

21 世纪高职高专计算机系列教材

ASP.NET

程序设计

赵增敏 朱粹丹 赵朱曦 编著

西安交通大学出版社

21世纪高职高专计算机系列教材

TP393.4

174

2006

ASP.NET 程序设计

赵增敏 朱粹丹 赵朱曦 编著

西安交通大学出版社
· 西 安 ·

内 容 提 要

本书全面系统地讲述了如何使用 Visual Studio .NET 设计 ASP.NET Web 应用程序。本书共分 9 章,主要内容包括:ASP.NET 使用基础、Visual Basic .NET 编程语言、HTML 服务器控件应用、Web 服务器控件应用、验证控件与用户控件、ASP.NET 基本对象编程、ADO.NET 数据库编程、数据服务控件应用、网络论坛设计实例。本书论述准确、内容翔实、可操作性强,突出实用性、适用性和先进性,提供了丰富的操作实例,深入浅出、循序渐进地引导读者学习。每章后面均附有习题。

本书可作为大中专院校计算机相关专业的教材,也可作为 ASP.NET 中、初级用户以及 Web 应用开发人员的参考书。

图书在版编目(CIP)数据

ASP.NET 程序设计/赵增敏,朱粹丹,赵朱曦编著.
—西安:西安交通大学出版社,2006.10
(21 世纪高职高专计算机系列教材)
ISBN 7-5605-1644-0

I. A... II. ①赵...②朱...③赵... 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 120036 号

书 名:ASP.NET 程序设计

编 著:赵增敏 朱粹丹 赵朱曦

责任编辑:屈晓燕 贺峰涛

出版发行:西安交通大学出版社

地 址:西安市兴庆南路 25 号(邮编:710049)

网 址:<http://press.xjtu.edu.cn>

电 话:(029)82668357 82667874(发行部)

(029)82668315 82669096(总编办)

电子邮箱:eibooks@163.com

印 刷:陕西江源印刷科技有限公司

版 次:2006 年 10 月第 1 版 2006 年 10 月第 1 次印刷

开 本:787 mm×1 092 mm 1/16

印 张:24.5

字 数:593 千字

印 数:0001~3000

书 号:ISBN 7-5605-1644-0/TP·311

定 价:30.00 元

21 世纪高职高专计算机系列教材编委会

顾 问：冯博琴（教育部高等学校非计算机专业计算机课程教学
指导委员会主任委员，陕西省计算机教育学会理事长，西安
交通大学教授，博士生导师）

主 编：陈建铎（陕西省计算机教育学会副理事长兼秘书长，西
安石油大学教授）

副 主 编：谢膺白 王四万 何东健 龚尚福

编 委：（以姓氏笔画为序）

王 津 王四万 王晓奇 何东健 张水平

张俊兰 张晓云 李 青 李银兴 陈建铎

姜 荣 段宏斌 龚尚福 谢膺白 魏玉梅

策划编辑：贺峰涛 屈晓燕

序

随着我国科学技术的发展,全民高等教育已经成为时代的要求。扩大招生规模,发展高等职业教育,已经成为各级政府和广大教育工作者的共识。为了指导和推动全国高等职业教育的健康发展,教育部先后制定了“高职高专教育基础课程教学基本要求”和“高职高专教育专业人才培养目标和规格”两个文件。在此基础上,许多出版社先后出版了相关的系列教材,对推动我国的高等职业教育起到了积极的作用。

但是,时代在前进,科学技术在发展,尤其是计算机信息技术发展的速度更是惊人。这就要求高等学校的教学内容应能跟上科学技术的发展,应能满足新技术对新型人才的需求。因此,其教材应当不断地修改和更新。故此,我们组织高校中长期从事高等职业教育的专家、学者编写了“21 世纪高职高专计算机系列教材”。在编写过程中,我们以教育部上述两个文件为依据,参阅同类教材,汲取多年来在高等专科教育、成人教育中培养应用型人才的成功经验,充分体现高职高专实用型人才的特征,“以应用为目的,以必须、够用为度”,尽量做到从实际应用的需求出发,减少枯燥乏味的纯理论和概念,使学生理论联系实际,学中有用,边学边用。通过学习,提高学生的应用和解决实际问题的能力。在编排顺序方面,尽量做到由浅入深,循序渐进,内容多样,结构合理,语言简练,文字流畅,使学生易学、易懂、易掌握。

这套教材目前已列入选题的有 19 种^①,既有专业基础知识,又有最新技术,可作为高职高专基础课、专业基础课以及最新技术课的教材,也可供自考和学历文凭教育使用。

在人类社会进入新世纪以来,我国高等职业教育迅猛发展的格局已经形成。这就要求教育界的志士仁人奋发努力,以自己的心血和汗水去培养时代所需要的一代有理想、有道德、有知识、有能力的高素质、高水平的应用型专业人才。

陈建铎

(陕西省计算机教育学会副理事长
兼秘书长、西安石油大学教授)

^① 截止 2006 年 8 月,列入这套教材的选题已增至 23 种,其中 21 种已出版。

前 言

ASP.NET 是 Microsoft 公司推出的新一代 Web 应用开发技术,它是建立在公共语言运行库上的编程框架,可以用于在服务器上生成功能强大的 Web 应用程序。ASP.NET 虽然是 ASP 的下一代版本,但 ASP.NET 与 ASP 有着本质的区别。ASP.NET 完全基于模块和组件,提供了一种新的编程模型和结构,具有更好的可扩展性和可定制性,在数据访问方面也引入了许多开创性的技术,使用 ASP.NET 可以快速生成更加安全、更加稳定的企业级 Web 应用程序。

ASP.NET 可以无缝地与可视化开发工具 Visual Studio .NET 一起工作,这不仅使得 Web 开发更加方便,而且还能提供这些工具必须提供的所有优点。Visual Studio .NET 是 Microsoft 公司最新推出的世界级软件开发平台,在其可视化的集成开发环境中提供了大量的工具箱和设计器,可以用于创建 ASP.NET Web 应用程序,在开发过程中可以方便地进行即见即所得的编辑、拖放服务器控件、完全集成的调试和自动部署。

本书共分为 9 章,详细讲述了使用 Visual Studio .NET 开发 ASP.NET Web 应用程序的实用技术和设计技巧。第 1 章介绍 ASP.NET 基础知识,包括 ASP.NET 基本概念、构建 ASP.NET 运行环境以及使用 Visual Studio .NET 集成开发环境;第 2 章讲述 Visual Basic .NET 编程语言,包括基本语言元素、条件语句、循环语句、数组、过程、面向对象编程以及异常处理;第 3 章讲述 HTML 服务器控件应用,包括 HTML 服务器控件的共有特征和各种 HTML 服务器控件的使用方法;第 4 章讲述 Web 服务器控件应用,包括 Web 服务器控件的共有特征和各种 Web 服务器控件的使用方法;第 5 章首先讲述如何在 Web 窗体页上使用验证控件,然后介绍如何创建和应用 Web 用户控件;第 6 章讲述 ASP.NET 内置对象的应用,包括 Page 对象、Response 对象、Request 对象、Server 对象、Session 对象和 Application 对象的各种应用;第 7 章讲述 ADO.NET 数据库编程,包括 ADO.NET 结构概述、创建数据连接、执行 SQL 语句以及创建和使用数据集;第 8 章讲述数据列表控件的应用,包括 Repeater、DataList 以及 DataGrid 控件的使用方法;作为前面各章知识的综合应用,在第 9 章讲述了一个 ASP.NET Web 应用程序设计实例,即以 Visual Studio .NET 作为开发工具使用 ASP.NET 设计一个网络论坛。

为了帮助读者快速掌握用 Visual Studio .NET 可视化开发工具创建 ASP.NET Web 应用程序的编程方法和技巧,本书提供了丰富的操作实例,深入浅出、循序渐进地引导读者学习和掌握 ASP.NET Web 应用开发的技术要点,并辅以上机操作的屏幕画面,使读者有一种亲临其境的感觉。每章后面均附有习题,可供读者自我测试之用。

本书中的所有实例均基于 .NET Framework SDK v1.1 设计,所用的开发环境为 Visual Studio .NET 2003,所用的后台数据库主要是 SQL Server 2000。

本书中所用到的一些人名、通信地址均为虚构,如有雷同,实属巧合。

本书由赵增敏、朱粹丹、赵朱曦共同编著。参加本书编写、文字录入和程序测试的还有郭宏、李菲、朱永天、王庆建、李慧敏、张岩、孙宇鹏、王永列、李娴、刘旭、王静、李伟伟、李小洁、郭改文、赵明星、李自晓、谢飞、盛兴文、杨可伟等。

由于作者水平所限,书中疏漏和错误之处在所难免,欢迎广大读者提出宝贵意见。

编 者

2006 年 2 月

目 录

前 言

第 1 章 ASP.NET 使用基础

1.1 ASP.NET 概述	(1)
1.1.1 什么是 .NET	(1)
1.1.2 ASP.NET 的工作原理	(2)
1.1.3 ASP.NET 的主要优点	(3)
1.2 构建 ASP.NET 运行环境	(4)
1.2.1 ASP.NET 平台要求	(4)
1.2.2 安装 IIS 服务器	(5)
1.2.3 创建和配置 Web 站点	(6)
1.2.4 创建和配置虚拟目录	(6)
1.2.5 安装 .NET Framework	(8)
1.3 Visual Studio .NET 开发工	(9)
1.3.1 安装 Visual Studio .NET	(9)
1.3.2 认识 Visual Studio .NET 集成 开发环境	(11)
1.3.3 在 Visual Studio .NET 中创建 ASP.NET Web 应用程序	(14)
习题 1	(18)

第 2 章 Visual Basic .NET 编程语言

2.1 基本语言元素	(19)
2.1.1 数据类型	(19)
2.1.2 常量	(20)
2.1.3 变量	(21)
2.1.4 运算符	(21)
2.1.5 基本语句	(25)
2.2 条件语句	(26)
2.2.1 If 语句	(26)
2.2.2 Select Case 语句	(29)
2.3 循环语句	(33)
2.3.1 While...End While 语句	(33)
2.3.2 Do...Loop 语句	(34)
2.3.3 For...Next 语句	(36)
2.3.4 For Each...Next 语句	(38)

2.4 数组	(40)
2.4.1 声明数组	(40)
2.4.2 调整数组大小	(42)
2.5 过程	(44)
2.5.1 Sub 过程	(44)
2.5.2 Function 过程	(47)
2.5.3 常用内部函数	(49)
2.6 面向对象编程	(53)
2.6.1 类与对象	(53)
2.6.2 命名空间	(59)
2.6.3 访问控制	(60)
2.6.4 重载方法	(60)
2.6.5 继承与重写	(63)
2.7 异常处理	(65)
2.7.1 Try...Catch...Finally 语句	(65)
2.7.2 Exception 类	(66)
习题 2	(68)

第 3 章 HTML 服务器控件应用

3.1 HTML 服务器控件概述	(69)
3.1.1 ASP.NET 服务器控件层次结构	(69)
3.1.2 HTML 服务器控件及其功能	(70)
3.1.3 HTML 服务器控件层次结构	(70)
3.1.4 添加 HTML 服务器控件	(72)
3.1.5 HTML 服务器控件的通用属性	(73)
3.2 HTML 服务器控件详解	(75)
3.2.1 HtmlGenericControl 控件	(75)
3.2.2 HtmlForm 控件	(77)
3.2.3 HtmlInputButton 控件	(77)
3.2.4 HtmlInputText 控件	(79)

3.2.5	HtmlInputRadioButton 控件	控件	(133)
			(81)
3.2.6	HtmlInputCheckBox 控件	4.2.16 Panel 控件	(136)
			(82)
3.2.7	HtmlInputImage 控件	4.2.17 Xml 控件	(138)
			(84)
3.2.8	HtmlInputFile 控件	习题 4	(140)
			(86)
3.2.9	HtmlInputHidden 控件	第 5 章 验证控件与用户控件	
			(88)
3.2.10	HtmlAnchor 控件	5.1 验证控件	(142)
			(89)
3.2.11	HtmlButton 控件	5.1.1 验证控件概述	(142)
			(91)
3.2.12	HtmlSelect 控件	5.1.2 RequiredFieldValidator 控件	
			(92)
3.2.13	HtmlTextArea 控件		(145)
			(95)
3.2.14	HtmlImage 控件	5.1.3 CompareValidator 控件	(146)
			(97)
3.2.15	HtmlTable、HtmlTableRow 和	5.1.4 RangeValidator 控件	(149)
	HtmlTableCell 控件	5.1.5 RegularExpressionValidator 控件	
			(98)
习题 3			(151)
			(102)
第 4 章 Web 服务器控件应用			
4.1	Web 服务器控件概述	5.1.6 CustomValidator 控件	(154)
			(104)
4.1.1	Web 服务器控件层次结构	5.1.7 ValidationSummary 控件	(156)
			(104)
4.1.2	Web 服务器控件的类型	5.2 用户控件	(158)
			(105)
4.1.3	Web 服务器控件的通用属性	5.2.1 用户控件概述	(158)
			(107)
4.1.4	通过编程设置 Web 服务器控	5.2.2 创建用户控件	(158)
	件属性	5.2.3 添加用户控件	(159)
			(160)
		5.2.4 动态加载用户控件	(160)
		5.2.5 公开构成控件的属性	(161)
		5.2.6 自定义用户控件的属性	(163)
		习题 5	(165)
			(107)
4.2	Web 服务器控件详解	第 6 章 ASP.NET 内置对象应用	
			(109)
4.2.1	Label 控件	6.1 Page 对象	(166)
			(109)
4.2.2	Literal 控件	6.1.1 Web 窗体代码模型	(166)
			(110)
4.2.3	HyperLink 控件	6.1.2 Page 对象的属性	(168)
			(111)
4.2.4	Image 控件	6.1.3 Page 对象的方法	(169)
			(112)
4.2.5	Button 控件	6.1.4 Page 对象的事件	(169)
			(114)
4.2.6	LinkButton 控件	6.1.5 Page 对象应用示例	(170)
			(116)
4.2.7	ImageButton 控件	6.2 Response 对象	(172)
			(117)
4.2.8	TextBox 控件	6.2.1 Response 对象的主要属性和方法	
			(119)
4.2.9	CheckBox 控件		(172)
			(121)
4.2.10	CheckBoxList 控件	6.2.2 输出信息	(173)
			(122)
4.2.11	RadioButton 控件	6.2.3 输出文件	(175)
			(125)
4.2.12	RadioButtonList 控件	6.2.4 缓冲处理	(176)
			(127)
4.2.13	ListBox 控件	6.2.5 重定向网址	(177)
			(129)
4.2.14	DropDownList 控件	6.3 Request 对象	(178)
			(131)
4.2.15	Table、TableRow 和 TableCell		
			
			(133)

6.3.1 Request 对象的主要属性和方法	(178)	7.2.2 连接 Access 数据库	(219)
6.3.2 读取窗体变量	(179)	7.2.3 连接 SQL Server 数据库	(221)
6.3.3 读取查询字符串变量	(181)	7.2.4 在服务器资源管理器中创建数据库连接	(224)
6.3.4 读取服务器变量	(184)	7.3 执行 SQL 语句	(226)
6.3.5 读取客户端浏览器功能信息	(186)	7.3.1 使用 Command 对象执行 SQL 语句	(226)
6.3.6 读取客户端 Cookie	(188)	7.3.2 使用 DataReader 对象读取数据	(227)
6.4 Server 对象	(191)	7.3.3 使用 Parameter 对象传递数据	(231)
6.4.1 Server 对象的主要属性和方法	(191)	7.3.4 使用 SELECT 语句查询记录	(234)
6.4.2 执行其他 ASP.NET 网页	(192)	7.3.5 使用 INSERT 语句添加记录	(238)
6.4.3 将流程控制转移到其他 ASP.NET 网页	(193)	7.3.6 使用 UPDATE 语句更新记录	(241)
6.4.4 将虚拟路径转换为物理文件路径	(194)	7.3.7 使用 DELETE 语句删除记录	(244)
6.4.5 编码与解码	(195)	7.3.8 使用 Command 对象调用存储过程	(247)
6.5 Session 对象	(196)	7.4 创建和使用数据集	(250)
6.5.1 Session 对象概述	(196)	7.4.1 使用 DataAdapter 对象执行 SQL 语句	(251)
6.5.2 保存会话信息	(197)	7.4.2 创建和使用 DataSet 对象	(255)
6.5.3 判断是否创建了新的会话	(199)	7.4.3 创建和使用 DataTable 对象	(258)
6.5.4 设置会话的有效期限	(200)	7.4.4 创建和使用 DataRow 对象	(264)
6.5.5 强制结束会话	(202)	7.4.5 创建和使用 DataColumn 对象	(267)
6.5.6 处理会话事件	(204)	7.4.6 创建和使用 DataView 对象	(273)
6.6 Application 对象	(204)	习题 7	(275)
6.6.1 Application 对象概述	(205)		
6.6.2 保存应用程序状态	(206)		
6.6.3 处理应用程序事件	(207)		
6.6.4 Global.asax 文件	(208)		
习题 6	(210)		
第 7 章 ADO.NET 数据库编程		第 8 章 数据列表控件应用	
7.1 ADO.NET 概述	(212)	8.1 Repeater 控件应用	(277)
7.1.1 ADO.NET 结构	(212)	8.1.1 Repeater 控件概述	(277)
7.1.2 .NET Framework 数据提供程序	(213)	8.1.2 使用 Repeater 控件显示数据	(279)
7.2 创建数据连接	(216)		
7.2.1 Connection 对象概述	(216)		

8.1.3 设置 Repeater 控件的模板 ... (282)	
8.1.4 实现 Repeater 控件的分页显示 ... (284)	
8.2 DataList 控件应用 ... (288)	
8.2.1 DataList 控件概述 ... (288)	
8.2.2 使用 DataList 控件显示数据 ... (292)	
8.2.3 使用 DataList 控件显示选定项信息 ... (295)	
8.2.4 实现 DataList 控件的分页显示功能 ... (298)	
8.2.5 使用 DataList 控件编辑数据 ... (302)	
8.3 DataGrid 控件应用 ... (307)	
8.3.1 DataGrid 控件概述 ... (307)	
8.3.2 DataGrid 控件中的列 ... (311)	
8.3.3 使用 DataGrid 控件分页显示数据 ... (313)	
8.3.4 使用 DataGrid 控件实现数据排序 ... (316)	
8.3.5 在 DataGrid 控件中动态隐藏/显示某列 ... (319)	
8.3.6 使用 DataGrid 控件实现主详细页 ... (321)	
8.3.7 使用 DataGrid 控件更新记录 ... (326)	
8.3.8 使用 DataGrid 控件删除记录 ... (331)	
习题 8 ... (336)	
	第 9 章 网络论坛设计实例
	9.1 项目设计 ... (337)
	9.1.1 系统功能分析 ... (337)
	9.1.2 数据库设计与实现 ... (338)
	9.1.3 项目模块组成 ... (340)
	9.2 Web 窗体页设计 ... (341)
	9.2.1 准备工作 ... (341)
	9.2.2 用 Web 用户控件创建导航条 ... (342)
	9.2.3 设计会员注册页 ... (343)
	9.2.4 设计出错信息显示页 ... (345)
	9.2.5 设计会员登录页 ... (347)
	9.2.6 设计会员资料修改页 ... (348)
	9.2.7 设计密码查询页 ... (351)
	9.2.8 设计会员注销页 ... (354)
	9.2.9 设计会员管理页 ... (354)
	9.2.10 设计论坛首页 ... (357)
	9.2.11 设计新帖发表页 ... (359)
	9.2.12 设计帖子回复页 ... (361)
	9.2.13 设计帖子阅读页 ... (363)
	9.2.14 设计帖子编辑页 ... (370)
	9.2.15 设计帖子删除页 ... (373)
	9.2.16 设计作者信息查看页 ... (374)
	9.3 项目部署 ... (376)
	9.3.1 创建部署项目 ... (376)
	9.3.2 部署论坛 ... (378)
	9.3.3 卸载论坛 ... (379)
	习题 9 ... (379)

第 1 章 ASP.NET 使用基础

ASP.NET 是 Microsoft 公司推出的新一代 Web 应用开发技术,它是建立在公共语言运行库上的编程框架,可以用于在服务器上生成功能强大的 Web 应用程序。ASP.NET 虽然是 ASP 的下一代版本,但 ASP.NET 与 ASP 在许多方面有着本质的区别。本章讲述使用 ASP.NET 开发技术所需的一些基础知识,主要包括 ASP.NET 概述、构建 ASP.NET 平台以及 Visual Studio .NET 集成开发环境介绍等。

1.1 ASP.NET 概述

从字面上看,术语 ASP.NET 包含由 ASP 和 .NET 两个部分。其中 ASP 是 Active Server Page 的简称,意即活动服务器页面;.NET 是指 Microsoft .NET 服务平台。要真正理解 ASP.NET 的工作机制,首先需要了解什么是 .NET,在这个基础上还要搞清楚 ASP.NET 有哪些新特点和主要优点。

1.1.1 什么是 .NET

用 Microsoft 自己的话来说,所谓 .NET 就是 Microsoft XML Web 服务平台。XML Web 服务允许应用程序通过 Internet 进行通讯和共享数据,而不管所采用的是哪种操作系统,也不管所采用的是哪种设备或编程语言。开发人员可以利用 Microsoft .NET 平台创建功能强大的 XML Web 服务并将这些服务集成在一起。对个人用户来说,.NET 带来的好处则是无缝的、吸引人的体验。

Microsoft .NET 的核心部分是 .NET 框架,即 Microsoft .NET Framework。.NET 框架是一个完整的 Windows 组成部分,允许构建和运行下一代软件应用程序和 Web 服务。.NET 框架包括这样一些技术:Web 服务、Web 应用程序(ASP.NET)、数据访问(ADO.NET)、客户应用程序(Windows 窗体)等等,如图 1.1 所示。

.NET 框架是一个安全、高性能和扩展性极佳的运行环境,它包括以下三个结构层次。

(1) 最上层是用户接口与应用程序接口。ASP.NET 和 Windows Forms 是 .NET 框架的两种主要界面技术。ASP.NET 用于创建服务器端的 Web 应用程序,包括 Web 窗体和 Web 服务两种模式;Windows Forms 用于创建基于 Windows 平台的视窗应用程序,也用于开发多层结构的分布式系统的本地界面。

(2) 中间层是 .NET 类库,提供了许多类和接口,例如 ADO.NET、XML、IO、网络、调试、安全、多线程等。.NET 框架通过命名空间来组成类库,命名空间与类库的关系犹如文件系统中文件夹与文件的关系一样。例如,System 命名空间是 .NET 框架中基本类型的根命名空

间,其中包含的基础类和基类可以用于定义常用值和引用数据类型、事件和事件处理程序、接口、属性和处理异常;System.Data 命名空间包含构成 ADO.NET 结构的类,可以生成用于数据访问的组件。

(3) 最底层是一个公共语言运行库(Common Language Runtime,CLR),这是 .NET 框架的运行环境,它除了负责执行程序外,还提供了内存管理、线程管理、安全管理、版本管理、异常管理、通用类系统(Common Type System,CTS)以及生命周期监控等核心服务。CLR 为多种编程语言提供了一种统一的运行环境。采用 CLR 支持的编程语言所编写的源代码经过编译之后,将生成一种称为微软中间语言(Microsoft Intermediate Language,MSIL)的语言,当程序运行时 CLR 将使用即时编译器(Just In Time Compiler)生成机器代码以便计算机执行。

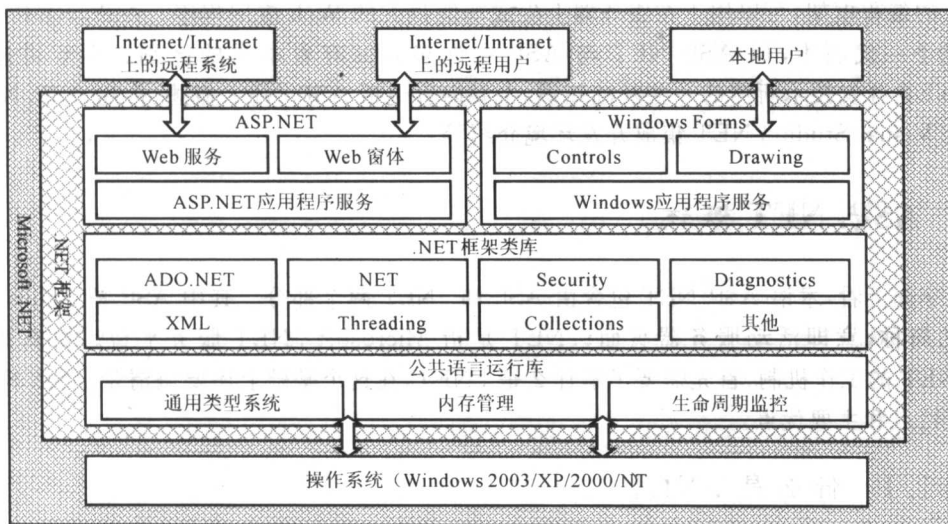


图 1.1 Microsoft .NET 结构

1.1.2 ASP.NET 的工作原理

ASP.NET 是 Microsoft .NET 框架的一个组成部分,通过它可以充分利用公共语言运行库的功能来创建、部署和运行 Web 应用程序和分布式应用程序。ASP.NET 是一个已编译的、基于 .NET 的环境,可以使用任何与 .NET 兼容的编程语言来开发 Web 应用程序,目前可以使用的语言主要包括 Visual Basic .NET、C# 和 JScript .NET。另外,任何 ASP.NET 应用程序都可以使用整个 .NET 框架。开发人员可以方便地获得这些技术的优点,其中包括托管的公共语言运行库环境、类型安全、继承等等。ASP.NET 应用程序包括 Web 窗体和 XML Web 服务两种模式。本书主要讲述如何使用 ASP.NET 开发技术设计基于 Web 窗体的应用程序。

使用 Web 窗体可以来创建可编程的动态网页,其文件扩展名是 .aspx,也称为 ASP.NET 动态网页,它们可以用作 Web 应用程序的用户界面。Web 窗体页在任何浏览器或客户端设备中向用户提供信息,并使用服务器端代码来实现应用程序逻辑。Web 窗体页输出几乎可以包含任何支持 HTTP 的语言,包括 HTML、XML、WML 和 ECMAScript(JScript、JavaScript)。

Web 窗体页的执行流程可以描述如下。

(1) 当用户在客户端 Web 浏览器的地址栏中输入一个 ASP.NET Web 窗体页的 URL 地址并按 Enter 键时,通过网络向 Web 服务器发出一个 HTTP 请求。

(2) Web 服务器收到该请求后,根据文件扩展名 .aspx 判断这是一个 ASP.NET Web 窗体页并定位该文件的位置,然后将其发送给 ASP.NET ISAPI 扩展 aspnet_isapi.dll 进行处理。

(3) aspnet_isapi.dll 将 ASP.NET 代码提交给公共语言运行库 CLR 进行处理。若先前未执行过这个程序,则由 CLR 进行编译并将执行地结果写入 HTML 流;若先前已执行过这个程序,则直接执行编译好的代码并生成 HTML 内容。

MSIL 是将 .NET 代码转化为机器语言的一个中间过程。它是一种介于高级语言和基于 Intel 的汇编语言的伪汇编语言。当用户编译一个 .NET 程序时,编译器将源代码翻译成一组可以有效地转换为本机代码且独立于 CPU 的指令。当执行这些指令时,实时(JIT)编译器将它们转化为 CPU 特定的代码。由于公共语言运行库支持多种实时编译器,因此同一段 MSIL 代码可以被不同的编译器实时编译并运行在不同的结构上。

Web 窗体页具有下列特点。

(1) 基于 Microsoft ASP.NET 技术。在该技术中,在服务器上运行的代码动态地生成到浏览器或客户端设备的 Web 页输出。

(2) 兼容所有浏览器或移动设备。Web 窗体页自动为样式、布局等功能呈现正确的、符合浏览器的 HTML。此外,还可以选择将 Web 窗体页设计为在特定浏览器(如 Microsoft Internet Explorer 5)上运行并利用多样式浏览器客户端的功能。

(3) 兼容 .NET 公共语言运行库所支持的任何语言,其中包括 Visual Basic .NET、Visual C#、Visual C++、JScript .NET 和 Visual J#。

(4) 基于 Microsoft .NET 框架生成。它提供了该框架的所有优点,包括托管环境、类型安全性和继承。

(5) 在 Visual Studio .NET 中通过强大的快速应用程序开发(RAD)工具受到支持,该工具用于对窗体进行设计和编程。

(6) 可以使用为 Web 开发提供 RAD 功能的控件进行扩展,从而可以快速地创建多样式的用户界面。

(7) 具有灵活性,因为可以向它们添加用户创建的控件和第三方控件。

1.1.3 ASP.NET 的主要优点

ASP.NET 的语法在很大程度上与 ASP 兼容,同时它还提供一种新的编程模型和结构,用于生成更安全、可伸缩和稳定的应用程序。与以前的 Web 开发模型相比,ASP.NET 具有许多重要的优点,主要包括以下几个方面。

(1) 增强的性能。ASP.NET 是在服务器上运行的编译好的公共语言运行库代码。与解释执行的 ASP 代码不同,ASP.NET 可以利用早期绑定、实时编译、本机优化和盒外缓存服务。这相当于在编写代码行之前便显著提高了性能。

(2) 世界级的工具支持。ASP.NET 框架补充了 Visual Studio .NET 集成开发环境中的大量工具箱和设计器,在这个开发环境中可以方便地进行即见即所得的编辑、拖放服务器控件和自动部署。

(3) 威力和灵活性。由于 ASP.NET 是基于公共语言运行库,因此 Web 应用程序开发人

员可以利用整个平台的威力和灵活性。.NET 框架类库、消息处理和数据访问解决方案都可以从 Web 无缝访问。ASP.NET 也与语言无关,所以可以选择最适合应用程序的语言,或跨多种语言分割应用程序。另外,公共语言运行库的交互性保证在迁移到 ASP.NET 时保留基于 COM 的开发中的现有投资。

(4) 简易性。ASP.NET 使执行常见任务变得容易,从简单的窗体提交和客户端身份验证到部署和站点配置。例如,ASP.NET 页框架可以生成将应用程序逻辑与表示代码清楚分开的用户界面,和在类似 Visual Basic 的简单窗体处理模型中处理事件。另外,公共语言运行库利用托管代码服务(如自动引用计数和垃圾回收)简化了开发。

(5) 可管理性。ASP.NET 采用基于文本的分层配置系统,简化了将设置应用于服务器环境和 Web 应用程序。由于配置信息是以纯文本形式存储的,因此可以在没有本地管理工具帮助的情况下应用新设置。只需要将必要的文件复制到服务器,就可以将 ASP.NET 框架应用程序部署到服务器。不需要重新启动服务器,即使是在部署或替换运行的编译代码时。

(6) 可缩放性和可用性。ASP.NET 在设计时考虑了可缩放性,增加了专门用于在聚集环境和多处理器环境中提高性能的功能。另外,进程受到 ASP.NET 运行库的密切监视和管理,以便当进程行为不正常(泄漏、死锁)时,可就地创建新进程,以帮助保持应用程序始终可以用于处理请求。

(7) 自定义性和扩展性。ASP.NET 随附了一个设计周到的结构,它使开发人员可以在适当的级别“插入”代码。实际上,可以用自己编写的自定义组件扩展或替换 ASP.NET 运行库的任何子组件。实现自定义身份验证或状态服务一直没有变得更容易。

(8) 安全性。借助内置的 Windows 身份验证和基于每个应用程序的配置,可以保证应用程序是安全的。

1.2 构建 ASP.NET 运行环境

要在 Web 服务器计算机上开发和运行 ASP.NET Web 应用程序,首先需要构建和配置 ASP.NET 的运行环境。

1.2.1 ASP.NET 平台要求

在 Windows 2000、Windows XP Professional 及 Windows Server 2003 系列产品上,客户端和服务端应用程序都支持 ASP.NET。要开发 ASP.NET Web 应用程序,还必须具备以下软件。

(1) 具有 Service Pack 2 的 Windows 2000 Server 或 Advanced Server、Windows XP Professional 或 64 位版本,或 Windows Server 2003 系列产品之一。

(2) Internet 信息服务 IIS 5.0 或更高版本。

(3) 数据访问组件 MDAC 2.7 或更高版本。

(4) .NET Framework。

在 Windows XP 和 Windows 2000 Server 上,ASP.NET 与 .NET Framework 一起安装,既可以单独安装,也可以作为 Visual Studio .NET 的一部分安装。要将 ASP.NET Web 应用程序部署到成品服务器,在安装 .NET Framework 之前,必须确保该服务器上已经安装并运

行了 Internet 信息服务 IIS。

如果在服务器上安装了 ASP.NET 和 .NET Framework,然后卸载并重新安装 IIS,则脚本映射将中断,此时 ASP.NET 将不再工作。一旦发生了这种情况,可以用 ASP.NET IIS 注册工具(Aspnet_regiis.exe)来修复 ASP.NET 脚本映射。

在 Windows Server 2003 系列产品上,ASP.NET 是作为操作系统上的角色安装的。要将 ASP.NET Web 应用程序部署到成品服务器,在分布该应用程序之前,必须在成品服务器上启用 ASP.NET 和 IIS 角色。在 Windows Server 2003 系列产品中,ASP.NET 与 .NET Framework 1.1 版作为操作系统的组成部分一起安装。只需要通过控制面板将它添加为新的程序,或者使用“配置您的服务器向导”启用它。在运行 Windows Server 2003 的服务器上,当使用“配置您的服务器向导”或“添加/删除程序”对话框安装 ASP.NET 时,ASP.NET 在“IIS 管理器”中自动启用。但是,如果通过 Web 下载安装 ASP.NET 或作为应用程序下载(如 Visual Studio .NET)的一部分安装,则必须手动启用 ASP.NET。

当在运行 Windows XP Professional 或 Windows 2000 Server 的计算机上安装 Visual Studio .NET 时,.NET Framework 和 ASP.NET 也会自动安装。如果要单独安装 ASP.NET 和 .NET Framework,可以从网上下载它们并安装到服务器上。

1.2.2 安装 IIS 服务器

在 Windows 2000 Server、Windows XP Professional 和 Windows 2003 Server 平台上,IIS 的版本分别为 5.0、5.1 和 6.0。下面是以安装和配置 IIS 5.0 为例来加以说明。

(1) 在 Windows 2000 中,选择“开始”>“设置”>“控制面板”命令。

(2) 在“控制面板”窗口中,双击“添加或删除程序”图标。

(3) 在“添加/删除程序”窗口中,单击“添加/删除 Windows 组件”。

(4) 在“Windows 组件向导”对话框中,选中“Internet 信息服务(IIS)”组件,单击“下一步”按钮,然后根据向导的提示进行操作,直至完成安装。

完成安装后,在 IE 浏览器地址栏中输入 URL 地址 <http://localhost> 进行测试。如果已经正确地安装了 IIS 5.0,并且在计算机上正确地配置了 Web 站点,则会在 IE 浏览器窗口中看到如图 1.2 所示的内容。

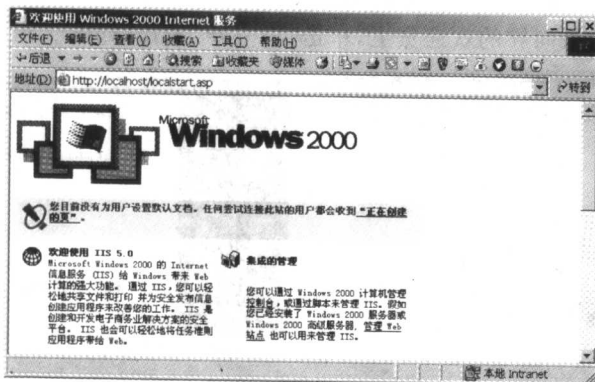


图 1.2 在 IE 浏览器中测试 IIS 5.0

1.2.3 创建和配置 Web 站点

Web 站点用来接收和响应 Web 客户端的请求。每个 Web 站点都具有惟一的标识,该标识由 IP 地址、TCP 端口号和主机头名三个部分组成。通过更改其中的一个标识,就可以在一台计算机上维护多个 Web 站点。在 IIS 中创建的 Web 站点也称为虚拟主机或虚拟服务器,它是驻留在 Web 服务器上的虚拟计算机。在运行 Windows 2000 Server 的计算机上,通过 IIS 可以维护多个 Web 站点,从而可以在同一台服务器计算机上创建多个虚拟主机。

当在 Windows 2000 Server 中安装 IIS 5.0 时,将会创建一个默认的 Web 站点,可以直接在这个默认站点上发布信息。也可以根据需要在服务器计算机上添加更多的 Web 站点。在 Windows 2000 中,可以利用 Internet 信息服务管理单元来创建新的 Web 站点。

在创建 Web 站点过程中,已经对该站点的主目录、IP 地址、TCP 端口号以及主机头名等属性进行了配置。若要对已有站点的配置进行修改,请在 Internet 信息服务管理单元窗口中右键单击该站点,然后选择“属性”,以打开站点属性对话框。在该对话框中,可以对 Web 站点的标识、主目录、默认文档等属性进行设置。

1.2.4 创建和配置虚拟目录

在安装 IIS 的过程中时,安装向导会自动创建一个默认的 Web 站点并将其主目录设置为 \Inetpub\Wwwroot,也可以通过根据需要更改主目录的位置。如果要从站点主目录之外的文件夹发布信息,则应当在 Web 站点上创建虚拟目录。

虚拟目录是指在物理上未包含在站点主目录下的特定文件夹,但客户浏览器却将其视为包含在主目录下的目录。虚拟目录与一个实际物理目录相对应。虚拟目录具有别名,这个别名映射到 Web 内容所在实际物理目录,Web 浏览器通过别名来访问此目录。

在 Windows 2000 Server 中,可以使用 Internet 信息服务管理单元在 Web 站点中创建虚拟目录,操作步骤如下。

(1) 选择“开始”>“程序”>“管理工具”>“Internet 服务管理器”命令。

(2) 在 Internet 信息服务管理单元中,用鼠标右键单击要添加虚拟目录的 Web 站点,然后从弹出菜单中选择“新建”>“虚拟目录”命令,如图 1.3 所示。

(3) 在新建虚拟目录向导的欢迎页上,单击“下一步”按钮,如图 1.4 所示。



图 1.3 在 Web 站点中新建虚拟目录

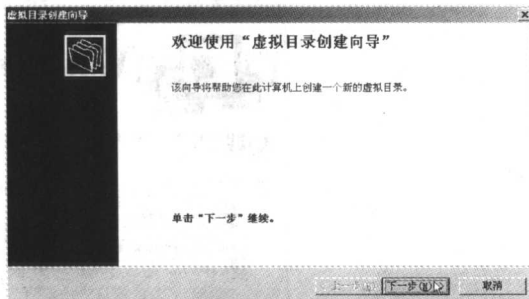


图 1.4 新建虚拟目录向导的欢迎页