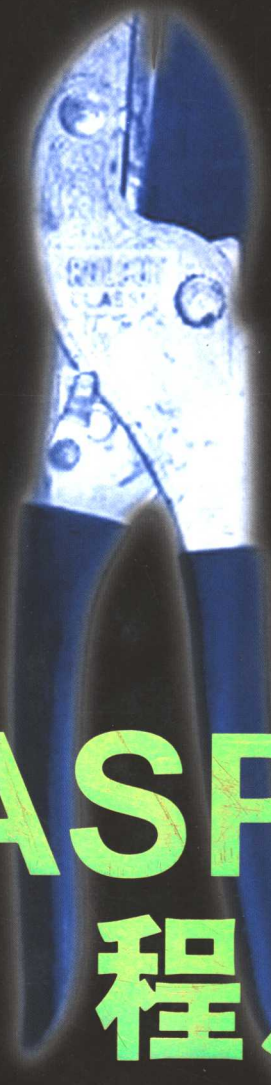


微软.NET程序设计系列

Microsoft Press



精通ASP.NET 程序设计

Programming Microsoft ASP.NET

(意) **Dino Esposito** 著
罗 兵 顾雁宏 詹文军 译



清华大学出版社

TP393.09
87

微软.NET 程序设计系列

精通 ASP.NET 程序设计

(意)Dino Esposito 著

罗 兵 顾雁宏 詹文军 译

清华大学出版社

北 京

内 容 简 介

本书的作者是 ASP.NET 和 ADO.NET 方面的专家, 书中内容覆盖了 ASP.NET 编程的所有方面, 不但从专业角度介绍了其内部体系结构, 包括 HTTP 运行库、标准控件、配置、安全性、固有控件和自定义控件等内容, 而且从实践出发给出了许多实例, 读者可以修改这些实例作为自己的应用程序通过本书的学习, 可以帮助读者全面而深入地掌握如何用 ASP.NET 创建高效率、高性能的 Web 应用程序。

本书不仅适用于希望全面了解 ASP.NET 体系结构以及编程技巧的读者, 而且对于从事网站或网页开发的网络工程师或工程技术人员, 本书也是一本即权威又非常具有参考价值的指导书。

Programming Microsoft ASP.NET, by Dino Esposito(0-7356-1903-4)

Copyright © 2003 by Dino Esposito.

Original English Language Edition Copyright © 2003 by Dino Esposito.

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A.

本书中文简体版由 Microsoft Press 授权清华大学出版社出版发行, 未经出版者书面许可, 不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号 图字: 01-2003-3105

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 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。

本书防伪标签采用特殊防伪技术, 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

精通 ASP.NET 程序设计/(意)埃斯波西托(Esposito,D.)著; 罗兵, 顾雁宏, 詹文军译. —北京: 清华大学出版社, 2006.7

(微软.NET 程序设计系列)

书名原文: Programming Microsoft ASP.NET

ISBN 7-302-13185-6

I. 精… II. ①埃…②罗…③顾…④詹… III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字(2006)第 062086 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

文稿编辑: 宣 颖

封面设计: 陈刘源

排版人员: 房书萍

印 刷 者: 清华大学印刷厂

装 订 者: 三河市李旗庄少明装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 53 字数: 1505 千字

版 次: 2006 年 7 月第 1 版 2006 年 7 月第 1 次印刷

书 号: ISBN 7-302-13185-6/TP·8341

印 数: 1~4000

定 价: 85.00 元

前言

几年前，在意大利的一个 11 月份雨天的早晨，我正在接受一家顶级咨询公司的高层经理对我的面试。我做过开发人员、顾问、解决方案构造人员，但是 Internet 当时还不算什么，我只是靠 Win 32 和 C++ 编程技能谋生而已。尽管我已经听说了广泛用于浏览 Web 页面的 HTTP 协议，但是我还没有打算编写这样的应用程序。在那时，要将 Web 和应用程序联系在一起，我一定会犹豫的。对我来讲，应用程序是很严肃的一件事情，不可能仅用一些简单的脚本代码和标记代码就能够完成的。

为了证明我的薪水要求是合理的，我简略地提了提 Internet 将会对业界日益重要的观点。面试我的经理严肃地看了我一眼，然后说了一句改变了我整个职业生涯的话，他说：“Internet 是一个实实在在的突破；它能让你获得使用传统方法不能取得的结果。”你可以想像接下来所发生的事情。我被录用了，不久就开始了我的第一个 Web 项目。几个星期后，我的项目经理神情忧虑地走过来问我：“你做过多少 C++ 编程？”我答道：“C++？我一出生就已经被 C++ 实例化了。”这是一个什么问题？Internet 为整个业界打开了一个充满商业机会的世界。不久，Web 专家就成为了宝贵的资源，但是，他们仍然很少能解决很棘手的系统或者网络问题。尽管在一定程度上 Internet 正越来越受到欢迎，但是其后台编程模型仍然是独立的应用于操作系统的特殊 API 而且还有不同的 API 和方案。

微软公司最初尝试使用 COM 来解决这个问题，而 ASP 是在 Web 上使用 COM 的绝好工具。但是一直到微软的 .NET Framework 和 ASP.NET 才最终将这个问题圆满解决。使用 ASP.NET，你就拥有了一个与操作系统的编程框架完美集成的多功能 Web 开发平台，而且该编程框架——.NET Framework——不依赖于任何语言。我在几年前离开了上面提到的那家公司，但是我相信如果当时就有 ASP.NET 的话，我在那里的生活会更容易，效率更高。

因此，如果你希望使用微软的技术来编写 Web 应用程序，那么，你的选择非常正确。本书的最终目标是将你带到 ASP.NET 技术的更高层次，帮你缩短宝贵的开发时间。无论你的起点如何，本书都将使你成为一个更精明的程序员、更高效的开发人员。柏拉图曾说过，一个人只要有进步，无论进步多慢，都不应当放弃。

本书从多个层次、多个视角进行剖析。你无需熟悉 ASP 就可以使用本书，但是如果你曾是一个 ASP 程序员，你将更完整地抓住这个新的 Web 开发平台的精髓。同样，你也无需了解 ASP.NET 1.0，但是如果你是具有 ASP.NET 开发经验的开发人员，你将更喜欢 ASP.NET 1.1 和微软 Internet Information Services 6.0 (IIS)。最后，如果你是一个有经验的程序员，本书的部分章节对 ASP.NET 组件的内部细节进行了充分的讨论，这些内容可以作为公开的 MSDN 文档的理想的补充资料。

1. 本书的内容

本书按照一般的学习过程分为六个部分。在第一部分“构建 ASP.NET 页面”中，你将构建一个页面，了解 ASP.NET 模型和控件的基本知识。在第二部分“在 ASP.NET 站点中添加数据”中，你将添加数据支

持,并接触到 ADO.NET 对象模型。在第三部分“ASP.NET 控件”中,将学习 ASP.NET 体系结构中处于中心位置的组件,即服务器控件,这将涉及到许多控件,包括移动控件。此时,你将学习到足够的知识,进行实际的项目开发,并且具备相当的工作效率。如果你想学习更多的知识,你就需要深入了解 ASP.NET 应用程序的其他方面,这就是第四部分“ASP.NET 应用程序的基础”的内容,其中包括配置、安全性和状态管理,以及文件 I/O 和 Web 服务(即在 ASP.NET 项目中常用的一个系统组件和一个应用程序模型)。一旦你了解服务器控件的价值,你就不使用自定义控件了吗?第五部分“自定义 ASP.NET 控件”将专门讨论自定义控件。第六部分“附录”包括图像处理、HTTP 处理程序和模块,以及在 IIS 以外的地方寄宿 ASP.NET 等内容。

在本书中,我将回答下列常见问题:

- ASP.NET 如何编译页面?
- 在哪里创建和存储动态程序集?
- 什么是 HTTP 管道?
- 在页面生命周期中事件发生的确切顺序是什么?
- HTML 控件和 Web 控件之间有何不同?
- 为什么总是使用单表单?
- 如何能够更有效地使用 DataSet 对象?
- 数据集的最好分页方式是什么?
- 能够使用 DataGrid 控件实现那个效果吗?(是的,通常都可以实现。)
- DataList 控件和 Repeater 控件比 DataGrid 控件更灵活吗?
- 怎样利用 ASP.NET 技巧为移动设备编写应用程序?
- 什么是 Web 用户控件?其中包括什么?
- 各个 Session 模式之间有什么区别?
- 如何才能有效利用 Cache 对象?
- ViewState 是被加密还是编码?使用起来真的安全吗?
- 验证用户最灵活的方式是什么?
- 什么是 ASP.NET 用户?
- 能够在 Web 服务和 ASP.NET 应用程序之间共享会话状态吗?
- 如何在设计时配置自定义控件使其具有合理的外观?
- 能够动态地生成图像吗?
- 如何使用 HTTP 处理程序和 HTTP 模块?
- 有没有不使用 IIS 运行 ASP.NET 的方法?

你在本书中将看到一些很有趣的示例程序:

- 使用常规表达式将 Web 站点名称和电子邮件地址转换为超链接的 LinkLabel Web 控件
- 将分页条呈现为 Tab Strip 的 DataGrid 控件
- 允许显式地设置目的 URL 的 Form 控件
- 在 DataGrid 控件中的数据捆绑工具提示和小结
- 一个基于 HTTP 的生成条形图的控件
- 响应输出过滤程序的演示

- 多行 DataGrid 控件

2. 系统要求

本书需要使用下列软件：

- 安装有 IIS 的微软 Windows 系统(以下之一)
 - ◆ Windows 2000
 - ◆ Windows XP Professional
 - ◆ Windows Server 2003
- 微软 Visual Studio .NET 2003(以下之一)
 - ◆ Visual C# .NET Standard 2003
 - ◆ Visual Basic .NET Standard 2003
 - ◆ Visual Studio .NET Professional 2003
 - ◆ Visual Studio .NET Enterprise Developer 2003
 - ◆ Visual Studio .NET Enterprise Architect 2003
- 微软 SQL Server(以下之一)
 - ◆ SQL Server 2000
 - ◆ SQL Server 2000 Desktop Engine (可下载)

本书是为 .NET Framework 1.1 和 Visual Studio .NET 2003 编写的。书中大多数代码可以在 .NET Framework 1.0 上运行，但是所有的代码示例程序都能够使用 .NET Framework 1.1 和 Visual Studio .NET 2003 进行测试。一些代码示例程序要求安装 SQL Server 2000 或者 SQL Server 2000 Desktop Engine (MSDE 2000)。尽管在任何严肃的开发环境中都不应该使用空密码，但是本书中使用 SQL Server 的代码示例程序都假定了一个空的 sa 用户密码。如果你在安装 SQL Server 时没有使用空的 sa 密码，你需要将自己设置的密码添加到连接字符串中。例如，如果你的密码为“Hello”，你应当将连接字符串修改为：

```
string nwind = "SERVER=localhost;UID=sa;PWD=Hello;DATABASE=northwind;"
```

3. 代码示例程序

本书没有附带 CD，但是在书中讨论的所有代码示例程序都可以在下面的 Web 地址上获得。请单击页面右边的 More Information 框中的 Companion Content 链接。

- <http://www.microsoft.com/mspress/books/6667.asp>

书中使用的语言为 C#，但是所有的代码都使用 C# 和 Visual Basic .NET 两种语言编写。

4. 如何与基地联系

请将有关本书的任何问题直接发给作者。你可以通过下面的电子邮件联系 Dino Esposito：

- dinoe@wintellect.com
- desposito@vb2themax.com

关于其他信息和资源，请访问下列网站：*Wintellect* <http://www.wintellect.com> 和 *VB2-The-Max* <http://www.vb2themax.com>。

5. 微软出版社的支持

我们尽了最大的努力保证本书和相关内容的准确性。微软出版社通过互联网提供了本书的勘误：

<http://www.microsoft.com/mspress/support/>

如果你希望对本书或者其相关内容发表评论、提出问题或者想法，或者有在 Knowledge Base 中没有找到答案的问题，请将它们通过以下电子邮件发送到微软出版社：mspinput@microsoft.com

或者通过邮件发送到：

Microsoft Press

Attn: Programming Microsoft ASP.NET Editor

One Microsoft Way

Redmond, WA 98052-6399

注意，产品支持并不通过上述地址提供。

目 录

第一部分 构建 ASP.NET 页面

第 1 章 Web Forms 模型 3	
1.1 ASP.NET..... 3	
1.1.1 Web Forms 时代的编程..... 4	
1.1.2 基于 HTTP 的 Web Forms 编程..... 5	
1.1.3 HTTP 协议..... 6	
1.2 ASP.NET 页面的结构..... 8	
1.2.1 ASP.NET 页面示例..... 8	
1.2.2 页面的处理指令..... 10	
1.2.3 <script>部分..... 17	
1.2.4 布局部分..... 20	
1.3 ASP.NET 组件模型..... 20	
1.3.1 组件交互模型..... 21	
1.3.2 ASP.NET 服务器控件..... 24	
1.4 从 ASP 移植到 ASP.NET..... 25	
1.4.1 同步执行..... 25	
1.4.2 将 ASP 页面改写为 ASP.NET 页面..... 25	
1.4.3 推荐的移植方案..... 26	
1.5 设置 ASP.NET 应用程序..... 28	
1.5.1 创建 Visual Studio 项目..... 28	
1.5.2 使用 Visual Studio .NET 开发多语言项目..... 36	
1.5.3 Web Matrix 编辑器..... 37	
本章小结..... 40	
资源..... 41	
第 2 章 Web Forms 内幕 42	
2.1 执行 ASP.NET 页面..... 42	
2.1.1 IIS 资源映射..... 43	
2.1.2 ASP.NET 工作进程..... 44	
2.1.3 ASP.NET 的 HTTP 管道..... 49	
2.1.4 ASP.NET 页面工厂对象..... 52	
2.1.5 处理页面请求..... 60	
2.2 事件模型..... 61	
2.2.1 单表单模型..... 62	
2.2.2 回传事件..... 64	
2.3 Page 类..... 66	
2.3.1 Page 对象的属性..... 67	
2.3.2 Page 对象的方法..... 69	
2.3.3 Page 对象的事件..... 72	
2.4 代码分离技术..... 72	
2.4.1 代码与布局分离..... 72	
2.4.2 页面继承..... 75	
2.4.3 ASP.NET 中的主页面..... 79	
本章小结..... 79	
资源..... 80	
第 3 章 ASP.NET 核心服务器控件 81	
3.1 ASP.NET 服务器控件..... 82	
3.1.1 Control 类的属性..... 82	
3.1.2 Control 类的方法..... 83	
3.1.3 Control 类的事件..... 83	
3.2 HTML 控件..... 84	
3.2.1 HTML 控件概况..... 84	
3.2.2 HTML 容器控件..... 87	
3.2.3 HTML 输入控件..... 90	
3.2.4 HtmlImage 控件..... 94	
3.3 Web 控件..... 95	
3.3.1 Web 控件概述..... 95	
3.3.2 核心 Web 控件..... 97	
3.3.3 杂项 Web 控件..... 99	
3.4 验证控件..... 102	
3.4.1 验证控件的类型..... 102	

3.4.2 建立数据输入页面	108	4.2 使用脚本代码	137
3.4.3 客户端验证	113	4.2.1 支持多浏览器	138
3.5 数据捆绑控件	117	4.2.2 添加客户端脚本	140
3.5.1 可用于数据绑定的数据源	117	4.3 页面本地化	144
3.5.2 简单数据绑定	119	4.3.1 文化和地域	144
3.5.3 复杂数据绑定	120	4.3.2 ASP.NET 资源管理器	145
本章小结	124	4.4 页面错误与跟踪	147
资源	125	4.4.1 定义自定义错误页面	147
第 4 章 使用页面	126	4.4.2 在 .NET 中跟踪执行 过程	150
4.1 使用表单编程	126	4.4.3 ASP.NET 跟踪	152
4.1.1 HtmlForm 类	127	4.4.4 跟踪查看器	154
4.1.2 多表单	129	本章小结	156
4.1.3 在页面之间传递变量	131	资源	157
4.1.4 弹出表单	135		

第二部分 在 ASP.NET 站点中添加数据

第 5 章 ADO.NET 对象模型	161	5.5.2 DataTable 对象	200
5.1 .NET 数据访问层	162	5.5.3 数据关系	204
5.1.1 .NET 托管提供程序 的体系结构	163	5.5.4 DataView 对象	206
5.1.2 通过 ADO.NET 访问 的数据源	166	本章小结	209
5.2 连接到数据源	169	资源	210
5.2.1 SqlConnection 类	169	第 6 章 创建可绑定数据网格	211
5.2.2 连接池	172	6.1 DataGrid 控件	211
5.2.3 一般数据库编程	174	6.1.1 DataGrid 控件的属性	213
5.2.4 其他连接对象	175	6.1.2 DataGrid 控件的事件	216
5.3 执行命令	176	6.1.3 DataGridItem 对象	217
5.3.1 SqlCommand 类	176	6.2 将数据绑定到网格	220
5.3.2 其他命令对象	179	6.2.1 数据捆绑列	221
5.3.3 ADO.NET 数据读取器	179	6.2.2 超链接列	223
5.3.4 数据读取器的特殊功能	185	6.2.3 命令列	226
5.4 ADO.NET 数据适配器	186	6.3 扩展 DataGrid	231
5.4.1 SqlDataAdapter 类	186	6.3.1 对网格的内容进行分页	231
5.4.2 表映射机制	189	6.3.2 对数据进行排序	235
5.4.3 批量更新的工作方式	192	6.3.3 模板列	239
5.5 ADO.NET 容器对象	194	6.4 DataGrid 就地编辑	244
5.5.1 DataSet 对象	195	6.4.1 配置内置就地编辑引擎	244
		6.4.2 保存修改	249
		本章小结	251

资源	252	7.4.2 绑定数据源	274
第 7 章 通过数据源分页	254	本章小结	278
7.1 使用 DataGrid 控件进行 自定义分页	254	资源	278
7.2 后一页/前一页导航	260	第 8 章 现实世界数据访问	279
7.2.1 定义 SQL 代码	260	8.1 复合命令	280
7.2.2 添加第一页按钮和 最后一页按钮	265	8.1.1 实施多表查询	280
7.3 随机页面导航	268	8.1.2 执行跨表更新	286
7.3.1 选择指定页面的记录	269	8.2 数据报告和分类汇总	291
7.3.2 自定义分页器的结构	270	8.2.1 合计	291
7.3.3 构建完全自定义的 个人分页器	271	8.2.2 设置 DataGrid	294
7.4 逻辑导航	273	8.3 大纲/详细视图	299
7.4.1 改变分页器的外观	273	8.3.1 列举客户清单	299
		8.3.2 列举订单	302
		本章小结	305
		资源	306

第三部分 ASP.NET 控件

第 9 章 ASP.NET 循环控件	309	资源	343
9.1 循环控件	309	第 10 章 Web Forms 用户控件	344
9.1.1 循环控件与列表控件	309	10.1 用户控件	344
9.1.2 循环控件与 DataGrid 控件	310	10.1.1 UserControl 类	345
9.2 Repeater 控件	310	10.1.2 将页面转变为用户控件	346
9.2.1 Repeater 控件的编程接口	311	10.2 开发用户控件	348
9.2.2 Repeater 控件模板	313	10.2.1 构建一个 TabStrip 用户控件	349
9.3 使用 Repeater 控件	318	10.2.2 构建用户控件对象模型	352
9.3.1 设计模板	319	10.2.3 处理用户控件事件	357
9.3.2 数据项的可滚动列表	322	10.3 使用用户控件	360
9.3.3 使用自定义类循环	325	10.3.1 构建数据捆绑用户控件	360
9.4 DataList 控件	326	10.3.2 动态加载控件	366
9.4.1 DataList 控件的 编程接口	327	本章小结	369
9.4.2 格式化功能	330	资源	369
9.4.3 DataList 控件的模板	334	第 11 章 ASP.NET 移动控件	370
9.5 使用 DataList 控件	335	11.1 移动控件的体系结构	370
9.5.1 添加对选择的支持	336	11.1.1 多标记语言	371
9.5.2 添加分页支持	340	11.1.2 无线应用程序协议	372
本章小结	342	11.2 移动控件概况	373

11.2.1 无线控件的层次结构	373	11.3.1 CustomerFinder 应用程序.....	389
11.2.2 ASP.NET 移动页面	375	11.3.2 应用程序的导航模型	392
11.2.3 容器控件	380	11.3.3 自适应呈现	396
11.2.4 列表控件	381	11.3.4 状态管理	397
11.2.5 文本控件	384	本章小结	399
11.2.6 验证控件	386	资源	400
11.3 开发移动应用程序	388		

第四部分 ASP.NET 应用基础

第 12 章 配置和部署	403	13.2.3 跟踪错误和异常	456
12.1 .NET 框架配置	403	13.3 HttpContext 类	458
12.1.1 配置文件的 XML 架构	403	13.3.1 静态成员	459
12.1.2 管理配置设置	407	13.3.2 HttpContext 类的属性	459
12.2 ASP.NET 应用程序配置	409	13.3.3 HttpContext 类的方法	460
12.2.1 ASP.NET 的配置 层次结构.....	410	13.4 Server 对象	461
12.2.2 <location> 节	412	13.4.1 是内部属性而非 内部对象.....	461
12.2.3 <system.web> 节	413	13.4.2 HttpServerUtility 类 的属性.....	462
12.2.4 读取用户配置数据	430	13.4.3 HttpServerUtility 类 的方法.....	462
12.2.5 创建新的配置节	432	13.5 Response 对象.....	467
12.2.6 保持应用程序设置	434	13.5.1 HttpResponse 类的属性.....	467
12.3 ASP.NET 应用程序部署	437	13.5.2 HttpResponse 类的方法.....	470
12.3.1 创建安装程序	438	13.6 Request 对象	471
12.3.2 更新配置选项	441	13.6.1 HttpRequest 类的属性	472
本章小结	442	13.6.2 HttpRequest 类的方法	474
资源	443	本章小结	475
第 13 章 HTTP 请求上下文	444	资源	476
13.1 应用程序初始化	444	第 14 章 ASP.NET 状态管理	477
13.1.1 HttpApplication 类 的属性.....	445	14.1 应用程序状态	478
13.1.2 应用程序模块	446	14.1.1 HttpApplicationState 类 的属性.....	478
13.1.3 HttpApplication 类 的方法.....	447	14.1.2 HttpApplicationState 类 的方法.....	479
13.1.4 HttpApplication 类 的事件.....	448	14.1.3 状态同步	480
13.2 global.asax 文件	451	14.1.4 保存应用程序状态 的代价.....	480
13.2.1 编译 global.asax	451		
13.2.2 global.asax 语法	453		

14.2 会话状态.....481	15.2.2 ASPNET 账户.....537
14.2.1 会话状态 HTTP 模块.....481	15.2.3 改变 ASP.NET 进程标识.....542
14.2.2 HttpSessionState 类 的属性.....484	15.2.4 代码访问安全性和 CLR.....544
14.2.3 HttpSessionState 类 的方法.....485	15.2.5 使用身份验证和授权.....547
14.2.4 标识一个会话.....486	15.2.6 ASP.NET 和 SQL Server.....549
14.2.5 会话生存期.....489	15.3 Windows 身份验证.....551
14.2.6 保存会话数据到 远程服务器.....491	15.3.1 通过 IIS 对用户进行 身份验证.....551
14.2.7 保存会话状态数据到 SQL Server.....496	15.3.2 Windows 身份验证 提供程序.....553
14.2.8 工作于 Web Farm 方案.....498	15.4 Passport 身份验证.....556
14.3 页面的视图状态.....499	15.4.1 理解 Passport 倡议.....556
14.3.1 StateBag 类.....499	15.4.2 Passport 身份验证 提供程序.....556
14.3.2 视图状态的常见问题.....500	15.5 Forms 身份验证.....557
14.3.3 在服务器上保存 视图状态.....506	15.5.1 Forms 身份验证 提供程序.....557
14.4 ASP.NET 缓存系统.....509	15.5.2 登录过程.....561
14.4.1 Cache 类.....509	15.5.3 基于角色的身份验证.....565
14.4.2 使用 ASP.NET 缓存.....511	15.5.4 URL 授权.....567
14.4.3 生成缓存查看器.....517	本章小结.....569
14.5 缓存 ASP.NET 页面.....519	资源.....570
14.5.1 @OutputCache 指令.....519	第 16 章 使用文件系统.....571
14.5.2 HttpCachePolicy 类.....522	16.1 .NET 框架中的文件 I/O.....571
14.5.3 缓存页面的多个版本.....524	16.1.1 使用文件.....571
14.5.4 缓存 ASP.NET 页面 的一部分.....526	16.1.2 使用目录.....577
本章小结.....528	16.1.3 使用路径名.....581
资源.....529	16.1.4 读取器和写入器.....583
第 15 章 ASP.NET 安全.....530	16.2 读取本地文件.....586
15.1 编写安全的代码.....530	16.2.1 StreamReader 类.....586
15.1.1 Web 攻击类型.....531	16.2.2 BinaryReader 类.....591
15.1.2 编写安全代码的 黄金规则.....532	16.3 写入本地文件.....593
15.2 配置 ASP.NET 应用程序 的安全性.....535	16.3.1 StreamWriter 类.....593
15.2.1 选择强口令.....536	16.3.2 写二进制文件.....595
	16.3.3 序列化数据结构.....597
	16.3.4 监视文件变化.....600
	16.4 加载 XML 文档.....603
	16.4.1 读取 XML 文件.....604

16.4.2 XML DOM 解析器	608	17.1.3 Web 服务的基础设施	623
16.5 创建 XML 文档	610	17.2 创建一个 ASP.NET Web 服务	626
16.5.1 使用 XML 写入器	611	17.2.1 编写一个 Web 服务类	626
16.5.2 使用 XML DOM 写入器	611	17.2.2 发布 Web 方法	627
16.5.3 ASP.NET 中的 XML 文档	612	17.2.3 引用 ASP.NET Web 服务	633
16.6 服务器端 XML 转换	613	17.2.4 Web 方法的最佳实践	639
16.6.1 Xml 服务器端控件编程	613	17.3 Web 服务的安全	646
16.6.2 使用 Xml 服务器端控件	614	17.3.1 HTTP 用户身份验证	647
本章小结	614	17.3.2 自定义用户身份验证	649
资源	615	17.3.3 使用应用程序特定令牌	653
第 17 章 使用 Web 服务	616	17.4 管理 Web 服务状态	655
17.1 Web 服务的基础设施	617	17.4.1 启用会话状态	655
17.1.1 SOAP 协议	617	17.4.2 使用 Cookie 容器	655
17.1.2 WSDL 语言	620	本章小结	657
		资源	658

第五部分 自定义 ASP.NET 控件

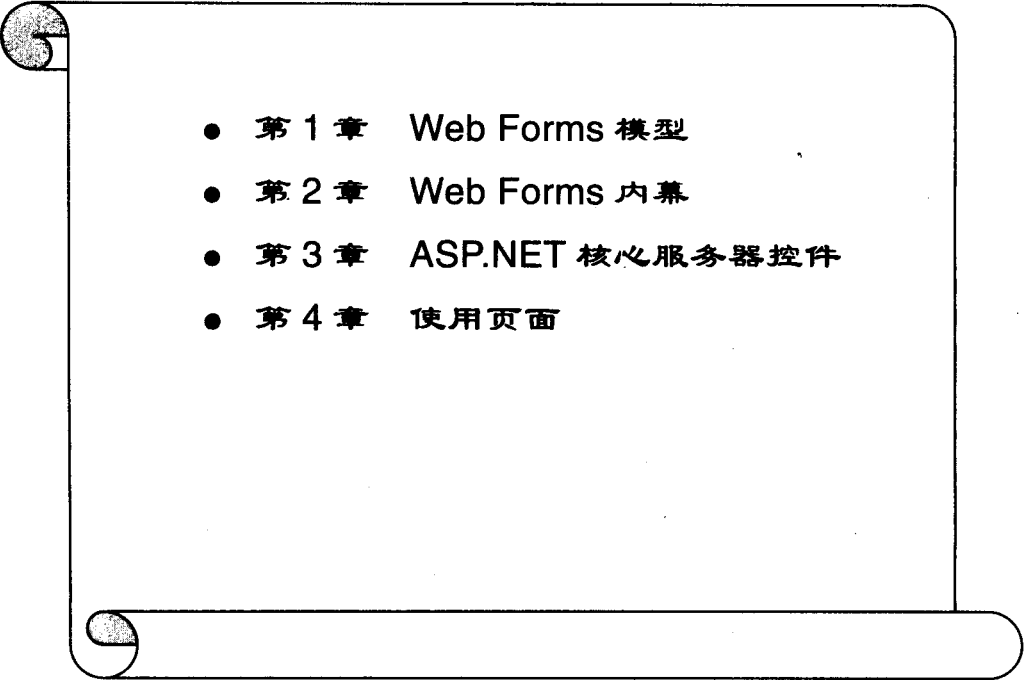
第 18 章 扩展现有的 ASP.NET 控件	661	19.2.1 定义对象模型	691
18.1 更好的 Label 控件	662	19.2.2 呈现控件	695
18.1.1 LinkLabel 服务器控件	662	19.2.3 运行中的量表	696
18.1.2 使用 LinkLabel 控件	666	19.3 创建图形生成器控件	697
18.2 一种更好的 DataGrid 控件	670	19.3.1 定义对象模型	697
18.2.1 改进页面导航栏	670	19.3.2 呈现控件	702
18.2.2 改进页脚	676	19.3.3 运行中的 BarChart 控件	707
18.3 更好的服务器端窗体控件	680	19.4 创建复合控件	709
18.3.1 支持多个服务器窗体	680	19.4.1 复合控件的结构	709
18.3.2 创建多个类似窗体 的控件	683	19.4.2 设计复合控件	710
本章小结	685	19.4.3 使用 LabelTextBox 控件	714
资源	686	本章小结	715
第 19 章 创建新的 ASP.NET 控件	687	资源	715
19.1 基本控件类和接口	687	第 20 章 数据绑定和模板控件	716
19.1.1 Control 和 WebControl	687	20.1 设计数据绑定控件	716
19.1.2 相关接口	688	20.1.1 使 BarChart 控件 可以绑定数据	717
19.1.3 HTML 文本编写器	689	20.1.2 使用数据绑定 BarChart 控件	720
19.2 创建量表控件	690		

20.2 设计模板控件.....722	21.1.1 组件、站点、容器 和控件.....739
20.2.1 模板 BarChart 控件.....723	21.1.2 控件和设计器.....740
20.2.2 使用模板 BarChart 控件.....728	21.1.3 GraphicPanel 控件.....741
20.3 Control Bulider 类.....731	21.2 设计时属性.....744
20.3.1 设计 TextBoxList 控件.....731	21.2.1 使用设计时属性.....744
20.3.2 实现 TextBoxList 控件.....732	21.2.2 扩展 IntelliSense 支持.....747
20.3.3 使用 TextBoxList 控件.....736	21.2.3 呈现类型.....751
本章小结.....738	21.2.4 实现类型转换器.....751
资源.....738	21.2.5 创建类型编辑器.....758
第 21 章 针对自定义控件的 设计时支持.....739	21.2.6 编写自定义设计器.....760
21.1 .NET 中的设计时架构.....739	本章小结.....765
	资源.....765
第六部分 高级操作	
第 22 章 在 ASP.NET 中处理图像.....769	
22.1 从 Web 页面访问图像.....769	23.2.2 非常简单的 HTTP 处理程序.....800
22.1.1 元素.....770	23.2.3 用于快速数据报告的同步 HTTP 处理程序.....801
22.1.2 引用非文件图像.....770	23.2.4 高级 HTTP 处理程序编程.....807
22.2 访问数据库图像.....772	23.3 编写 HTTP 模块.....810
22.2.1 处理二进制文件.....772	23.3.1 IHttpModule 接口.....811
22.2.2 实际情况.....776	23.3.2 自定义 HTTP 模块.....811
22.2.3 优化数据库查询.....776	本章小结.....814
22.3 动态生成图像.....778	资源.....815
22.3.1 GDI+类概览.....779	第 24 章 在 IIS 之外寄宿 ASP.NET.....816
22.3.2 写入图像到内存.....782	24.1 寄宿 ASP.NET 运行库.....816
22.3.3 创建图形生成器页面.....788	24.1.1 ApplicationHost 类.....818
本章小结.....792	24.1.2 SimpleWorkerRequest 类.....819
资源.....792	24.1.3 一个示例 ASP.NET 宿主.....821
第 23 章 HTTP 处理程序和模块.....793	24.2 Cassini 个人 Web 服务器.....824
23.1 IIS 可扩展 API 概览.....793	24.2.1 Cassini 介绍.....824
23.1.1 ISAPI 模型.....794	24.2.2 Cassini 的体系结构.....825
23.1.2 ISAPI 组件的结构.....796	24.3 将 Web 站点内容部署到 CD 上.....827
23.1.3 ISAPI 扩展.....796	
23.1.4 ISAPI 过滤器.....796	
23.2 编写 HTTP 处理程序.....797	
23.2.1 IHttpHandler 接口.....798	

24.3.1 逐步操作过程	828	本章小结	830
24.3.2 在没有 IIS 的情况下		资源	830
安装 ASP.NET	829	结束语	831

第一部分

构建 ASP.NET 页面

- 
- 第 1 章 Web Forms 模型
 - 第 2 章 Web Forms 内幕
 - 第 3 章 ASP.NET 核心服务器控件
 - 第 4 章 使用页面

