3	TreeView 样式	598
17.5	Menu 控件	601
17.5.1	Menu 样式	603
17.5.2	Menu 模板	605
17.6	总结	606
第 18 章	网站部署	607
18.1	安装和配置 IIS	607
18.1.1	安装 IIS 7	607
18.1.2	管理 IIS 7	608
18.2	部署网站	610
18.2.1	通过复制文件进行部署	611
18.2.2	使用 Web 部署	614
18.2.3	通过 FTP 部署	621
18.3	管理网站	626
18.3.1	创建新站点	626
18.3.2	创建虚拟目录	627
18.3.3	使用 VirtualPathProvider	628
18.3.4	使用应用程序池	631
18.3.5	使用应用程序预热	633
18.4	扩展集成管道	634
18.4.1	创建处理程序	634
18.4.2	部署处理程序	635
18.4.3	配置处理程序	635
18.4.4	测试处理程序	636
18.5	总结	636
第四部分 安 全		
第 19 章	ASP.NET 安全模型	638
19.1	为何创建安全软件	638
19.1.1	理解潜在的威胁	638
19.1.2	安全编程原则	639
19.1.3	理解 Gatekeeper	640
19.2	理解安全级别	641
19.2.1	验证	641
19.2.2	授权	642
19.2.3	机密性和完整性	643
19.2.4	互相协作	643
19.3	理解 SSL	644
19.3.1	理解证书	645
19.3.2	理解 SSL	645
19.3.3	在 IIS 7.x 中配置 SSL	646
19.4	总结	649
第 20 章	表单验证	650
20.1	表单验证介绍	650
20.1.1	为什么使用表单验证	650
20.1.2	为什么不使用表单验证	652
20.1.3	为什么不自己实现 cookie 验证	653
20.1.4	表单验证类	654
20.2	实现表单验证	655
20.2.1	配置表单验证	655
20.2.2	禁止匿名用户访问	657
20.2.3	创建自定义登录页面	658
20.2.4	自定义凭证存储	662
20.2.5	表单验证中的持久化 cookie	663
20.3	IIS 7.x 和表单验证	665
20.4	总结	669
第 21 章	成员资格	670
21.1	介绍 ASP.NET 的成员资格 API	670
21.2	使用成员资格 API	672
21.2.1	配置表单验证	673
21.2.2	创建数据存储	674
21.2.3	配置连接字符串和成员资格 提供程序	679

21.2.4 创建并验证用户	681	23.3.2 使用 PrincipalPermission 类	740
21.3 使用安全控件	683	23.4 为基于角色的授权使用角色 API	742
21.3.1 Login 控件	684	23.4.1 和角色一起使用 LoginView 控件	747
21.3.2 LoginStatus 控件	693	23.4.2 通过编程访问角色	748
21.3.3 LoginView 控件	693	23.4.3 在 Windows 验证中使用角色 API	750
21.3.4 PasswordRecovery 控件	694	23.5 IIS 7.x 里的授权和角色	752
21.3.5 ChangePassword 控件	697	23.5.1 在 IIS 7.x 里使用 ASP.NET 角色授权	754
21.3.6 CreateUserWizard 控件	698	23.5.2 在 IIS 7.x 中管理 ASP.NET 角色	756
21.4 在 IIS 7.x 里配置成员资格	702	23.6 总结	757
21.4.1 配置提供程序和用户	702	第 24 章 用户配置	758
21.4.2 在其他应用程序里使用成员 资格 API	703	24.1 理解用户配置	758
21.5 使用 Membership 类	705	24.1.1 用户配置的性能	758
21.5.1 从存储中获得用户	706	24.1.2 用户配置如何存储数据	759
21.5.2 更新存储中的用户	708	24.1.3 用户配置和验证	760
21.5.3 创建和删除用户	708	24.1.4 用户配置与自定义的数据 组件	760
21.5.4 检验用户	709	24.2 使用 SqlProfileProvider	761
21.6 总结	709	24.2.1 创建用户配置表	761
第 22 章 Windows 验证	710	24.2.2 配置提供程序	763
22.1 介绍 Windows 验证	710	24.2.3 定义用户配置属性	764
22.1.1 为什么使用 Windows 验证	710	24.2.4 使用用户配置属性	764
22.1.2 为什么不使用 Windows 验证	711	24.2.5 用户配置序列化	766
22.1.3 Windows 验证机制	712	24.2.6 用户配置组	767
22.2 实现 Windows 验证	717	24.2.7 用户配置和自定义的数据 类型	768
22.2.1 配置 IIS 7.x	717	24.2.8 用户配置 API	771
22.2.2 配置 ASP.NET	718	24.2.9 匿名用户配置	773
22.2.3 IIS 7.x 管道	719	24.3 自定义的用户配置提供程序	775
22.2.4 禁止匿名用户访问	722	24.3.1 自定义的用户配置提供程序 类	775
22.2.5 访问 Windows 用户信息	723	24.3.2 设计 FactoredProfileProvider	777
22.3 身份模拟	727	24.3.3 通过编码实现 FactoredProfileProvider	778
22.3.1 Windows 中的身份模拟和 委托	728	24.3.4 测试 FactoredProfileProvider	781
22.3.2 经过配置的身份模拟	729	24.4 总结	784
22.3.3 可编程的身份模拟	729	第 25 章 加密	785
22.4 总结	732	25.1 加密数据: 机密性问题	785
第 23 章 授权和角色	733		
23.1 URL 授权	733		
23.2 文件授权	738		
23.3 在代码中检查授权	739		
23.3.1 使用 IsInRole() 方法	739		

第五部分 高级用户界面

29.3 脚本注入攻击.....	920	30.5 控件扩展器.....	991
29.3.1 请求验证.....	921	30.5.1 安装 ASP.NET AJAX 控件 工具包.....	992
29.3.2 禁用请求验证.....	922	30.5.2 AutoCompleteExtender.....	993
29.3.3 扩展请求验证.....	923	30.5.3 ASP.NET AJAX 控件工具包.....	996
29.4 带有 JavaScript 的自定义控件.....	925	30.6 总结.....	999
29.4.1 弹出窗口.....	925	第 31 章 使用 Web 部件页面的门户	
29.4.2 滚动按钮.....	928	网站.....	1000
29.5 框架.....	931	31.1 典型门户页面.....	1000
29.5.1 框架导航.....	932	31.2 基本 Web 部件页面.....	1002
29.5.2 内联框架.....	933	31.2.1 创建页面设计.....	1002
29.6 理解 Ajax.....	934	31.2.2 WebPartManager 和 WebPartZone.....	1003
29.6.1 XMLHttpRequest 对象.....	935	31.2.3 向页面添加 Web 部件.....	1004
29.6.2 Ajax 示例.....	936	31.2.4 定制页面.....	1008
29.7 在客户端回调中使用 Ajax.....	939	31.3 创建 Web 部件.....	1010
29.7.1 创建客户端回调.....	940	31.3.1 简单的 Web 部件任务.....	1010
29.7.2 深入了解客户端回调.....	945	31.3.2 开发高级 Web 部件.....	1017
29.7.3 自定义控件里的客户端回调.....	946	31.3.3 Web 部件编辑器.....	1024
29.8 总结.....	949	31.3.4 连接 Web 部件.....	1029
第 30 章 ASP.NET AJAX	950	31.3.5 自定义动词和 Web 部件.....	1036
30.1 介绍 ASP.NET AJAX.....	950	31.3.6 用户控件和高级 Web 部件.....	1037
30.1.1 客户端的 ASP.NET AJAX: 脚本库.....	951	31.3.7 动态上传 Web 部件.....	1040
30.1.2 服务器端的 ASP.NET AJAX: ScriptManager.....	952	31.3.8 对 Web 部件的授权.....	1044
30.2 服务器回调.....	952	31.3.9 个性化的最后任务.....	1045
30.2.1 ASP.NET AJAX 中的 Web 服务.....	953	31.4 总结.....	1046
30.2.2 在页面里放置 Web 方法.....	959	第 32 章 MVC	1047
30.2.3 ASP.NET AJAX 应用程序 服务.....	960	32.1 MVC 和 Web 表单的选择.....	1047
30.3 ASP.NET AJAX 服务器控件.....	966	32.2 创建基本的 MVC 应用程序.....	1047
30.3.1 使用 UpdatePanel 的局部 呈现.....	966	32.2.1 创建模型.....	1048
30.3.2 使用 Timer 的定时刷新.....	973	32.2.2 创建控制器.....	1049
30.3.3 使用 UpdateProgress 的耗时 更新.....	974	32.2.3 创建 Index 视图.....	1050
30.3.4 管理浏览器历史.....	976	32.2.4 测试 (未完成的) 应用程序.....	1050
30.4 深入客户端库.....	979	32.2.5 完成控制器和视图.....	1051
30.4.1 理解客户端模型.....	980	32.2.6 修改 Site.Master 文件.....	1053
30.4.2 JavaScript 的面向对象编程.....	981	32.3 扩展基础的 MVC 应用程序.....	1053
30.4.3 网页框架.....	987	32.3.1 配置路由.....	1054
		32.3.2 增加错误处理.....	1055
		32.3.3 增加验证.....	1056
		32.3.4 增强数据存储访问.....	1057
		32.3.5 增加对外键约束的支持.....	1059