

Pro ASP.NET 3.5 in C# 2008 **Second Edition**

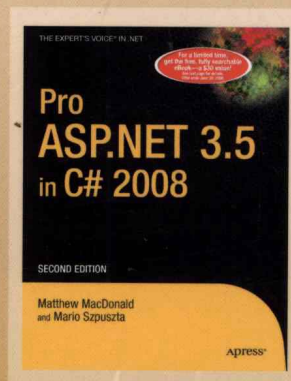
ASP.NET 3.5

高级程序设计 (第2版)

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



- ASP.NET经典著作之屠龙刀，
涵盖VS 2008和IIS 7.0
- 全面、深入、贴近实战，
让你感受世界顶尖专家的不凡功力
- 国内MVP担纲翻译，名著名译，相得益彰



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书 微软技术系列

Pro ASP.NET 3.5 in C# 2008 **Second Edition**

ASP.NET 3.5

高级程序设计 (第2版)

[美] Matthew MacDonald 著
Mario Szpuszta

博思工作室 译

人民邮电出版社
北 京

图书在版编目(CIP)数据

ASP.NET 3.5 高级程序设计: 第2版 / (美) 麦克唐纳
(MacDonald, M.), (美) 兹普兹塔 (Szpuszta, M.) 著;
博思工作室译. —北京: 人民邮电出版社, 2008.11
(图灵程序设计丛书)
ISBN 978-7-115-18771-0

I. A… II. ①麦…②兹…③博… III. 主页制作—程序
设计IV. TP393.092

中国版本图书馆CIP数据核字(2008)第136013号

内 容 提 要

本书是 ASP.NET 领域的鸿篇巨制, 全面讲解了 ASP.NET 3.5 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, 还介绍了微软的新一代 RIA 技术 Silverlight。

本书适用于各层次 ASP.NET 3.5 程序员。

图灵程序设计丛书

ASP.NET 3.5高级程序设计 (第2版)

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

责任编辑 王军花

◆ 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>

北京铭成印刷有限公司印刷

◆ 开本: 800×1000 1/16

印张: 76

字数: 2034千字 2008年11月第1版

印数: 1—4 000册 2008年11月北京第1次印刷

著作权合同登记号 图字: 01-2008-2033号

ISBN 978-7-115-18771-0/TP

定价: 148.00元

读者服务热线: (010)88593802 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 3.5 in C# 2008, Second Edition* by Matthew MacDonald and Mario Szpuszta by Apress, 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705 USA.

Copyright © 2007 by Matthew MacDonald and Mario Szpuszta. Simplified Chinese-language edition copyright © 2008 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。

译者序

随着2001年.NET技术的正式发布，微软正式拉开了下一代开发技术的序幕。.NET技术一经问世，就势不可挡地统治了整个Windows平台的开发领域。.NET技术横跨桌面应用开发、Web应用程序开发和嵌入式开发的各个领域，并提供了一个统一高效的集成开发环境Visual Studio。

在2007年11月，微软正式发布了Visual Studio 2008以及ASP.NET 3.5版本，这是一个更加重要的升级和里程碑版本。

Visual Studio 2008允许你构建针对多个.NET Framework版本的应用。这意味着你可以使用Visual Studio 2008打开、编辑和编译现有的.NET 2.0和ASP.NET 2.0应用（包括使用了ASP.NET AJAX 1.0的ASP.NET 2.0应用），继续在.NET 2.0机器上部署这些应用。

ASP.NET AJAX已经成为.NET 3.5的一部分，不再需要单独下载了。除了包括ASP.NET AJAX 1.0的所有功能外，ASP.NET 3.5现在还包括下面这些丰富的支持：集成了WebParts的UpdatePanel，与<asp:menu>和<asp:treeview>等控件的ASP.NET AJAX集成，WCF对JSON的支持，以及许许多多AJAX方面的改进。

LINQ to SQL是.NET 3.5中内置的对象关系映射（ORM）。它允许你使用.NET对象模型对关系数据库进行建模。然后你可以使用LINQ对数据库进行查询，并能更新、插入和删除数据。LINQ to SQL完全支持事务、视图和存储过程。它还提供了一个简易方式把业务逻辑和验证规则结合进你的数据模型。

ASP.NET 3.5还有更多激动人心的功能，在此我们不一一描述了，相信随着不断深入了解ASP.NET 3.5，你将会愈发喜欢ASP.NET 3.5这个版本的各种重要更新以及强大的新功能。

本书尝试对最新的ASP.NET 3.5技术做一个较为全面的阐述，由于时间仓促，难免有翻译欠妥之处，希望读者能谅解。

如果有任何建议以及错误反馈，请联系：uestc95@gmail.com

博思工作室

译者简介

苏红超 网名Uestc95，喜好.NET技术，曾著有《ASP.NET深入解析》（科学出版社，2003），并于2004/2005/2006/2007连续4年荣获微软MVP，目前主要从事超大规模Web应用研究以及Web安全研究。E-mail/MSN: uestc95@gmail.com

朱琦 资深.NET开发人员，企业顾问，主持研发数个大型企业级软件产品，目前就职于明基研发中心。E-mail/MSN: miraculeux@gmail.com

引言

毫无疑问，你已经知道ASP.NET是微软创建服务器端Web应用程序的下一代技术。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，你要用它去开发高性能的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的设计原理是通过一套快速但不清晰的工具集来将动态内容插入到普通Web页面。相比较而言，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 3.5所需的相关知识。如果你曾使用前一个版本的ASP.NET进行过编程，那么可以将重点放在学习ASP.NET 3.5的新特性上，例如第13章的LINQ、第32章的ASP.NET AJAX和第33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，将引导你逐步地学习基础知识和隐藏在ASP.NET技术背后的工作原理。阅读本书的唯一前提是，需具备良好的C#语言基础并了解基本的.NET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#，你会发现，先看介绍.NET基础知识的其他书籍会更好一些，比如，Andrew Troelsen所著的《C#2008与.NET 3.5高级程序设计（第4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，并说明了ASP.NET 3.5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～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核心问题之一——访问及操作数据。第7章和第8章介绍ADO.NET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的一整套数据访问控件，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～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章会涉及两个新的ASP.NET革新——主题（自动控制控件样式）和母版页（可

^① 人民邮电出版社即将出版。——编者注

供多个页面重用的布局模板)。第17章展示了ASP.NET如何使用新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章讲述如何部署技术和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。先是第19章简要介绍安全概念,然后第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API (profile API),这是一个用来存储用户特定信息的新的、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章和第28章介绍自定义控件技术。第29章介绍手工控制图片的GDI+技术。最后,第30章展示了用于构建Web门户的ASP.NET的Web部件架构。

第六部分 这部分将展示现代Web开发中最精彩的创新。首先,第31章和第32章介绍如何在ASP.NET Web页面里使用JavaScript和Ajax技术。我们将学习如何让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33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,否则,那样的一本书可能会有本书的两倍厚。相反,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,你就可以通过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,而且还可以学到实际将用到的技术,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。

为了更好地理解本书,你最好熟悉C#语法以及面向对象的概念。你不一定非要使用过老版本的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,你需要考虑同时阅读一些介绍.NET的书籍,比如Andrew Troelsen的《C#2008与.NET 3.5高级程序设计(第4版)》。

学习本书的硬件和软件环境

你需要一个安装有Visual Studio 2008的电脑来配合阅读本书。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。

注解 你可以使用精简了功能的Visual Studio Web Developer 2008 Express版本,但这样就不能运行某些示例代码了。最重要的是无法使用Visual Studio Web Developer 2008 Express版本来创建类库,而这是现代面向组件设计的核心所在(但是可以通过使用两个Express版本解决这个限制,即使用Visual Studio Web Developer Express版本创建页面,使用Visual C# 2008 Express版本创建组件)。

此外，如果要承载ASP.NET网站，你需要使用Windows XP Professional或者（理想情况下）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。IIS在第18章中介绍。

最后，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、安全技术以及其他功能。你可以使用任何版本的SQL Server来演示这些例子，比如SQL Server 2005 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://msdn.microsoft.com/sql/express>上免费下载）。如果你使用其他关系数据库引擎，就需要修改示例代码。

客户支持

我们非常重视读者的反馈，我们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，还有你希望我们应该在下次改进的地方。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的站点<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，需要解压缩下载文件。代码是按章节顺序排列的，在使用代码之前，请阅读readme.txt文件来了解相关注意事项。

附录章节

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。有如下一些内容。

- **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，则本章内容对于开发人员十分重要。
- **Bonus Chapter 2, 3, 4.** 这些章节是关于Web服务的，Web服务让你能够创建由Internet上的其他应用程序调用的子程序。在富客户端开发中Web服务更有意思一些（因为它可以为普通的桌面应用程序带来Web特性），而且它们目前正在被一项叫做WCF（Windows Communication Foundation，Windows通信基础）的技术所取代。由于这些原因，本书不讨论Web服务。

注解 与Web服务相关的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 3.5依然适用，因为Web服务特性在3.5中没有发生任何变化。

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。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	2.1.4 创建无项目文件的网站	20
1.1 Web 开发技术的演进	2	2.1.5 多目标	23
1.1.1 早期的 Web 开发	2	2.1.6 设计网页	24
1.1.2 传统 ASP 的不足	3	2.2 Visual Studio 集成开发环境	28
1.1.3 ASP.NET	4	2.2.1 解决方案资源管理器	30
1.2 ASP.NET 的 7 大要点	5	2.2.2 文档窗口	31
1.2.1 要点 1: ASP.NET 同 .NET Framework 集成在一起	5	2.2.3 工具箱	31
1.2.2 要点 2: ASP.NET 是编译执行的, 而不是解释执行的	5	2.2.4 错误列表和任务列表	32
1.2.3 要点 3: ASP.NET 具备多语言支持	7	2.2.5 服务器资源管理器	33
1.2.4 要点 4: ASP.NET 运行在公共语言运行库内	9	2.3 代码编辑器	34
1.2.5 要点 5: ASP.NET 是面向对象的	10	2.3.1 添加程序集引用	35
1.2.6 要点 6: ASP.NET 是跨设备和跨浏览器的	11	2.3.2 智能感知和大纲显示	36
1.2.7 要点 7: ASP.NET 易于部署和配置	11	2.4 编码模型	38
1.3 ASP.NET 3.5: 故事在继续	12	2.4.1 代码隐藏文件如何与页面连接	40
1.3.1 ASP.NET 2.0	12	2.4.2 控件标签如何与页面变量连接	41
1.3.2 ASP.NET 3.5	13	2.4.3 事件如何与事件处理程序连接	42
1.3.3 Silverlight	17	2.5 Web 项目	43
1.4 总结	17	2.5.1 创建 Web 项目	45
第2章 Visual Studio	18	2.5.2 迁移旧版 Visual Studio 创建的网站	46
2.1 .NET 开发模型	19	2.6 Visual Studio 调试	47
2.1.1 编译器	19	2.6.1 单步调试	48
2.1.2 Visual Studio 集成开发环境	19	2.6.2 变量监视	50
2.1.3 网站和 Web 项目	20	2.6.3 高级断点	51
		2.7 Visual Studio 宏	51
		2.8 Web Development Helper	53
		2.9 总结	54
		第3章 Web 窗体	55
		3.1 页面处理	55
		3.1.1 HTML 表单	55

3.1.2 动态用户界面	57	4.3.2 基本的 Web 控件类	101
3.1.3 ASP.NET 事件模型	58	4.3.3 单位	102
3.1.4 自动回送	58	4.3.4 枚举	102
3.1.5 视图状态	60	4.3.5 颜色	103
3.1.6 XHTML 兼容性	63	4.3.6 字体	103
3.2 Web 窗体处理阶段	67	4.3.7 焦点	104
3.2.1 页面框架初始化	67	4.3.8 默认按钮	106
3.2.2 用户代码初始化	68	4.3.9 可滚动面板	106
3.2.3 验证	68	4.3.10 处理 Web 控件事件	107
3.2.4 事件处理	68	4.4 List 控件	109
3.2.5 自动数据绑定	69	4.4.1 Selectable 列表控件	110
3.2.6 清除	69	4.4.2 BulletedList 控件	112
3.2.7 页面流示例	70	4.5 输入验证控件	113
3.3 作为控件容器的页面	72	4.5.1 验证控件	113
3.3.1 显示控件树	72	4.5.2 验证流程	114
3.3.2 页头	76	4.5.3 BaseValidator 类	115
3.3.3 动态控件的创建	76	4.5.4 RequiredFieldValidator 控件	116
3.4 Page 类	78	4.5.5 RangeValidator 控件	116
3.4.1 Session、Application 和 Cache	78	4.5.6 CompareValidator 控件	117
3.4.2 Request	79	4.5.7 RegularExpressionValidator 控件	117
3.4.3 Response	80	4.5.8 CustomValidator 控件	119
3.4.4 Server	81	4.5.9 ValidationSummary 控件	120
3.4.5 User	83	4.5.10 通过编程使用验证器	121
3.4.6 Trace	84	4.5.11 验证组	122
3.4.7 访问其他类中的 HTTP 上下文	89	4.6 富控件	124
3.5 总结	89	4.6.1 AdRotator 控件	125
第 4 章 服务器控件	90	4.6.2 Calendar 控件	126
4.1 服务器控件的类型	90	4.7 总结	128
4.2 HTML 服务器控件	92	第 5 章 ASP.NET 应用程序	129
4.2.1 HtmlControl 类	93	5.1 剖析 ASP.NET 应用程序	129
4.2.2 HtmlContainerControl 类	93	5.1.1 应用程序域	129
4.2.3 HtmlInputControl 类	93	5.1.2 应用程序生命期	130
4.2.4 HTML 服务器控件类	94	5.1.3 应用程序更新	131
4.2.5 设置 Style 特性和其他属性	95	5.1.4 应用程序目录结构	131
4.2.6 编程创建服务器控件	96	5.2 global.asax 应用程序文件	132
4.2.7 处理服务器端事件	97	5.2.1 应用程序事件	133
4.3 Web 控件	99	5.2.2 演示应用程序事件	135
4.3.1 WebControl 基类	100	5.3 ASP.NET 配置	136
		5.3.1 machine.config 文件	136

5.3.2	web.config 文件	139	6.7	总结	199
5.3.3	<system.web>设置	142		第二部分 数据访问	
5.3.4	通过编程读写配置节	145		第7章 ADO.NET 基础	202
5.3.5	网站管理工具 (WAT)	148	7.1	ADO.NET 架构	203
5.3.6	扩展配置文件结构	149	7.1.1	ADO.NET 数据提供程序	203
5.3.7	加密配置节	153	7.1.2	ADO.NET 的标准化	204
5.4	.NET 组件	154	7.1.3	SQL Server 2005	205
5.4.1	创建组件	155	7.1.4	基本 ADO.NET 类	206
5.4.2	通过 App_Code 目录使用组件	156	7.2	Connection 类	206
5.4.3	通过 Bin 目录使用组件	157	7.2.1	连接字符串	207
5.5	扩展 HTTP 管道	159	7.2.2	测试连接	208
5.5.1	HTTP 处理程序和 HTTP 模块	159	7.2.3	连接池	209
5.5.2	创建自定义的 HTTP 处理程序	161	7.2.4	连接统计	210
5.5.3	配置自定义的 HTTP 处理程序	161	7.3	Command 类和 DataReader 类	211
5.5.4	不配置 IIS 的 HTTP 注册处理 程序	163	7.3.1	Command 基础	211
5.5.5	创建高级的 HTTP 处理程序	164	7.3.2	DataReader 类	212
5.5.6	为非 HTML 内容创建 HTTP 处理程序	165	7.3.3	ExecuteReader()方法和 DataReader	213
5.5.7	创建自定义 HTTP 模块	168	7.3.4	ExecuteScalar()方法	217
5.6	总结	170	7.3.5	ExecuteNonQuery()方法	217
第6章 状态管理	171		7.3.6	SQL 注入攻击	218
6.1	ASP.NET 状态管理	171	7.3.7	使用参数化命令	220
6.2	视图状态	173	7.3.8	调用存储过程	221
6.2.1	视图状态示例	173	7.4	事务	224
6.2.2	在视图状态中保存对象	175	7.4.1	事务和 ASP.NET 应用程序	224
6.2.3	保持成员变量	177	7.4.2	隔离级别	228
6.2.4	视图状态评估	178	7.4.3	保存点	229
6.2.5	视图状态安全	179	7.5	提供程序无关的代码	230
6.3	在页面间传送信息	180	7.5.1	创建工厂	230
6.3.1	查询字符串	180	7.5.2	用工厂创建对象	231
6.3.2	跨页面发送	182	7.5.3	使用程序无关代码的查询	232
6.4	cookie	187	7.6	总结	233
6.5	会话状态	188		第8章 数据组件和 DataSet	234
6.5.1	会话架构	188	8.1	构建数据访问组件	234
6.5.2	使用会话状态	190	8.1.1	数据包	235
6.5.3	配置会话状态	190	8.1.2	存储过程	236
6.5.4	会话状态安全	195			
6.6	应用程序状态	196			

8.1.3 数据应用类	237	9.4.1 选择记录	293
8.1.4 测试数据库组件	242	9.4.2 更新记录	297
8.2 非连接的数据	244	9.4.3 使用数据对象执行更新	298
8.2.1 Web 应用程序和 DataSet	245	9.5 数据源控件的限制	301
8.2.2 XML 集成	246	9.5.1 问题	301
8.3 DataSet 类	246	9.5.2 添加其他项目	302
8.4 DataAdapter 类	247	9.5.3 使用 SqlDataSource 处理其他 选项	303
8.4.1 填充 DataSet	248	9.5.4 使用 ObjectDataSource 处理其他 选项	303
8.4.2 使用多个表和关系	249	9.6 总结	304
8.4.3 查找特定行	252	第 10 章 富数据控件	305
8.4.4 在数据访问类里使用 DataSet	252	10.1 GridView	306
8.4.5 数据绑定	253	10.2 格式化 GridView	309
8.5 DataView 类	254	10.2.1 格式化字段	309
8.5.1 使用 DataView 排序	254	10.2.2 样式	310
8.5.2 使用 DataView 过滤	256	10.2.3 格式化特定值	313
8.5.3 使用关系的高级过滤	257	10.3 GridView 行选择	314
8.5.4 计算列	258	10.3.1 使用选择来创建主-从表单	315
8.6 强类型 DataSet	259	10.3.2 SelectedIndexChanged 事件	317
8.6.1 自定义 TableAdapter	260	10.3.3 将数据字段用作选择按钮	317
8.6.2 创建强类型数据集	261	10.4 对 GridView 排序	318
8.6.3 深入强类型数据集	262	10.4.1 使用 SqlDataSource 排序	318
8.6.4 使用强类型数据集	264	10.4.2 使用 ObjectDataSource 排序	319
8.7 总结	266	10.4.3 排序和选择	320
第 9 章 数据绑定	267	10.4.4 高级排序	321
9.1 基本的数据绑定	267	10.5 GridView 分页	322
9.1.1 单值绑定	268	10.5.1 自动分页	322
9.1.2 其他类型的表达式	270	10.5.2 ObjectDataSource 的自定义 分页	323
9.1.3 重复值绑定	273	10.5.3 自定义分页栏	326
9.2 数据源控件	279	10.6 GridView 模板	327
9.3 SqlDataSource	280	10.6.1 使用多个模板	329
9.3.1 选择记录	281	10.6.2 在 Visual Studio 中编辑模板	329
9.3.2 参数化命令	284	10.6.3 绑定到方法	330
9.3.3 错误处理	287	10.6.4 处理模板中的事件	331
9.3.4 更新记录	288	10.6.5 使用模板编辑	332
9.3.5 删除记录	291	10.7 ListView	337
9.3.6 插入记录	291	10.7.1 分组	339
9.3.7 SqlDataSource 的不足	292		
9.4 ObjectDataSource	293		

10.7.2 分页	341	11.5.1 基本的自定义缓存依赖	387
10.8 DetailsView 和 FormView	342	11.5.2 使用消息队列的自定义缓存 依赖	388
10.8.1 DetailsView	342	11.6 异步页面	390
10.8.2 FormView	344	11.6.1 创建异步页面	391
10.9 高级网格	346	11.6.2 在异步页面中查询数据	393
10.9.1 GridView 中的汇总	346	11.6.3 错误处理	394
10.9.2 单个表中的父/子视图	347	11.6.4 在异步任务中使用缓存	397
10.9.3 使用检查表编辑字段	349	11.6.5 多异步任务和超时	399
10.9.4 处理来自数据库的图片	351	11.7 总结	400
10.9.5 探测并发冲突	356	第 12 章 文件和流	401
10.10 总结	360	12.1 使用文件系统	401
第 11 章 缓存和异步页面	361	12.1.1 Directory 类和 File 类	401
11.1 理解 ASP.NET 缓存	361	12.1.2 DirectoryInfo 类和 FileInfo 类	403
11.2 输出缓存	362	12.1.3 DriveInfo 类	406
11.2.1 声明性的输出缓存	362	12.1.4 使用 Attributes	406
11.2.2 缓存和查询字符串	363	12.1.5 使用通配符过滤文件	408
11.2.3 使用特定查询字符串参数的 缓存	364	12.1.6 获取文件的版本信息	408
11.2.4 自定义缓存控制	364	12.1.7 Path 类	409
11.2.5 使用 HttpCachePolicy 类 进行缓存	366	12.1.8 文件浏览器	411
11.2.6 缓存后替换和部分页缓存	366	12.2 使用流读写文件	415
11.2.7 缓存用户配置	368	12.2.1 文本文件	416
11.2.8 缓存配置	369	12.2.2 二进制文件	417
11.3 数据缓存	370	12.2.3 上传文件	418
11.3.1 向缓存添加项目	370	12.2.4 使文件对多用户安全	420
11.3.2 简单的缓存测试	372	12.2.5 压缩	424
11.3.3 缓存优先级	373	12.3 序列化	425
11.3.4 使用数据源控件的缓存	374	12.4 总结	428
11.4 缓存依赖	376	第 13 章 LINQ	429
11.4.1 文件和缓存项目依赖	376	13.1 LINQ 基础	429
11.4.2 聚合依赖	378	13.1.1 延迟执行	431
11.4.3 移除项目回调	378	13.1.2 LINQ 是如何工作的	431
11.4.4 理解 SQL 缓存通知	380	13.1.3 LINQ 表达式	432
11.4.5 SQL Server 2000 和 SQL Server 7 中的缓存通知	381	13.1.4 揭秘 LINQ 表达式	438
11.4.6 SQL Server 2005 和 SQL Server 2008 中的缓存通知	385	13.2 LINQ to DataSet	440
11.5 自定义缓存依赖	387	13.2.1 强类型的 DataSet	442
		13.2.2 空值	443

13.3 LINQ to SQL	443	14.7 转换 XML 内容	508
13.3.1 数据实体类	445	14.7.1 基本的样式表	509
13.3.2 DataContext	446	14.7.2 使用 XslCompiledTransform	510
13.3.3 揭秘 LINQ to SQL 查询	447	14.7.3 使用 Xml 控件	511
13.3.4 LINQ to SQL 和数据库组件	449	14.7.4 使用 LINQ to XML 转换 XML	511
13.3.5 选取单个记录或值	452	14.8 XML 数据绑定	513
13.3.6 自动生成数据类	453	14.8.1 非层次化绑定	514
13.3.7 关系	458	14.8.2 使用 XPath	515
13.3.8 为存储过程生成方法	464	14.8.3 嵌套的网格	517
13.3.9 提交变更	466	14.8.4 使用 TreeView 的层次化绑定	519
13.4 LinqDataSource	471	14.8.5 使用 XSLT	520
13.4.1 显示数据	472	14.8.6 绑定到来自其他源的 XML 内容	522
13.4.2 获取关联数据	474	14.8.7 通过 XmlDataSource 更新 XML	522
13.4.3 编辑数据	475	14.9 XML 和 ADO.NET 数据集	523
13.4.4 验证	475	14.9.1 把 DataSet 转换为 XML	523
13.5 总结	478	14.9.2 把 DataSet 作为 XML 访问	525
第 14 章 XML	479	14.10 总结	526
14.1 什么时候使用 XML	479	第三部分 构建 ASP.NET 网站	
14.2 XML 简介	480	第 15 章 用户控件	528
14.2.1 XML 的优点	481	15.1 用户控件基础	528
14.2.2 格式良好的 XML	481	15.1.1 创建简单的用户控件	529
14.2.3 XML 命名空间	482	15.1.2 把页面转换为用户控件	530
14.2.4 XML 架构	483	15.2 为用户控件添加代码	531
14.3 基于流的 XML 处理	485	15.2.1 处理事件	531
14.3.1 写 XML 文件	485	15.2.2 添加属性	532
14.3.2 读 XML 文件	488	15.2.3 使用自定义对象	533
14.4 内存中的 XML 处理	490	15.2.4 添加事件	536
14.4.1 XmlDocument	491	15.2.5 暴露内部 Web 控件	538
14.4.2 XPathNavigator	494	15.3 动态加载用户控件	539
14.4.3 XDocument	496	15.4 缓存部分页面	542
14.5 搜索 XML 内容	500	15.4.1 VaryByControl	543
14.5.1 搜索 XmlDocument	501	15.4.2 共享缓存控件	544
14.5.2 使用 XPath 搜索 XmlDocument	502	15.5 总结	545
14.5.3 使用 LINQ 搜索 XDocument	504	第 16 章 主题和母版页	546
14.6 验证 XML 内容	506	16.1 层叠样式表	546
14.6.1 基本架构	506		
14.6.2 验证 XmlDocument	506		
14.6.3 使用 XDocument 进行验证	508		

16.1.1	创建 CSS	546	17.3	TreeView 控件	601
16.1.2	应用样式表规则	548	17.3.1	TreeNode	601
16.2	主题	550	17.3.2	按需填充结点	604
16.2.1	主题文件夹和外观	551	17.3.3	TreeView 样式	605
16.2.2	应用简单主题	552	17.4	Menu 控件	608
16.2.3	处理主题冲突	553	17.4.1	Menu 样式	610
16.2.4	为控件创建多个外观	554	17.4.2	Menu 模板	612
16.2.5	含有模板和图片的外观	555	17.5	总结	613
16.2.6	在主题里使用 CSS	556	第 18 章	网站部署	614
16.2.7	通过配置文件来应用主题	557	18.1	IIS	614
16.2.8	动态应用主题	557	18.1.1	IIS 站点和虚拟目录	615
16.3	标准化网站布局	559	18.1.2	IIS 管理控制台和 IIS 配置	616
16.4	母版页基础	560	18.1.3	将网址、虚拟目录和文件 映射到 URL	617
16.4.1	简单的母版页	560	18.1.4	深入 IIS 架构	618
16.4.2	简单的内容页	562	18.1.5	安装 IIS	630
16.4.3	默认内容	563	18.2	管理网站	635
16.4.4	具有表格和 CSS 布局的 母版页	564	18.2.1	管理虚拟目录以及使用 IIS 5.x 和 IIS 6.0 的网站	636
16.4.5	母版页和相对路径	566	18.2.2	管理 IIS 6.0 中的应用程序池	640
16.4.6	通过配置文件应用母版页	567	18.2.3	在 IIS 7.0 里管理虚拟目录和 站点	645
16.5	高级母版页	567	18.2.4	在 IIS 7.0 里管理应用 程序池	655
16.5.1	和母版页类交互	567	18.3	部署你的 ASP.NET 应用程序	657
16.5.2	动态设置母版页	569	18.3.1	在 IIS 5.x 和 IIS 6.0 上验证 ASP.NET 安装	658
16.5.3	嵌套母版页	569	18.3.2	在 IIS 5.x 和 IIS 6.0 上 ASP.NET 并行执行	659
16.6	总结	571	18.3.3	在 IIS 7.0 里的 ASP.NET 并行 执行	660
第 17 章	网站导航	572	18.3.4	在 IIS 5.x 上部署时配置 HTTP 运行时设置	661
17.1	多视图页面	572	18.3.5	在 ASP.NET 中编译模型	661
17.1.1	MultiView 控件	573	18.3.6	使用 Visual Studio 进行部署	664
17.1.2	Wizard 控件	576	18.3.7	Visual Studio 2005 Web 部署 项目	665
17.2	站点地图	583	18.4	ASP.NET 中的 VirtualPathProvider	670
17.2.1	定义站点地图	583	18.5	ASP.NET 里的健康监测	674
17.2.2	绑定站点地图	585			
17.2.3	导航路径	586			
17.2.4	显示站点地图的一部分	587			
17.2.5	站点地图对象	591			
17.2.6	加入自定义站点地图信息	592			
17.2.7	创建自定义的 SiteMapProvider	592			
17.2.8	URL 映射	599			
17.2.9	安全调整	599			

18.5.1 理解基本结构	674	20.2.5 表单验证中的持久 cookie	721
18.5.2 事件和提供程序	675	20.3 IIS 7.0 和表单验证	722
18.6 总结	678	20.4 总结	727
第四部分 安 全			
第 19 章 ASP.NET 安全模型 680			
19.1 何为创建安全软件	680	第 21 章 成员资格 728	
19.1.1 理解潜在的威胁	680	21.1 介绍 ASP.NET 的成员资格 API	728
19.1.2 安全编程原则	681	21.2 使用成员资格 API	730
19.1.3 理解 Gatekeeper	682	21.2.1 配置表单验证	731
19.2 理解安全级别	683	21.2.2 创建数据存储	732
19.2.1 验证	683	21.2.3 配置连接字符串和成员资格 提供程序	737
19.2.2 授权	684	21.2.4 创建并验证用户	740
19.2.3 机密性和完整性	685	21.3 使用安全控件	741
19.2.4 互相协作	686	21.3.1 登录控件	742
19.3 IIS 安全	687	21.3.2 登录状态控件	751
19.3.1 IIS 5.x 和 IIS 6.0 中的验证和 授权	688	21.3.3 登录视图控件	752
19.3.2 IIS 7.0 的安全配置	690	21.3.4 恢复密码控件	753
19.3.3 理解 SSL	694	21.3.5 修改密码控件	756
19.4 ASP.NET 安全架构	701	21.3.6 创建用户向导 (CreateUser Wizard) 控件	757
19.4.1 验证	702	21.4 在 IIS 7.0 里配置成员资格	761
19.4.2 授权	703	21.4.1 配置提供程序和用户	761
19.4.3 安全上下文	704	21.4.2 在其他应用程序里使用成员 资格 API	762
19.4.4 成员资格和 Roles API	705	21.5 使用成员资格类	764
19.5 总结	706	21.5.1 从存储中获得用户	765
第 20 章 表单验证 707		21.5.2 更新存储中的用户	767
20.1 表单验证介绍	707	21.5.3 创建和删除用户	767
20.1.1 为什么使用表单验证	708	21.5.4 检验用户	768
20.1.2 为什么不使用表单验证	709	21.5.5 在 Windows 窗体中使用成员 资格	768
20.1.3 为什么不自己实现 cookie 验证	710	21.6 总结	770
20.1.4 表单验证类	711	第 22 章 Windows 验证 771	
20.2 实现表单验证	712	22.1 介绍 Windows 验证	771
20.2.1 配置表单验证	712	22.1.1 为什么使用 Windows 验证	771
20.2.2 禁止匿名用户访问	714	22.1.2 为什么不使用 Windows 验证	772
20.2.3 创建自定义登录页面	715	22.1.3 Windows 验证机制	773
20.2.4 自定义凭证存储	720	22.2 实现 Windows 验证	778
		22.2.1 配置 IIS 5.x 和 IIS 6.0	779

22.2.2	配置 IIS 7.0	780	23.7	总结	827
22.2.3	配置 ASP.NET	781	第 24 章	用户配置	828
22.2.4	禁止匿名用户访问	784	24.1	理解用户配置	828
22.2.5	访问 Windows 用户信息	785	24.1.1	用户配置的性能	828
22.3	身份模拟	790	24.1.2	用户配置如何存储数据	829
22.3.1	在 Windows 2000 中执行身份模拟	791	24.1.3	用户配置和验证	830
22.3.2	在 Windows XP 上面执行身份模拟	791	24.1.4	用户配置与自定义的数据组件	830
22.3.3	在 Windows Server 2003 上运行身份模拟和委托	792	24.2	使用 SqlProfileProvider	831
22.3.4	Windows Vista 上的身份模拟	794	24.2.1	创建用户配置表	831
22.3.5	在 Windows Server 2008 上的身份模拟和代理	794	24.2.2	配置提供程序	833
22.3.6	经过配置的身份模拟	795	24.2.3	定义用户配置属性	834
22.3.7	可编程的身份模拟	797	24.2.4	使用用户配置属性	834
22.4	总结	799	24.2.5	用户配置序列化	836
第 23 章	授权和角色	800	24.2.6	用户配置组	837
23.1	URL 授权	800	24.2.7	用户配置和自定义的数据类型	838
23.2	文件授权	805	24.2.8	用户配置 API	841
23.3	在代码中检查授权	806	24.2.9	匿名用户配置	843
23.3.1	使用 IsInRole() 方法	806	24.3	自定义的用户配置提供程序	845
23.3.2	使用 PrincipalPermission 类	807	24.3.1	自定义的用户配置提供程序类	845
23.4	为基于角色的授权使用角色服务	809	24.3.2	设计 FactoredProfileProvider	847
23.4.1	和角色一起使用 LoginView 控件	814	24.3.3	编码实现 FactoredProfileProvider	848
23.4.2	通过程序访问角色	815	24.3.4	测试 FactoredProfileProvider	851
23.4.3	在 Windows 验证中使用角色服务	817	24.4	总结	854
23.5	在 IIS 5 和 IIS 6 中保护非网页资源	819	第 25 章	加密	855
23.5.1	添加一个文件类型的映射	819	25.1	加密数据: 机密性问题	855
23.5.2	写一个自定义的 HTTP 处理程序	820	25.2	.NET 加密命名空间	856
23.6	IIS 7.0 里的授权和角色	821	25.3	理解.NET 加密类	859
23.6.1	在 IIS 7.0 里使用 ASP.NET 角色授权	823	25.3.1	对称加密算法	860
23.6.2	在 IIS 7.0 中管理 ASP.NET 角色	825	25.3.2	非对称加密	861
			25.3.3	抽象加密类	861
			25.3.4	ICryptoTransform 接口	862
			25.3.5	CryptoStream 类	862
			25.4	加密敏感数据	863
			25.4.1	管理秘密信息	864
			25.4.2	使用对称算法	865