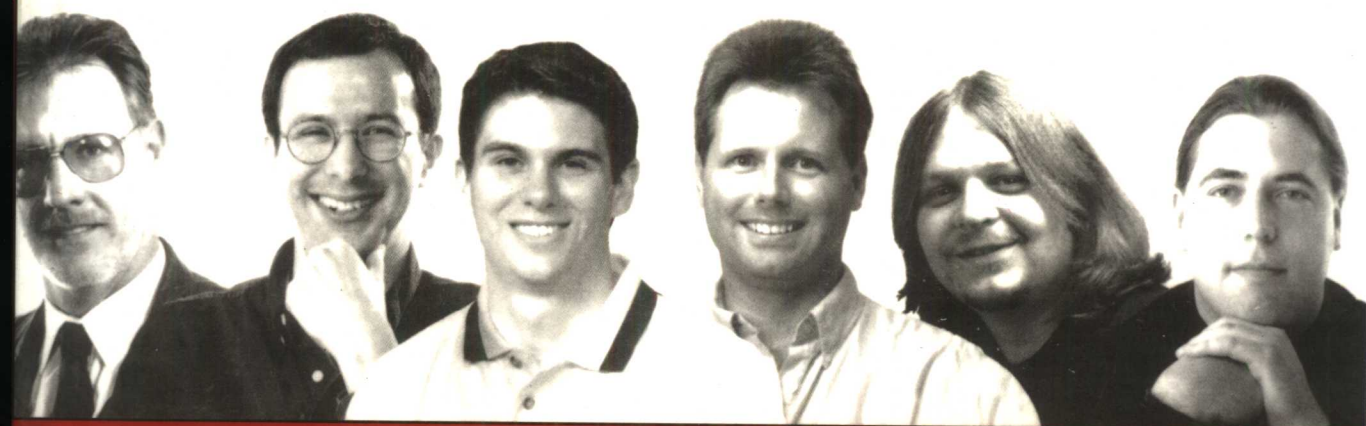




Professional ASP.NET 1.1

ASP.NET 1.1

高级编程

(美) Alex Homer 等著
Dave Sussman 译
李敏波



清华大学出版社

ASP.NET 1.1 高级编程

(美) Alex Homer
Dave Sussman
李敏波

等著
译



清华大学出版社

北 京

Alex Homer, Dave Sussman, et al

Professional ASP.NET 1.1

EISBN: 0-7645-5890-0

Copyright © 2004 by John Wiley & Sons, Inc.

All Rights Reserved. Authorized translation from the English language edition published by John Wiley & Sons, Inc.

本书中文简体字版由 John Wiley & Sons, Inc. 授权清华大学出版社在全球范围内独家出版、发行。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。

北京市版权局著作权合同登记号 图字: 01-2004-3041

版权所有, 翻印必究。举报电话: 010-62782989 13901104297 13801310933

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

本书防伪标签采用清华大学核研院专有核径迹膜防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

ASP.NET 1.1 高级编程/(美)荷马(Homer, A.), (美)苏士曼(Sussman, D.)等著; 李敏波译.

—北京: 清华大学出版社, 2005.1

书名原文: Professional ASP.NET 1.1

ISBN 7-302-09618-X

I. A… II. ①荷…②苏…③李… III. ①主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字(2004)第 097091 号

出版者: 清华大学出版社 地址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮编: 100084

社总机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 王 黎

封面设计: 康 博

版式设计: 康 博

印装者: 北京鑫海金澳胶印有限公司

发行者: 新华书店总店北京发行所

开本: 185×260 印张: 73 字数: 1868 千字

版次: 2005 年 1 月第 1 版 2005 年 1 月第 1 次印刷

书号: ISBN 7-302-09618-X/TP·6672

印数: 1~3500

定 价: 128.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770175-3103 或 (010)62795704

译者序

ASP.NET 是微软整个 Visual Studio.NET 战略中的一个重要组成部分，具有开发效率高、平台无关性、安全性强等方面的优势，广泛用于创建动态 Web 站点和基于 Web 的分布式应用，是广大 Windows 程序员开发 Web 应用的有效工具。本书介绍的是 ASP.NET 的最新版本——ASP.NET 1.1，该版本修正了原有 ASP.NET 1.0 中的一些 Bug，在移动应用开发、代码安全以及数据库 Oracle 和 ODBC 的支持等方面，都做了很多改进。

本书由浅入深地介绍了 ASP.NET 的基本概念和开发机理，内容涵盖了 ASP.NET 与 .NET Framework 原理，如何创建 ASP.NET 页面和使用服务器控件，数据处理与 ADO.NET，ASP.NET 类库，配置、部署问题，作为数据处理有机组成部分的 XML 技术，以及如何使用 Web 服务为应用程序提供异步后台服务，同时还深入分析了软件的迁移问题。

本书比较全面地囊括了在 .NET Framework 下构建 ASP.NET 应用程序的方方面面。在讲解各个主题时，既有系统化的理论说明，也提供了相关的实际样例代码。书中的大量实例演示了实际应用中 ASP.NET 的大部分通用特性，便于读者理解与立即着手练习。

书中所有的源代码都可以在 www.worx.com 下载。该站点为读者提供了学习本书的丰富资源，包括源代码、相关控件、技术文章、多媒体教学工具等，读者也可以注册并登录论坛，与图书的作者、编辑直接进行交流。

由于受时间和译者自身水平的限制，翻译过程中难免出现错误和疏漏，敬请大家多多批评指正，反馈信息请发至 fwkbook@tup.tsinghua.edu.cn。

译者

前言

Microsoft 的 ASP.NET 自从第一次发布为预览技术以来，已经取得了巨大的成功。对于使用 Microsoft 技术开发 Web 站点的开发人员来说，ASP.NET 提供了丰富的编程模型，可以轻松地构建站点。ASP.NET 自发布以来对它的评论非常多，但撇开宣传和新闻这些炒作不谈，.NET 确实为开发人员提供了一个能够开发所有应用程序的坚实基础。

虽然目前 Microsoft 开发的 Active Server Pages(ASP)仍然是一个相对年轻的产品，但它已经成为最主要的 Web 开发工具。它的成功要归功于易用性和灵活性，这为创建动态网站提供了一种简单的方式。尽管这种成功还不完善，但其中出现了许多问题仅仅是因为 ASP 发展太快，超出了它的能力所限。当初设计 ASP 的初衷就是将 COM 的基础结构作为 ASP 的基础，而 COM 本身的功能就有一定的限制。

作为整个 .NET Framework 的一部分，ASP.NET 建立在 Common Language Runtime(公共语言运行库，CLR)的基础上——CLR 是一个功能强大而又灵活的结构，它的设计目标不仅满足开发人员的当前需要，而且还可以满足长远需要。此处需明确的是，ASP.NET 不仅是 ASP 以前的升级版本，还大大超越了对现有技术的升级——因为它开创了 Web 开发的全新时代。而本书将为您打开通往这项新技术的大门。

一种全新的 ASP

那么对开发人员来说，“一种全新的 ASP”意味着什么呢？毕竟，许多产品在发行时都把自己标榜为“有重大的突破”或“革命性的”，但事实上只不过是某些方面进行了升级而已。而 ASP.NET 不是这样，它致力于为 Internet 应用程序的开发提供一种功能丰富而灵活的环境。它不仅提供了许多新的功能，而且从根本上改变了基于 Web 的应用程序的传统设计方式。

之所以发生了如此大的变化，是由于 ASP.NET 的体系结构更加模块化，同时在更大程度上坚持了基于组件的原则。每个页面都是一个可编程访问的完全编译的对象，并且利用了诸如面向对象设计、实时编译和动态高速缓存这样的技术。同时，ASP.NET 所具有的向下兼容性表明现有的页面和应用程序仍然能够按以前的方式进行处理，因此没有进行突然移植的必要。

ASP.NET 的一个主要目标是对应用程序的安装、配置和更新进行重大的改进。因此组件再也不必在 Web 服务器上注册了，同时一个完整的应用程序可以简单地使用文件复制命令、FTP 或指定的应用程序(如 FrontPage Server Extensions)，从一个服务器上移植到另一个服务器上。

本书的内容

本书将介绍 ASP.NET 的基本概念、使用方法及其用途。第1章将简要介绍 ASP.NET 的相关问题，包括基本概念和对本书其他部分的概述。其目的是让您能够尽快掌握一些基础知识，同时运行一些页面范例。

第 2 章将介绍 .NET Framework, 并检验这个支撑起整个 .NET 的体系结构。其中将讨论有关 CLR 的问题, 说明使用它的原因以及使用它所带来的好处。另外还会介绍 ASP.NET 的设计目标, 并说明它们是如何提供一个强大的开发结构的。

第 3 章将详细介绍各种 .NET 语言, 学习面向对象的结构, 并讨论 Visual Basic 和 JScript 的变化, 以及新的 C# 语言。另外, 还将介绍有关这些语言的 CLR 的各种优点, 以及 CLR 是如何克服以前的语言问题的。

第 4 章开始详细介绍 ASP.NET, 考察 ASP.NET 页面是如何构建的。我们将介绍一个简单的 ASP 页面, 学习它是如何转换为 ASP.NET 页面的, 并研究一个更简练的新页面。此外, 还会介绍新 ASP.NET 页面中的代码管理问题, 同时说明 Visual Basic 相对 ASP 而言与新的事件模型更为相似。

第 5、6、7 章将仔细讨论 ASP.NET 服务器控件, 首先介绍这些控件的概念及其工作原理。接下来还要介绍验证控件, 这些控件提供了一种验证用户输入的方式, 然后介绍 Web Form 控件和列表控件, 这些控件都提供了强大的内容管理功能, 最后讨论数据绑定问题, 说明控件是怎样从数据源中自动显示数据的。

第 8 章开始讨论 ASP.NET 中的数据管理问题, 包括 ADO.NET 及其设计目标和体系结构。第 9 章将讨论关系数据和如何从数据库中操作数据的内容。第 10 章将介绍这些数据库中的数据更新问题。而在第 11 章中, 除了继续关注数据问题外, 还要介绍 .NET 中的 XML 使用问题, 以及 XML 对象是如何实现处理 XML 数据的强大功能的。

第 12 章的中心问题则是 Web 应用程序的准确概念, 以及怎样实现对应用程序的管理。要介绍的内容包括: 状态管理、应用程序事件体系结构, 以及应用程序体系结构的扩展。

应用程序编写完成以后, 接下来的工作就是对它们进行部署, 而这正是第 13 章要介绍的问题, 此外还要讨论配置问题。我们将介绍 XML 配置文件, 包括对其选项的详细讨论, 最后学习 ASP.NET 的扩展问题。

第 14 章的内容包括编写安全的 ASP.NET 应用程序, Windows 2000 和 IIS(Internet 信息服务系统)的安全性, 以及 ASP.NET 是如何整合这种安全性的。此外还要介绍声明和编程安全问题, 其中包括像基于 Form 和 Passport 身份验证这样的问题。

第 15 章和第 16 章要解决的是基类库问题, 先详细介绍集合和列表, 然后介绍文件系统对象、流、网络类和正则表达式。基类提供了一组非常丰富的功能, 这些功能的应用范围非常广泛, 使得开发人员付出比 ASP 少得多的编程工作量来建立网站。

在以前的 DNA(内外层网络寻址技术)体系结构中, 将业务组件放置在中间层是通行的做法。而在 .NET 中, 整个体系结构更加简化了。第 17 章将讨论业务对象和事务处理页面的使用问题。我们将看到新体系结构的优点, 以及如何设计应用程序来创建大多数新组件模型。

第 18 章将介绍可扩展性问题并考察服务器控件, 还将讨论如何编写这些控件。我们将介绍一些用于创建这些控件的简便编码技巧, 以及写好这些控件后, 它们是如何与现有的服务器控件并存的。

第 19 章和第 20 章将详细介绍 Web 服务。虽然这个问题并不是专门针对 ASP.NET 的, 但在设计和编写应用程序方面, 它确实是一种主要的手段。由于将现有的功能转换为 Web 服务是非常简单的, 因此利用 Web 服务来提供并使用 B2B(企业到企业)模型可以实现众多的功能。

第 21 章介绍的是用于不同领域的各种网络设备, 如电话、PDA 以及其他诸如此类的设备。目前网站的用途已不再局限于那些带有大显示屏幕的计算机了, 各种较小的设备在未来会得到越来越广泛的应用。本章将介绍 Mobile Internet Toolkit, 看看如何利用它来创建可供小设备访

问的网站。

第 22 章主要包括两个问题：调试和错误处理。虽然有些新功能属于 ASP.NET 所有，但其他新功能却是整个基础架构的一部分。不过不管它们属于谁，最后的受益者都必然是开发人员。这些功能为调试和处理错误工作提供了简单且灵活的方式。

第 23 章覆盖了迁移和互操作性能方面的内容。目前的 ASP 代码非常之多，因此如何将现有的应用程序迁移到新的架构中是一个非常重要的问题。另外，还要讨论与现有的 COM 组件的互操作性问题，这种互操作性可允许中间层逐渐迁移。

最后，第 24 章将进行案例研究。该案例融合了本书介绍的许多技巧和方法。该案例是一个简单的电子商务网站，其中使用了数据访问、服务器控件和类库等。

读者对象

本书面向具有 ASP 或 Visual Basic 开发经验的高级开发人员。而对于那些未掌握基本编程技术和编程语言的用户来说，需要进一步学习才可以使使用本书。

本书旨在对 ASP.NET 的概念作全面介绍——书中介绍了一些背景知识，同时对 ASP.NET 的设计开发思路进行了说明。之后本书对各种功能进行深入探讨，而这些功能要求开发人员之前必须使用过。此外还对如何快速明了地利用这些新功能进行了介绍。

如果您以前使用过 ASP，就不会对本书涉及的各种概念太陌生，那么在使用本书时，就无需借助其他参考资料(随 ASP.NET 提供的 SDK 和帮助文件除外)。同时，您应该对组件、Visual Basic 和 VBScript 语言的常规使用原则有所了解。本书中的某些范例是用其他语言编写的，如 CLR 所支持的 JScript 和 C#(这是一种新的语言)，但即使您对这些语言比较陌生，也不妨碍您使用本书。

使用本书的要求

要运行本书中的各种范例，需要事先满足以下条件：

- Windows 2000、Windows Server 2003 或 Windows XP
- ASP.NET。既可以是新发布的版本(包括在 .NET SDK 中)，也可以是 Visual Studio .NET。

本书中所有范例的完整源代码均可以从下面的网站中下载：<http://www.wrox.com/>。其中包含了 Visual Basic .NET 和 C# 版本(参见下面的“源代码”一节)。

源代码

对于本书中的所有范例来说，您可能更愿意亲自将所有代码都输入到计算机中。许多读者更喜欢采用这种方式，因为这可以更加有效地掌握那些正在使用的技术。本书中的所有源代码都放在 <http://www.wrox.com/> 网站中。在您登录到 Wrox.com 站点 <http://www.wrox.com/> 时，只需使用 Search 工具框或使用书名列表就可以找到本书。接着单击本书信息页面上的 Download Code 链接，就可以获得所有的源代码。

说明：

因为很多书的标题相类似，所以最简单的方法是通过 ISBN 进行搜索。本书的英文 ISBN 号是 0-7645-5890-0。

从该站点上下载的文件是已经使用了 WinZip 进行压缩的文件。在把附件保存到硬盘的一个文件夹中时, 需要使用解压缩文件(如 WinZip 或 PKUnzip)对该文件解压缩。或者, 您可以访问 Wrox 的下载页面: [Http://www.wrox.com/dynamic/books/download.aspx](http://www.wrox.com/dynamic/books/download.aspx), 来查看本书或其他所有由 Wrox 出版的书籍中的代码。

勘误表

尽管我们已经尽了各种努力来保证文章或代码不出现错误, 但是错误总是难免的, 如果您在本书中找到了错误, 请告诉我们, 我们将非常感激。通过勘误表, 可以让其他读者避免受挫, 当然, 这还有助于提供更高质量的信息。

要在网站上找到勘误表, 可以登录 <http://www.wrox.com>, 通过 Search 工具框或书名列表查找本书, 然后在本书的信息页面上, 单击 Book Errata 链接。在这个页面上, 可以查看已经提交并通过编辑检查的所有勘误。此外, 在 www.wrox.com/misc-pages/booklist.shtml 页面上还提供了完整的书籍列表以及相关的勘误表链接。

如果在 Book Errata 页面未找到相关错误, 则可以访问 www.wrox.com/contact/techsupport.shtml 并填写其中的表单, 将您发现的错误发送给我们。我们将核查该信息, 并将该消息发布到该书的勘误表页面中, 且在后续的版本中修正该问题。

p2p.wrox.com

P2P 论坛是为作者和同行之间的讨论而设立的。我们在邮件列表、论坛和新闻组中提供“由程序员为程序员的支持”(programmer to programmer support), 还包括一对一的电子邮件支持系统。您可以相信, 您的问题不仅仅是由支持专家解答, 而且还要提供给我们邮件列表中的许多 Wrox 作者和其他业界专家。在 p2p.wrox.com 上, 您可以从许多不同的列表中获得帮助, 不仅对阅读本书有所帮助, 还可以在开发应用程序时获得帮助。

要加入该论坛, 可以遵循下面的步骤:

- (1) 进入 p2p.wrox.com 并单击 Register 链接。
- (2) 阅读使用条款并单击 Agree 按钮。
- (3) 填写表单中要求填写的信息, 以及其他其他可选信息, 然后单击 Submit 按钮, 即可加入论坛。
- (4) 您将接受到一封确认电子邮件, 其中列出如何验证您的账户和加入论坛的完整步骤。

说明:

您不加入 P2P 论坛也即可浏览其中信息, 但要发布信新消息, 则必须注册。

在加入该论坛后, 您即可发布新消息并回复其他人的文章。可在任意时刻通过 Web 阅读其中信息。如果需要, 也可通过电子邮件订阅论坛中特定主题的新信息, 为此, 请在论坛列表选中特定论坛名称对应的 Subscribe to this Forum 图标。

要更详细地了解如何使用 Wrox P2P 论坛, 请阅读 P2P FAQ, 其中列出了论坛软件如何工作等众多常见的问题, 单击 P2P 论坛中任意页面中的 FAQ 链接即可阅读相关内容。

目 录

第 1 章 ASP.NET 概述	1
1.1 是升级还是创新	1
1.2 开始使用 ASP.NET	2
1.3 安装.NET	2
1.3.1 配置范例	3
1.3.2 运行范例	3
1.4 Visual Studio .NET 2003	5
1.5 在 Visual Studio .NET 2003 中创建 ASP.NET 应用程序	7
1.6 ASP.NET 与 ASP 的区别	8
1.6.1 使用新技术的原因	8
1.6.2 ASP.NET 的优点	9
1.7 编写 ASP.NET 页面	18
1.7.1 创建 Web 站点	18
1.7.2 页面范例	18
1.8 本章小结	21
第 2 章 认识 .NET Framework	22
2.1 .NET 简介	23
2.1.1 .NET 的组成	23
2.1.2 .NET 战略	23
2.1.3 Windows DNA 存在的问题	25
2.1.4 需要.NET 的原因	27
2.2 公共语言运行库(CLR)	31
2.2.1 CLR 服务	31
2.2.2 通用功能	32
2.2.3 使用对象	32
2.2.4 命名空间	33
2.2.5 通用类型系统	34
2.2.6 程序集——版本化和安全性代码	40
2.2.7 CLR 和 COM	44
2.2.8 中间语言	44
2.2.9 应用程序域	45

2.3	.NET Framework 详解	46
2.3.1	应用程序开发技术	47
2.3.2	类库	49
2.3.3	基类库	49
2.4	ASP.NET 的设计目标	50
2.4.1	消除对脚本引擎的依赖性	51
2.4.2	性能	51
2.4.3	在 ASP 基础上的升级	52
2.4.4	便于部署	52
2.4.5	强大的工具支持	53
2.4.6	更简单、更灵活的配置	54
2.4.7	分解的“开放”设计	55
2.4.8	与语言无关	57
2.4.9	代码更少、更简练、更易于维护	58
2.4.10	功能强大的身份确认模型	58
2.4.11	正确认识错误的存在	59
2.5	本章小结	59
第 3 章	.NET 语言	60
3.1	所提供的语言	61
3.1.1	VBScript 何去何从	61
3.1.2	Visual Studio 还是 Notepad	61
3.2	Visual Basic .NET	62
3.2.1	面向对象功能	62
3.2.2	语言的变化	71
3.3	C#	75
3.3.1	类	76
3.3.2	方法	77
3.3.3	属性	78
3.3.4	构造函数	79
3.3.5	析构函数	79
3.3.6	继承性	80
3.3.7	接口	81
3.3.8	引用	82
3.3.9	异常处理	82
3.3.10	XML 文档	84
3.3.11	非安全代码	87
3.3.12	运算符重载	88

3.4	JScript .NET	88
3.5	C++	89
3.6	Visual J# .NET	89
3.7	所支持的其他语言	89
3.8	.NET 语言编译器	90
3.9	CLR 的优点	93
3.9.1	通用 API	93
3.9.2	通用类型	94
3.9.3	跨语言继承性	95
3.9.4	跨语言调试和记录	97
3.10	性能问题	97
3.10.1	语言	97
3.10.2	后期绑定代码	97
3.11	常用范例	98
3.11.1	变量声明	98
3.11.2	函数和过程	98
3.11.3	语法的区别	99
3.11.4	循环	100
3.11.5	类型转换	102
3.12	本章小结	103
第 4 章	编写 ASP.NET 页面	104
4.1	编码问题	104
4.1.1	以前的编码方式	104
4.1.2	ASP.NET 页面中的编码	106
4.1.3	回送识别	110
4.2	Page 类	112
4.2.1	HttpRequest 对象	115
4.2.2	HttpResponse 对象	117
4.2.3	页面处理步骤	120
4.2.4	页面状态	129
4.2.5	页面指令	130
4.3	后台编码	135
4.3.1	开发工具中的后台编码	135
4.3.2	使用后台编码	136
4.3.3	在 ASP.NET 页面中继承后台编码类文件	138
4.4	页面高速缓存	139
4.4.1	输出高速缓存范例	139

4.4.2 通过浏览器实现高速缓存	140
4.5 智能导航	141
4.6 用户控件	141
4.6.1 建立用户控件的方法	142
4.6.2 用户控件属性	146
4.6.3 用户控件事件	147
4.6.4 用户控件的后台编码	148
4.6.5 用户控件的局部页面高速缓存	149
4.7 本章小结	151
第 5 章 服务器控件与验证控件	152
5.1 服务器控件的概念	152
5.1.1 再论 ASP.NET 页面模型	152
5.1.2 练习使用服务器控件	155
5.1.3 关于范例应用程序	156
5.1.4 使用服务器控件的场合	157
5.1.5 ASP.NET 中的可用控件	158
5.2 HTML 服务器控件	159
5.2.1 HtmlControl 基类	159
5.2.2 HTMLGenericControl 类	160
5.2.3 HTML 控件的专有类	161
5.2.4 HTML 服务器控件的用法	163
5.2.5 HTML 服务器控件范例	164
5.3 ASP.NET 输入验证控件	181
5.3.1 输入验证控件的作用	181
5.3.2 BaseValidator 类	182
5.3.3 验证控件的特有成员	182
5.3.4 验证控件的用法	183
5.4 本章小结	190
第 6 章 ASP.NET Web Form 控件	191
6.1 基本的 ASP.NET Web Form 控件	191
6.1.1 为什么要有另外一组控件	192
6.1.2 基类 WebControl	192
6.1.3 特殊的 Web Form 控件类	194
6.1.4 使用 Web Form 控件	195
6.1.5 ASP Web Form 控件范例	201
6.2 ASP.NET 列表控件	216
6.2.1 ListControl 基类	216

6.2.2 特殊的列表控件类	217
6.2.3 使用 ASP 列表控件	218
6.3 ASP.NET 中的其他多功能控件	223
6.4 本章小结	228
第 7 章 列表控件和数据绑定	229
7.1 获取范例文件	229
7.2 数据绑定的相关概念	230
7.2.1 在服务器上完成所有操作	230
7.2.2 显示数据——ASP 和 ASP.NET	231
7.2.3 数据绑定的语法	232
7.2.4 单值数据绑定	232
7.2.5 重复值数据绑定	236
7.2.6 重复值数据绑定的简单范例	243
7.3 添加样式和模板	256
7.3.1 使用样式属性	256
7.3.2 给 DataGrid 控件添加样式	257
7.3.3 对数据绑定控件使用模板	258
7.3.4 处理数据绑定事件	273
7.3.5 对 DataGrid 中的行进行排序和筛选	275
7.3.6 DataGrid 中的自动分页	280
7.4 使用数据绑定控件编辑数据	284
7.4.1 利用 DataGrid 控件编辑数据	284
7.4.2 利用 DataList 控件选择和编辑数据	289
7.5 本章小结	295
第 8 章 .NET 数据管理引论	296
8.1 数据存储和数据访问	296
8.1.1 转向分布式环境	297
8.1.2 XML——将来的数据格式	297
8.1.3 是否是另一项数据访问技术	298
8.1.4 .NET 意味着断开数据的连接	299
8.1.5 .NET 意味着 XML 数据	301
8.1.6 .NET 意味着托管代码	301
8.1.7 .NET 意味着新的编程模型	301
8.2 .NET 中的数据管理	302
8.2.1 数据管理的系统命名空间	302
8.2.2 导入所需的命名空间	303
8.3 基本的 ADO.NET 类	305

8.3.1	ADO 和 ADO.NET 的技术比较	306
8.3.2	Connection 类	307
8.3.3	Command 类	308
8.3.4	DataAdapter 类	308
8.3.5	DataSet 类	310
8.3.6	DataTable 类	311
8.3.7	DataRow 类	312
8.3.8	DataReader 类	313
8.3.9	DataReader 和 DataSet 的选取	315
8.3.10	.NET 中的关系数据提供程序	315
8.4	.NET 中常见的数据访问任务	316
8.4.1	在自己的系统上创建范例	317
8.4.2	ASP.NET 与 IIS 6.0 和 Windows Server 2003	318
8.4.3	使用 DataReader 对象	319
8.4.4	一个简单的 DataSet 范例	321
8.4.5	含有多个表的 DataSet 范例	323
8.4.6	返回 DataSet 对象的用户控件	327
8.5	.NET 中的 XML 引论	328
8.5.1	基本的 XML 对象	328
8.5.2	XPathNavigator 类	331
8.5.3	XmlTextWriter 类	333
8.5.4	XmlReader 类	333
8.5.5	XmlValidatingReader 类	334
8.5.6	XslTransform 类	335
8.6	.NET 中常见的 XML 任务	335
8.6.1	通过 DOM 的 XML 文档访问操作	336
8.6.2	使用 XPathNavigator 对象的 XML 文档访问操作	338
8.6.3	搜索 XML 文档	342
8.6.4	XML TextWriter 对象范例	344
8.6.5	XML TextReader 对象范例	347
8.6.6	XSL Transform 对象范例	349
8.7	选择一种数据存储的方法	350
8.7.1	关键在于访问和处理	351
8.7.2	查询的新方法	351
8.7.3	传输协议是大势所趋	352
8.7.4	.NET 是一套伟大的解决方案	352
8.8	本章小结	352

第 9 章 处理关系数据	353
9.1 获得范例文件	353
9.2 访问复杂数据	354
9.2.1 利用 DataReader 访问复杂数据	354
9.2.2 通过 DataSet 访问复杂数据	356
9.2.3 访问和显示嵌套的数据	357
9.2.4 使用数据库的存储过程	360
9.3 处理 DataTable 对象	368
9.3.1 创建和填充新 DataTable 的代码	368
9.3.2 在表中添加数据行	369
9.3.3 为 DataSet 添加表	370
9.3.4 管理约束和默认值	370
9.3.5 指定主键和外键	373
9.3.6 添加、修改、移去和删除行	377
9.3.7 处理 DataTable 事件	384
9.3.8 使用表和列映射	388
9.4 数据的排序和筛选	391
9.4.1 DataTable 对象中的排序和筛选	391
9.4.2 DataView 对象中的排序和筛选	396
9.5 本章小结	398
第 10 章 关系数据源的更新	399
10.1 获得范例文件	399
10.2 利用 Command 对象更新数据	400
10.2.1 使用 Command 对象和 SQL 语句	400
10.2.2 使用存储过程和 Command 对象	403
10.3 使用事务处理更新数据源	406
10.3.1 数据库事务	407
10.3.2 基于连接的事务处理	410
10.4 更新 DataSet 对象中的数据	414
10.4.1 深入到 DataAdapter 对象内部	414
10.4.2 使用 DataAdapter.Update 方法	419
10.4.3 使用存储过程更新 DataSet	424
10.4.4 DataAdapter 中的更新事件	431
10.5 对并发数据更新的管理	435
10.5.1 避免出现并发错误	436
10.5.2 一个并发错误范例	436
10.5.3 仅更新已改变的列	438

10.5.4	利用 RowUpdated 事件捕获错误	443
10.5.5	更新完成后查找错误	452
10.6	本章小结	456
第 11 章	.NET 中的 XML 数据管理	457
11.1	获取范例文件	457
11.2	XML 和 DataSet 对象	458
11.2.1	DataSet 基于 XML 的方法	458
11.2.2	将数据从 DataSet 写入 XML 文件	459
11.2.3	将数据从 XML 文件读入 DataSet	462
11.2.4	以字符串的形式从 DataSet 中得到 XML	465
11.2.5	DataSet 中的嵌套 XML 和关系数据	466
11.3	处理 XmlDataDocument 对象	470
11.4	直接从 SQL Server 中读取 XML 数据	476
11.5	验证 XML 文档的有效性	478
11.6	创建和编辑 XML 文档	486
11.7	使用 XSL 和 XSLT 转换	492
11.7.1	一个 XSL 转换的范例	492
11.7.2	用 XmlResolver 加载样式表和模式	496
11.8	本章小结	498
第 12 章	Web 应用程序和 global.asax	499
12.1	IIS Web 根目录以及应用程序	499
12.1.1	创建 IIS 虚拟目录	500
12.1.2	把文件夹标记为应用程序	501
12.1.3	Windows Server 2003	503
12.2	ASP.NET Web 应用程序	506
12.2.1	注册组件	506
12.2.2	应用程序代码: global.asax	510
12.3	应用程序状态的管理	514
12.3.1	Session——管理用户状态	515
12.3.2	Application——管理应用程序状态	517
12.3.3	Cache——管理瞬时状态	519
12.4	应用程序事件	528
12.4.1	事件的语法和原型	529
12.4.2	支持的事件	529
12.4.3	事件范例	532
12.5	高级主题	537
12.5.1	静态变量	538

12.5.2	为 global.asax 使用自己的基类	539
12.5.3	将文件扩展名映射到 ASP.NET 中	540
12.5.4	异步应用程序事件	542
12.6	本章小结	547
第 13 章	ASP.NET 的配置	549
13.1	配置综述	549
13.1.1	ASP 的配置	550
13.1.2	ASP.NET 的配置	551
13.1.3	配置文件格式	554
13.2	常用的配置设置	558
13.2.1	通用配置设置	558
13.2.2	页面配置	561
13.3	跟踪	571
13.3.1	传统的 ASP 跟踪	571
13.3.2	ASP.NET 跟踪	572
13.3.3	查看跟踪输出	572
13.3.4	跟踪配置设置	574
13.4	定制错误	575
13.4.1	错误模式	576
13.4.2	总是显示 ASP.NET 错误页面	577
13.4.3	定制错误页面	577
13.4.4	配置 IIS 和 ASP.NET 来支持相同的错误页面	578
13.5	身份验证和授权	579
13.5.1	身份验证	579
13.5.2	护照验证	581
13.5.3	授权	582
13.6	Web 服务	582
13.6.1	国际化和编码	583
13.6.2	编译选项	585
13.6.3	执行身份的控制	590
13.6.4	用 HTTP 处理程序来扩展 ASP.NET	592
13.6.5	用 HTTP 模块扩展 ASP.NET	596
13.6.6	配置 ASP 工作进程	597
13.6.7	机器密钥	606
13.7	高级问题	607
13.7.1	指定位置	607
13.7.2	设置路径	608