

微软精品课程  
系列

.NET  
技术大系

# 基于.NET的 Web程序设计 ——ASP.NET标准教程

刘振岩 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

微软精品课程  
系列

**.NET**  
技术大系

# 基于.NET的 Web程序设计 ——ASP.NET标准教程

刘振岩 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书以 ASP.NET 的基础知识和应用为主要讨论对象,以浅显的方式讲解 ASP.NET 的基础知识,并包含了相应的实例。以 HTML、DHTML、XML、C#入手,在夯实基础的同时由浅入深、循序渐进地对 ASP.NET 的具体应用进行较为详细的讨论。通过对实例的学习,读者可以更好地领会 ASP.NET 语法和编程技巧,有助于将所学的知识融会贯通。

本书内容丰富、语言简洁、通俗易懂,主要面向初、中级用户,对高级用户也有参考价值。本书特别适合作为高等院校计算机及相关专业的教学用书,也可作为 Web 程序设计的培训教材。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

### 图书在版编目(CIP)数据

基于.NET的Web程序设计:ASP.NET标准教程/刘振岩编著. —北京:电子工业出版社,2006.5

(.NET技术大系)

ISBN 7-121-02395-4

I. 基… II. 刘… III. 主页制作—程序设计—教材 IV. TP393.092

中国版本图书馆CIP数据核字(2005)第021996号

责任编辑:郭立 韩明

印刷:北京天宇星印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编100036

经销:各地新华书店

开本:787×1092 1/16 印张:20.25 字数:488千字

印次:2006年5月第1次印刷

印数:5000册 定价:29.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话:(010)68279077。质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

# 序 言

为响应教育部推出的“国家精品课程建设项目”精神，2004年春季，微软亚洲研究院推出了“微软精品课程支持计划”。该计划旨在帮助提高IT相关课程教学质量，帮助高校基础性课程与企业的最新技术相结合。

该计划自推行以来，受到各高校老师的热烈响应。在微软亚洲研究院的支持下，高校一线教师不断努力创新，完善教学内容，提高教学质量，并将教学成果及时总结、共享。目前，多门经过系统整理、更新的课件资源共享在因特网上，其中的一部分还被吸纳到微软全球课程资源库中，供全球各高校教师交流、共享。在精品课件的基础上，经过几轮课程的教学工作，很多重点院校的教师，将课程进行了更系统的总结，整理成系列的教材及图书。这样，也就有了我们今天看到的这一系列内容上兼备基础性、科学性和前瞻性的丛书。

目前，丛书的其他分册正在编写过程中，在不久的将来会相继出版，在此我代表微软亚洲研究院感谢丛书各个编写小组教师的辛勤工作；感谢电子工业出版社的大力支持；希望本套丛书能够对相关的高校学生、专业工作者有所帮助，对中国软件产业起到很好的促进作用，发挥应有的效果。

今后，微软将一如既往地支持高校教学工作的开展，帮助高校及时了解微软最新技术，促进微软的核心技术与高校课程建设的紧密结合。而在此基础上，微软也将继续支持一线教师写出更多更好的书籍，为中国信息产业的发展尽自己的一份力量。

微软亚洲研究院 院长

沈向洋

# 前 言

随着 Internet 的飞速发展,单纯的由 HTML 所构建的静态网页已经不能满足需求。这种静态网页只能单纯地显示网页内容,而无法与使用者产生互动,不能实时地对不同的网页浏览状况做出响应,如不同的浏览者、不同的浏览行为或不同的浏览时间等。人们更需要动态的、交互的网页。为了让网页能依照不同的情况做出动态的响应,在网页中加入程序建立动态响应机制成了网页制作技术的主要发展方向。

所谓 Web 应用程序就是网页中的可以提供动态响应机制的程序。Web 应用程序提供动态信息,而不是静态的 HTML 文件,用户的输入或身份等都可以决定浏览器的显示内容。依照程序执行位置的不同,可以将 Web 应用程序分为客户端和服务端两大类。

目前,编写客户端 Web 应用程序所采用的主要技术是 DHTML 技术。DHTML 除了具备传统的 HTML 所有的功能之外,还具有动画、多媒体以及动态交互能力。本书通过一个综合实例学习如何使用 DHTML。服务器端 Web 应用程序所采用的技术多种多样,本书选用由 Microsoft 最新推出的 ASP.NET,这也是本书的重点内容。微软公司的 ASP.NET 是一个用于 Web 应用开发的全新框架,和 ASP 相比,它在运行速度、易于开发性、易于维护性和可靠性等方面都有显著的提高,具有比其他 Web 开发模式更强大的优势,目前正逐渐被广大程序员普遍采用。

## 本书内容

全书共分为 15 章,首先是基于 .NET 的 Web 应用程序设计的基础知识,然后是 Web 应用程序开发技术的详细介绍。这既包括客户端的 Web 程序开发技术——DHTML,又包括服务器端的 Web 应用程序开发技术——ASP.NET,还包括 ASP.NET 所采用的 C# 语言,以及进行 ASP.NET 程序设计必备的 XML 和 ADO.NET 相关知识。

全书的内容安排如下。

**第 1 章 基于 .NET 的 Web 程序设计的基础知识** 介绍 Web 应用程序基本知识, .NET Framework 及其体系结构, ASP.NET 的重要特色。

**第 2 章 HTML** 讲解 HTML 的基本语法,主要包括 HTML 文件的基本结构和一些常用的标记。

**第 3 章 用实例学习 DHTML** 基于一个如何创建电子图书的实例展开讲解,主要包括 JavaScript、DOM、CSS。

**第 4 章 XML** 介绍 XML 基本知识,主要包括 XML 文件结构和基本语法、XSL、DTD、XML Schema。

**第 5 章 C# 语言基础** 介绍 C# 语言的基本语法和简单使用。

**第 6 章 ASP.NET 开发环境和工具** 建立 ASP.NET 开发环境,记事本、DreamWeaver MX 和 Visual Studio.NET 等开发工具的使用。

**第 7 章 ASP.NET Web Forms** 用实例解析 Web Forms 的关键技术,介绍 ASP.NET 常用的内部对象的使用。

**第 8 章 Web 服务器控件** 介绍 ASP.NET 服务器控件的类型, 如何使用基本的 Web 服务器控件和数据验证控件等。

**第 9 章 ADO.NET** 介绍 SQL Server 2000 安装与使用、SQL 语言基础, 剖析 ADO.NET、数据绑定技术, 如何使用 Connection、Command、DataReader、DataAdapter 和 DataSet 等常用的对象。

**第 10 章 数据访问控件** 介绍 DataGrid、Repeater 和 DataList 控件的使用。

**第 11 章 状态管理** 介绍 ASP.NET 提供的多种状态管理方式, 以及如何使用 Global.asax 文件。

**第 12 章 配置应用程序** 介绍 ASP.NET 配置系统的工作原理, 以及如何使用 Web.config 配置应用程序。

**第 13 章 .NET 中的 XML** 主要介绍如何使用 System.Xml 命名空间中两个最常用的类, 即 XmlDocument 类和 XmlDataDocument 类。

**第 14 章 Web Service** 介绍 Web Service 的主要技术, 以及如何创建和使用 Web 服务。

**第 15 章 综合实例——BBS** 通过完成一个较大型的实例, 达到综合运用所学知识的目的。

## 本书特点

本书内容丰富, 语言简洁, 通俗易懂, 讲解深入浅出。本书最大特点是实用性, 目的是让读者触类旁通。对于每一个知识点, 选用恰当的实例, 以明确的步骤指导读者完成实例, 从而使读者在实际操作中掌握知识, 真正做到边学边用, 达到事半功倍的效果。

## 读者对象

本书主要面向初、中级用户, 对高级用户也有参考价值。本书特别适合作为高等院校计算机及相关专业的教学用书, 也可作为 Web 程序设计的培训教材。

## 关于本书

本书的编写得到“微软精品课程支持计划”的支持, 该计划是由微软亚洲研究院推出的。微软公司与教育部签署了“长城计划”, 陆续开展了教师培训、教材编写、教学联盟社区建设等大量工作, 帮助教师共享资源、提高教学质量。“微软精品课程支持计划”是“长城计划”的一个重要内容, 旨在帮助高等院校更好地开设微软技术相关课程, 帮助教师实现教学资源共享, 创建示范性精品课程。

在“微软精品课程计划”的大力支持下, 本书作者积极参与了微软亚洲研究院高校关系部主办的微软教师互动社区活动, 通过多种形式, 如培训、讲座等, 与微软教学联盟的教师们及来自微软总部的工程师们沟通、交流。并在总结教学经验的基础上, 精心组织编写了此书, 以期加强资源共享, 进一步提高教学质量。

为了便于阅读理解, 本书做如下约定。

- “【新建】|【项目】”表示先选择【新建】菜单, 然后再选择【项目】命令;
- 没有特指时, 单击、双击、拖动均指用鼠标的左键进行操作。

作者

2006年2月28日

# 目 录

第 1 章 基于 .NET 的 Web 程序设计的 基础知识 .....	1	3.2.2 JavaScript 基本语法 .....	38
1.1 Web 程序设计概述 .....	1	3.3 事件处理和文档对象模型 (DOM) .....	41
1.1.1 什么是 Web 应用程序 .....	1	3.3.1 事件处理 .....	42
1.1.2 服务器端 Web 开发技术 .....	3	3.3.2 文档对象模型 (DOM) .....	44
1.2 ASP.NET 概述 .....	3	3.4 级联式样式表 (CSS) .....	48
1.2.1 .NET Framework .....	3	3.4.1 使用 CSS .....	48
1.2.2 什么是 ASP.NET .....	5	3.4.2 定义 CSS .....	50
1.2.3 ASP.NET 的特色 .....	5	3.5 本章小结 .....	57
1.3 本章小结 .....	6	3.6 习题 .....	57
1.4 习题 .....	6	第 4 章 XML .....	58
第 2 章 HTML .....	7	4.1 XML 概述 .....	58
2.1 HTML 文件的基本结构 .....	7	4.1.1 HTML 的局限性 .....	58
2.2 构成网页的基本元素 .....	10	4.1.2 不同于 HTML 的 XML .....	59
2.3 超级链接标记 .....	13	4.1.3 XML 的主要特性 .....	60
2.4 在 HTML 文件中使用图像 .....	15	4.1.4 XML 与 ASP.NET 的联系 .....	61
2.5 框架结构的使用 .....	16	4.2 XML 文件结构和基本语法 .....	61
2.6 HTML 中的表格 .....	22	4.2.1 XML 文件结构 .....	61
2.7 列表格式的使用 .....	26	4.2.2 XML 的基本语法 .....	62
2.8 表单的使用 .....	28	4.3 XML 与 CSS .....	63
2.9 本章小结 .....	32	4.4 XSL .....	64
2.10 习题 .....	32	4.5 DTD .....	66
第 3 章 用实例学习 DHTML .....	33	4.6 XML Schema .....	69
3.1 创建电子图书 .....	33	4.7 本章小结 .....	71
3.1.1 浏览电子图书 .....	33	4.8 习题 .....	71
3.1.2 电子图书的实现步骤 .....	34	第 5 章 C# 语言基础 .....	72
3.2 JavaScript .....	37	5.1 数据类型 .....	72
3.2.1 在 HTML 中使用 JavaScript .....	37	5.2 操作符 .....	73
		5.2.1 常用的操作符 .....	73

5.2.2 操作符的优先级 .....	74	8.2 基本 Web 服务器控件 .....	103
5.3 控制语句 .....	74	8.2.1 Label 控件 .....	103
5.3.1 选择语句 .....	74	8.2.2 Button 控件 .....	103
5.3.2 循环语句 .....	75	8.2.3 LinkButton 控件 .....	104
5.3.3 跳出、继续和转向语句 .....	76	8.2.4 ImageButton 控件 .....	106
5.4 命名空间 .....	77	8.2.5 TextBox 控件 .....	107
5.4.1 什么是命名空间 .....	77	8.2.6 DropDownList 控件 .....	109
5.4.2 命名空间的声明和调用 .....	77	8.2.7 ListBox 控件 .....	111
5.5 C#实例 .....	78	8.2.8 CheckBox 控件和 CheckBoxList 控件 .....	113
5.6 本章小结 .....	78	8.2.9 RadioButton 控件和 RadioButtonList 控件 .....	115
5.7 习题 .....	78	8.2.10 Panel 控件 .....	117
<b>第 6 章 ASP.NET 开发环境和工具 .....</b>	<b>79</b>	8.3 数据验证控件 .....	119
6.1 程序开发环境的建立 .....	79	8.3.1 数据验证机制介绍 .....	119
6.1.1 单机仿真开发环境的建构 .....	79	8.3.2 RequiredFieldValidator 控件 .....	120
6.1.2 Web 站点的基本概念 .....	80	8.3.3 CompareValidator 控件 .....	121
6.2 不同开发工具的使用 .....	81	8.3.4 RangeValidator 控件 .....	122
6.2.1 记事本 .....	81	8.3.5 RegularExpressionValidator 控件 .....	124
6.2.2 DreamWeaver MX .....	83	8.3.6 CustomValidator 控件 .....	125
6.2.3 Visual Studio.NET .....	84	8.3.7 ValidationSummary 控件 .....	127
6.3 本章小结 .....	86	8.4 本章小结 .....	128
6.4 习题 .....	87	8.5 习题 .....	128
<b>第 7 章 ASP.NET Web Forms .....</b>	<b>88</b>	<b>第 9 章 ADO.NET .....</b>	<b>129</b>
7.1 创建 Web Forms .....	88	9.1 数据库服务器 .....	129
7.1.1 Web Forms 的代码组织方式 .....	88	9.1.1 SQL Server 2000 .....	129
7.1.2 Web Forms 的文件结构 .....	94	9.1.2 MSDE 2000 .....	133
7.1.3 Web Forms 的事件模型 .....	94	9.2 SQL 语言基础 .....	135
7.2 ASP.NET 的内部对象 .....	97	9.2.1 SQL 语言简介 .....	135
7.2.1 Response 对象和 Request 对象 .....	98	9.2.2 数据查询语言 .....	136
7.2.2 Server 对象 .....	101	9.2.3 数据处理语言 .....	142
7.3 本章小结 .....	101	9.3 ADO.NET .....	144
7.4 习题 .....	101	9.3.1 ADO.NET 简介 .....	144
<b>第 8 章 Web 服务器控件 .....</b>	<b>102</b>	9.3.2 ADO.NET 结构 .....	145
8.1 ASP.NET 服务器控件简介 .....	102	9.3.3 Connection .....	148
		9.3.4 Command & DataReader .....	151



9.3.5	DataAdapter & DataSet	158	11.4.2	启用会话状态	213
9.4	数据绑定	161	11.4.3	Session 对象的使用方法	214
9.4.1	数据绑定概述	161	11.4.4	Session 对象应用—— 购物车	214
9.4.2	绑定数组	161	11.5	视图状态	219
9.4.3	绑定数据库	162	11.5.1	视图状态概述	219
9.5	本章小结	167	11.5.2	ViewState 的使用方法	220
9.6	习题	167	11.5.3	ViewState 的应用—— DataGrid 的分页排序	220
<b>第 10 章</b>	<b>数据访问控件</b>	<b>168</b>	11.6	Cookie	223
10.1	用实例学 DataGrid 控件	168	11.6.1	什么是 Cookie	223
10.1.1	实例——图书信息管理 系统	168	11.6.2	Cookie 的使用方法	224
10.1.2	创建数据库	169	11.6.3	Cookie 的应用——用户 到访统计	224
10.1.3	新书登记	169	11.7	本章小结	226
10.1.4	显示图书信息	171	11.8	习题	227
10.1.5	图书信息排序	179	<b>第 12 章</b>	<b>配置应用程序</b>	<b>228</b>
10.1.6	分页浏览图书	182	12.1	配置概述	228
10.1.7	修改图书信息	184	12.2	分层配置结构	229
10.2	Repeater 控件	192	12.3	Web.config 文件	230
10.2.1	Repeater 控件简介	192	12.3.1	Web.config 的组成与结构	230
10.2.2	Repeater 控件的应用	193	12.3.2	常用的配置元素	232
10.3	DataList 控件	197	12.4	锁定配置设置	236
10.3.1	DataList 控件简介	197	12.5	自定义应用程序设置	238
10.3.2	DataList 控件的应用	198	12.6	本章小结	240
10.4	本章小结	201	12.7	习题	240
10.5	习题	202	<b>第 13 章</b>	<b>.NET 中的 XML</b>	<b>241</b>
<b>第 11 章</b>	<b>状态管理</b>	<b>203</b>	13.1	创建 XML 文件	241
11.1	状态管理分类	203	13.2	XmlDocument	242
11.2	Global.asax 文件	204	13.3	XmlDataDocument	245
11.3	应用程序状态	209	13.3.1	加载 XML 数据	245
11.3.1	应用程序状态概述	209	13.3.2	XmlDataDocument 与 DataSet	247
11.3.2	Application 对象的使用 方法	210	13.4	将 XML 数据导入到数据库	252
11.3.3	Application 对象应用—— 在线人数统计	210	13.5	本章小结	259
11.4	会话状态	212	13.6	习题	259
11.4.1	会话状态概述	212			

第 14 章 Web Service .....	260	14.5 习题 .....	273
14.1 Web Service 概述 .....	260	<b>第 15 章 综合实例——BBS .....</b>	<b>274</b>
14.1.1 为什么需要 Web Service .....	260	15.1 实例功能介绍 .....	274
14.1.2 什么是 Web Service .....	261	15.2 实现步骤 .....	275
14.1.3 何时需要 Web Service .....	261	15.2.1 创建数据库 .....	275
14.2 Web Service 的体系结构和 关键技术 .....	263	15.2.2 Web.config .....	275
14.2.1 Web Service 的体系结构 .....	263	15.2.3 用户登录 .....	275
14.2.2 SOAP .....	264	15.2.4 用户注册 .....	278
14.2.3 WSDL .....	265	15.2.5 显示所有留言 .....	286
14.2.4 UDDI .....	265	15.2.6 创建新的留言 .....	290
14.3 Web Service 开发与使用 .....	265	15.2.7 查看或回复某个主题的 留言 .....	294
14.3.1 创建并测试 Web Service .....	266	15.3 实例运行 .....	302
14.3.2 公布 Web Service .....	270	15.4 本章小结 .....	305
14.3.3 使用 Web Service .....	271	15.5 习题 .....	305
14.4 本章小结 .....	273		

## 实例目录

实例 2-1	第一个 HTML 文件 .....	7
实例 2-2	<Hn >标记的使用 1 .....	10
实例 2-3	<Hn >标记的使用 2 .....	10
实例 2-4	<PRE >标记的使用 .....	11
实例 2-5	 和<P>标记的使用 .....	12
实例 2-6	<HR>标记的使用 .....	13
实例 2-7	<A>标记的使用 .....	14
实例 2-8	<IMG>标记的使用 1 .....	15
实例 2-9	<IMG>标记的使用 2 .....	16
实例 2-10	使用框架结构组织网页 .....	16
实例 2-11	页内框架的使用 .....	21
实例 2-12	表格的使用 .....	22
实例 2-13	表格边框属性的使用 .....	24
实例 2-14	合并表格的单元格 .....	25
实例 2-15	无序列表的建立 .....	26
实例 2-16	有序列表的建立 .....	27
实例 2-17	说明列表的建立 .....	28
实例 2-18	表单的使用 .....	29
实例 3-1	综合运用 DHTML 制作电子书 .....	33
实例 3-2	练习 JavaScript 的事件处理 .....	42
实例 3-3	Alert、Confirm 和 Prompt 方法的使用 .....	45
实例 3-4	Open 和 Navigate 方法的使用 .....	47
实例 3-5	不使用 CSS 设置网页格式 .....	48
实例 3-6	使用 CSS 设置网页格式 .....	49
实例 3-7	具有上下文关系的 HTML 标记样式的使用 .....	52
实例 3-8	用户定义类的使用 .....	53
实例 3-9	虚类的使用 .....	56
实例 4-1	利用 CSS 显示 XML 文件 .....	63
实例 4-2	利用 XSL 显示 XML 文件 .....	64
实例 4-3	内部 DTD 的使用 .....	67
实例 4-4	外部 DTD 的使用 .....	68

实例 4-5	XML Schema 的使用 .....	70
实例 5-1	一个简单的 C#编程实例 .....	78
实例 6-1	ASP.NET 开发环境的建构——如何安装 Web 服务器 (IIS) .....	79
实例 6-2	如何利用记事本开发 ASP.NET Web 应用程序 .....	81
实例 6-3	如何利用 Dreamweaver MX 开发 ASP.NET Web 应用程序 .....	83
实例 6-4	如何利用 Visual Studio.NET 开发 ASP.NET Web 应用程序 .....	85
实例 7-1	不使用 CodeBehind 技术编写第一个简单的应用程序 .....	88
实例 7-2	使用 CodeBehind 技术编写第一个简单的应用程序 .....	90
实例 7-3	如何实现一个按钮的 Click 事件 .....	94
实例 7-4	使用 Response 对象和 Request 对象在不同页面之间传递变量 .....	98
实例 7-5	使用 Server 对象将文件的虚拟路径对应其实际路径 .....	101
实例 8-1	使用 Button 控件动态控制 Label 控件显示信息 .....	103
实例 8-2	动态改变 Link Button 控件的外观 .....	105
实例 8-3	动态改变 ImageButton 控件的外观 .....	106
实例 8-4	使用 TextBox 控件实现用户身份验证 .....	108
实例 8-5	实现 DropDownList 控件选项更改后的自动回发 .....	109
实例 8-6	实现 ListBox 控件选项的选取、添加和删除操作 .....	111
实例 8-7	使用 CheckBox 控件和 CheckBoxList 控件创建多个复选框 .....	114
实例 8-8	使用 RadioButton 控件和 RadioButtonList 控件创建多个单选按钮 .....	116
实例 8-9	使用 Panel 控件隐藏或显示一组控件 .....	117
实例 8-10	使用 RequiredFieldValidator 控件验证用户是否输入信息 .....	120
实例 8-11	使用 CompareValidator 控件验证用户两次输入的密码是否一致 .....	121
实例 8-12	使用 RangeValidator 控件验证用户的输入是否在指定的范围内 .....	123
实例 8-13	使用 RegularExpressionValidator 控件验证用户的输入是否符合电子邮件地址格式 .....	124
实例 8-14	使用 CustomValidator 控件验证用户的输入是否为偶数 .....	125
实例 8-15	使用 ValidationSummary 控件对网页进行综合验证 .....	127
实例 9-1	如何安装 SQL Server 2000 .....	130
实例 9-2	如何安装 MSDE 2000 .....	134
实例 9-3	查询 SQL Server 数据库中的数据信息 .....	136
实例 9-4	如何在 SQL Server 2000 中创建一个数据库 .....	142
实例 9-5	如何为 SQL Server 数据库添加、删除和更新数据 .....	143
实例 9-6	如何实现应用程序与 SQL Server 数据库的非信任连接 .....	148
实例 9-7	如何实现应用程序与 SQL Server 数据库的信任连接 .....	150
实例 9-8	Command 对象和 DataReader 对象结合使用访问数据库 .....	152

实例 9-9	使用 Command 对象的 ExecuteScalar 方法	154
实例 9-10	使用 Command 对象的 ExecuteNonQuery 方法	155
实例 9-11	实例 9-10 的扩展——简易留言板的制作	156
实例 9-12	使用 DataAdapter 对象将数据填充到 DataSet 中并显示	158
实例 9-13	如何将 DataSet 中改动的数据更新回数据库	160
实例 9-14	如何将数组中的数据绑定到 DropDownList 控件	162
实例 9-15	如何将数据库中的数据绑定到控件——一个简易的数据查询系统	163
<hr/>		
实例 10-1	用 DataGrid 控件实现图书信息管理系统	168
实例 10-2	用 Repeater 控件显示 Northwind 数据库中的客户信息	193
实例 10-3	用 DataList 控件显示 Northwind 数据库中的客户信息	198
<hr/>		
实例 11-1	应用程序级别事件的触发顺序模拟	206
实例 11-2	使用 Application 对象实现网络聊天室的在线人数统计	210
实例 11-3	使用 Session 对象实现网上购书系统的购物车	214
实例 11-4	使用 ViewState 实现 DataGrid 的分页排序	220
实例 11-5	使用 Cookie 统计用户访问站点的次数	225
<hr/>		
实例 12-1	如何声明配置节处理程序	230
实例 12-2	如何对配置节进行分组	231
实例 12-3	如何将配置应用到特定的子目录	237
实例 12-4	如何锁定一组配置设置	237
实例 12-5	如何在 Web.config 中设置应用程序的数据库连接字符串	238
<hr/>		
实例 13-1	如何创建一个 XML 文件	241
实例 13-2	如何使用 XmlDocument 和 DOM 读取 XML 文档	242
实例 13-3	如何使用 XmlDataDocument 加载 XML 数据	245
实例 13-4	如何将 XML 数据加载到 DataSet	248
实例 13-5	如何实现从 DataSet 关系数据到 XML 数据的转换	250
实例 13-6	如何将 XML 数据导入到数据库	253
<hr/>		
实例 14-1	如何创建一个简单的 Web Service	266
实例 14-2	实例 14-1 的扩展——实现一个可以判断某个整数是否为素数的 Web Service	268
实例 14-3	如何使用一个 Web Service	271

# 第 1 章

## 基于.NET 的 Web 程序设计的基础知识

本章首先简单介绍有关 Web 应用程序的基本知识，然后是 ASP.NET 的概述，主要包括 .NET Framework 简介，什么是 ASP.NET，以及 ASP.NET 几个重要的特色。

### 1.1 Web 程序设计概述

本节首先介绍什么是 Web 应用程序，并依照程序执行位置的不同将其分为客户端和服务端两大类，然后对用于开发服务器端 Web 应用程序的主流技术进行简单介绍。

#### 1.1.1 什么是 Web 应用程序

单纯由 HTML 所构建的静态网页，只能显示网页内容，无法与使用者产生互动，不能实时对不同的网页浏览状况做出响应，如不同的浏览者、浏览行为或浏览时间等。随着 Internet 的飞速发展，这种静态网页已无法满足人们的需求。人们更需要动态、交互的网页。为了让网页能依照不同的情况做出动态的响应，在网页中加入程序建立动态响应的机制，成了网页制作技术的主要发展方向。

所谓 Web 应用程序，就是网页中可以提供动态响应机制的程序。Web 应用程序提供动态信息，而不是静态的 HTML 文件，用户的输入或身份等都可以决定浏览器的显示内容。依照程序执行位置的不同，可以将 Web 应用程序分为客户端和服务端两大类。

客户端和服务端是 WWW 架构的两个主要组成部分。客户端即信息的接收者，是浏览网页的计算机和使用者的总称，而实际执行于计算机上供使用者浏览网页的软件为浏览器，目前常见的有 Microsoft 的 Internet Explorer (简称 IE) 和 Netscape 的 Navigator。服务器端即信息的提供者，在此可以先简单地理解为 Web 服务器。Web 服务器是 WWW 的核心，由它提供各种形式的信息，用户采用 Web 浏览器来使用这些服务。

网页的基本浏览过程：首先由客户端（Web 浏览器）向服务器端（Web 服务器）发送浏览网页的请求，Web 服务器响应这种请求，将其所请求的页面或文档传送给 Web 浏览器，浏览器负责解析此网页并呈现给使用者，如图 1.1 所示。

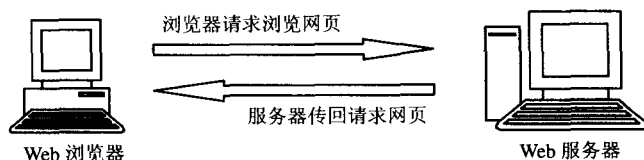


图 1.1 WWW 架构

对于执行在客户端的 Web 应用程序来说，程序的执行过程是从服务器端下载网页到客户端，

由浏览器负责解释并执行，最后将执行后的网页显示出来。目前编写客户端 Web 应用程序所采用的主要技术是 DHTML 技术。DHTML (Dynamic HTML) 并不是一种独立的语言，它是 HTML、CSS 和脚本语言的一种集成。除了具备传统的 HTML 所有的功能之外，DHTML 还具有动画、多媒体和动态交互能力。

### ! 注意

虽然 DHTML 能实现很强大的功能，但由于它是一个新技术，因此只能在比较新的浏览器中得到支持，如 IE 4.0 以上和 Netscape Navigator 4.0 以上。另外，对于不同的浏览器，DHTML 的实现也有所不同，需要编写不同的代码（本书以 IE 为例）。

执行于服务器端的 Web 应用程序最主要的用途是处理从客户端到服务器端的数据。目前，此类 Web 应用程序的体系结构多采用三层或多层的浏览器/服务器 (Browser/Server, 简称 B/S) 结构，这里的服务器通常指 Web 服务器（如 Microsoft 的 IIS）和数据库服务器，所有的应用程序都存储在 Web 服务器上，利用数据库服务器对分布在 Web 服务器上的大量信息进行动态管理，从而使所发布的信息具有交互性、动态性和实时性。

这种三层 B/S 体系结构的示意图如图 1.2 所示。工作原理是：Web 浏览器发出请求，要求访问 Web 服务器的某一网页，Web 服务器检查文件的扩展名是不是服务器端程序要处理的网页，即非 html 文件。如果是，Web 服务器会响应并处理。如果应用程序需要访问数据库，则 Web 服务器会利用相应的数据库访问技术来存取数据库服务器上的数据。如果有数据必须显示在浏览器上，则应用程序会形成动态的 HTML 文档，然后由 Web 服务器传送给前端的客户浏览器。

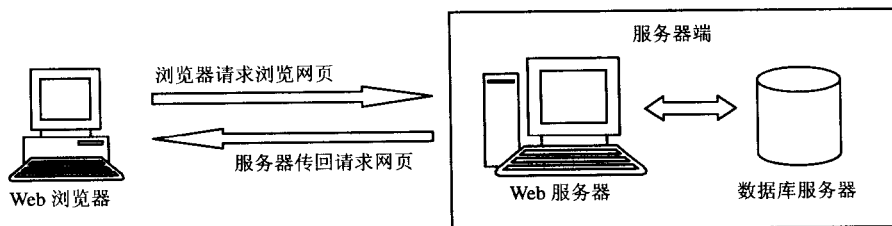


图 1.2 三层 B/S 体系结构示意图

三层 B/S 体系结构的优点如下。

#### 1. 简化了客户端

对于远程的前端用户来讲，不再需要特殊设置与软件安装，仅需要有能够支持 HTML 文档的浏览器（如 IE）即可，这样大大降低了学习障碍与培训成本。

#### 2. 可跨平台操作

各种平台上的用户都可以通过浏览器访问相应的信息，无须为不同的软硬件环境等开发不同的版本。

#### 3. 集中化管理与维护

对于 Web 开发者而言，所有的应用程序都集中在服务器端开发管理，那么当 Web 开发者进行软件改变或升级时，仅需在服务器端进行变动，而无需在远程的前端用户进行任何的操作。对于远程的前端用户来讲，程序的升级几乎是透明的，这样大大减少了系统管理和维护的工作量和成本。

执行于服务器端的 Web 应用程序所采用的技术多种多样,目前比较流行的有 ASP、JSP、PHP、ASP.NET 等。其中,ASP.NET 是由 Microsoft 最新推出的,这也正是本书的重点内容。下面一节将对这些技术进行简单介绍。

### 1.1.2 服务器端 Web 开发技术

目前开发执行于服务器端的 Web 应用程序的技术多种多样,比较流行的有 ASP、JSP、PHP、ASP.NET 等。下面将对它们进行简单的介绍。

- ASP

ASP 全名 Active Server Pages,是一个 Web 服务器端的开发环境,产生和运行动态的、交互的、高性能的 Web 服务应用程序。ASP 采用脚本语言 VBScript 或 JavaScript 作为开发语言的历史并不长,最早出现于 1996 年。

- JSP

JSP (Java Server Page) 是 Sun 公司推出的新一代站点开发语言,SUN 公司借助在 Java 上的不凡造诣,在 Java 应用程序和 Java Applet 之外又增加了一项新的硕果。JSP 可以在 Servlet 和 JavaBean 的支持下,完成功能强大的站点程序。

- PHP

PHP 是一种跨平台的服务器端的嵌入式脚本语言,大量地借用 C、Java 和 perl 语言的语法,并且支持目前绝大多数数据库。还有一点,PHP 是完全免费的,不用花钱就可以直接从 PHP 官方网站 (<http://www.php.net>) 自由下载,而且可以不受限制地获得源码,甚至可以从其中加进所需要的特色。

- ASP.NET

ASP 的出现使得网络应用程序开发变得相当容易,但由于它通过解释执行,而且脚本程序和静态网页混合在一起,所以在程序的执行速度和程序的可移植性上都存在一些问题,出于这样的原因,微软于 2002 年正式推出了 ASP.NET。

ASP.NET 不再采用解释型的脚本语言,而是采用编译型的程序语言,如 VB.NET、C# 等,执行速度加快了许多。ASP.NET 把网页的内容和程序代码分开,这样使得页面的编码看起来井井有条,并且可以重复使用。而 ASP、JSP、PHP 三者都是在 HTML 代码中混合某种程序代码。ASP.NET 还通过使用服务器端控件等技术,使 ASP.NET 面向对象的特征更加明显,所有这些都让 ASP.NET 获得了更高的开发效率。另外,ASP.NET 还拥有许多优点,如更强大的错误处理和调试特性、更好的安全管理机制、更多的组件服务等。

本书所采用的开发技术就是 ASP.NET。下面一节是对 ASP.NET 的基本介绍,本书的后续部分将对其展开详细的讲解。

## 1.2 ASP.NET 概述

ASP.NET 是 Microsoft .NET Framework 的组成部分。要学习 ASP.NET,首先应该对 .NET Framework 有一个大致的了解,因此,本节的开始是关于 .NET Framework 的简介,然后介绍什么是 ASP.NET,以及与其他 Web 开发模型相比,ASP.NET 所具有的重要特色。

### 1.2.1 .NET Framework

.NET Framework 是用于 .NET 平台的编程模型。.NET Framework 的关键组件是公共语言运行



库和 .NET Framework 类库（包括 ADO.NET、ASP.NET 和 Windows 窗体）。.NET Framework 提供了托管执行环境、简化的开发和部署，以及与其他各种编程语言的集成。

.NET Framework 的层次结构如图 1.3 所示。



图 1.3 .NET Framework 的层次结构

- 公共语言运行库（Common Language Runtime, CLR）

.NET Framework 提供一个称为公共语言运行库的运行环境，它运行代码并提供使开发过程更轻松的服务。

公共语言运行库负责运行时的服务，比如语言集成、安全增强，以及内存、子进程和线程的管理。此外，它在开发时也能发挥作用。在这个时候，具有诸如有效期管理、强类型命名、跨语言异常处理、动态绑定等功能。

公共语言运行库的两个主要的目标：

- ◆ 提高应用程序的稳定性和安全性；
- ◆ 减少应用程序开发者所必须写的冗长而又易出错的底层代码的容量。

- .NET Framework 类库

基本类提供标准的功能，如输入/输出、字符串操作、安全管理、网络通信、线程管理、文本管理、用户界面设计等底层功能。其他所有的类库都建立在这个基本类库之上。

ADO.NET 数据访问类支持持续的数据管理。它还包括 SQL 类，可通过标准的 SQL 接口进行持续的数据存储操作。

XML 类可以实现 XML 数据操作和 XML 搜索和转换。

ASP.NET 类支持 Web Forms 应用程序和 Web Service 的开发。

Windows Forms 类支持基于 Windows 的智能客户端应用程序的开发。

所有这些类库合在一起，提供一个跨所有 .NET Framework 所支持语言的公共、一致的开发接口，并且采用清晰而有条理的方式对类库进行分组和描述，这样开发者能更容易地找到他们的应用程序所需的大多数功能。

需要特别强调的是，.NET Framework 支持多种编程语言，并在一个应用程序中可以使用不同的语言。程序员可以使用一种或多种 .NET 编程语言来编写他们的代码，如微软自己的 VB.NET、C++、C# 和 JScript.NET 等。大量其他的 .NET 编程语言可以从第三方厂商获得。现在已经有 20 多种商业和学术编程语言支持 .NET Framework。

微软提供一个免费的 .NET Framework SDK，它包括 CLR、类库、ASP.NET 和用于 C# 与 VB 语言的编译器。使用这个 SDK，开发者可以使用任何编辑器写代码，然后从命令行界面访问编译器和其他工具。

.NET Framework SDK 的下载地址为 <http://www.microsoft.com/download>，下载后直接安装即可。

尽管如此，大多数开发者更喜欢在图形集成开发环境（Integrated Development Environment，