

Broadview®
www.broadview.com.cn

ASP.NET 2.0 网络编程

张英男 高喆 林丁报 靳海江 编著

学习
笔记

编程

网络

ASP

ASP



电子工业出版社

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

TP393.092/1016D

2008

ASP.NET 2.0 网络编程学习笔记

张英男 高喆 林丁报 靳海江 编著

ASP.NET 2.0
网络编程学习笔记

网络编程

ASP

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书由浅入深地讲解了ASP.NET 2.0开发技术,并且对ASP.NET 2.0新的核心技术进行了精辟的讲解,全书内容分为5个部分,共21章。第一部分介绍基础知识,包括Web和ASP.NET工作原理, C# 2.0语言基础,开发环境的搭建, ASP.NET 2.0页面的结构原理和ASP.NET 2.0常用的内置对象;第二部分介绍ASP.NET 2.0的常用技术,主要包括服务器控件,自定义控件,数据源控件,数据绑定控件和ADO.NET数据库访问技术;第三部分介绍ASP.NET 2.0核心技术,包括母版页、主题、皮肤、个性化设置和Web Part技术;第四部分介绍ASP.NET安全和缓存原理,以及程序员需要了解的Web框架技术;第五部分精选了几个能囊括本书介绍的所有理论的实例,能使读者彻底精通ASP.NET 2.0开发技术。

本书内容全面,不但适合于没有任何基础而想直接学习ASP.NET 2.0技术的初学者,而且也适合于有一定ASP.NET 1.x基础而想进一步了解ASP.NET 2.0的中级读者。

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

ASP.NET 2.0网络编程学习笔记 / 张英男等编著. —北京: 电子工业出版社, 2008.4
ISBN 978-7-121-05502-7

I. A… II. 张… III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆CIP数据核字(2007)第185048号

责任编辑: 朱沐红

印 刷: 北京天宇星印刷厂

装 订: 三河市鹏成印业有限公司

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

开 本: 860×1092 1/16 印张: 37.25 字数: 950千字 彩插: 3

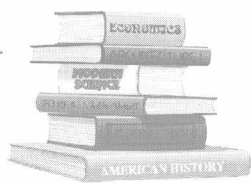
印 次: 2008年4月第1次印刷

印 数: 5000册 定价: 65.00元 (赠光盘1张)

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至zlts@phei.com.cn, 盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线: (010) 88258888。



本书独一无二的优势



本书由具有多年开发经验的程序员执笔写作，笔者在Web领域具有多年的开发研究经验，并且具有多年的培训讲解经验，尤其熟识ASP.NET 2.0 Web开发理论知识体系，笔者凭着娴熟的笔法和渊博的理论知识，将ASP.NET 2.0开发技术展现得淋漓尽致，能使读者能够很快地进入实际开发角色。本书和市场上其他类似书籍相比，具有下面与众不同的特色：

★形象

使抽象变形象，让高深的编程理论“赤裸裸”地呈现在你的面前，使你轻而易举地领悟编程奥秘。这是本书最大的特色。

★趣味

本书用拉链做素材，拉开学习的帷幕，用情趣似的学习笔记形式带你走进轻松的学习课堂，并且在书中让人感觉到烦躁的地方搭配以相关的对应图片，让你享受到学习中的乐趣，探索中的喜悦，胜利后的兴奋。

★实用

本书的实用性强，以经验为后盾，以实践为导向，以实用为目标，深入浅出地讲解ASP.NET 2.0开发中的种种问题。特别是，在讲解时非常注重实践与理论的形象结合，为了能使读者通过读懂源代码来掌握ASP.NET 2.0技术，在实例举证时还给源代码绘了大量的流程图。本书的所有源代码都已调试通过，并且放在了本书所附带的光盘中，读者拿来即可使用

★全面

本书内容全面，从基本的语法入手，以恰当的实例为导向，由浅入深地讲解ASP.NET 2.0技术的基本理论知识，所讲解的内容几乎囊括了ASP.NET 2.0技术的所有知识点，而且还特地安排了一章来深度剖析Web三层架构技术，这是市场上其他ASP.NET 2.0书籍没有涉及到的。



前言 Foreword

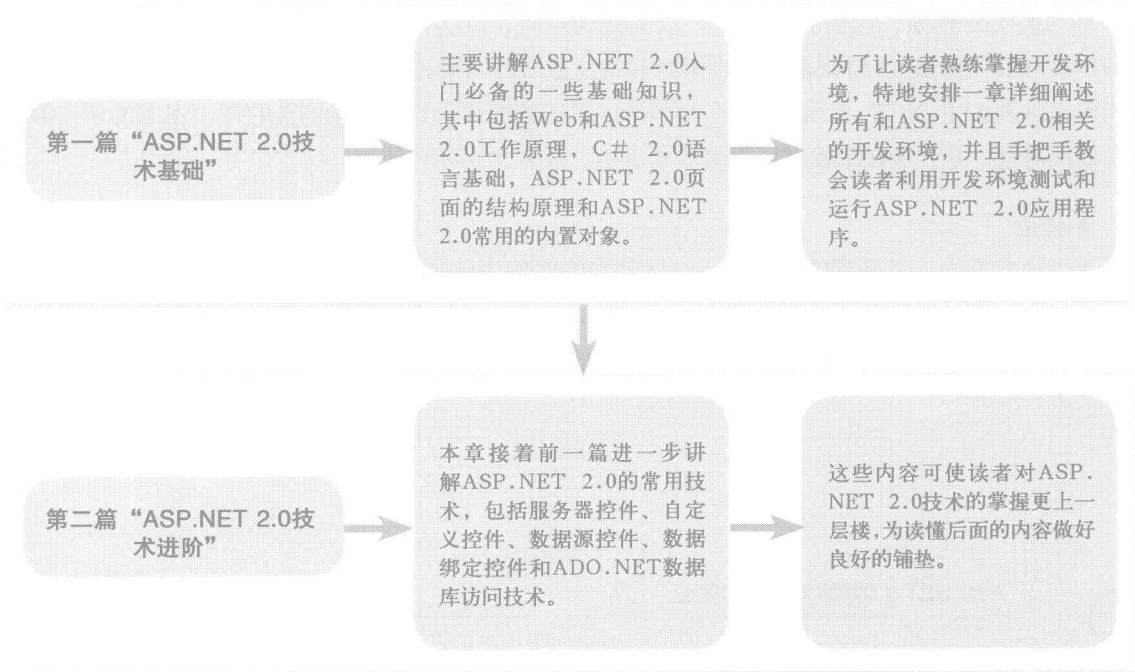


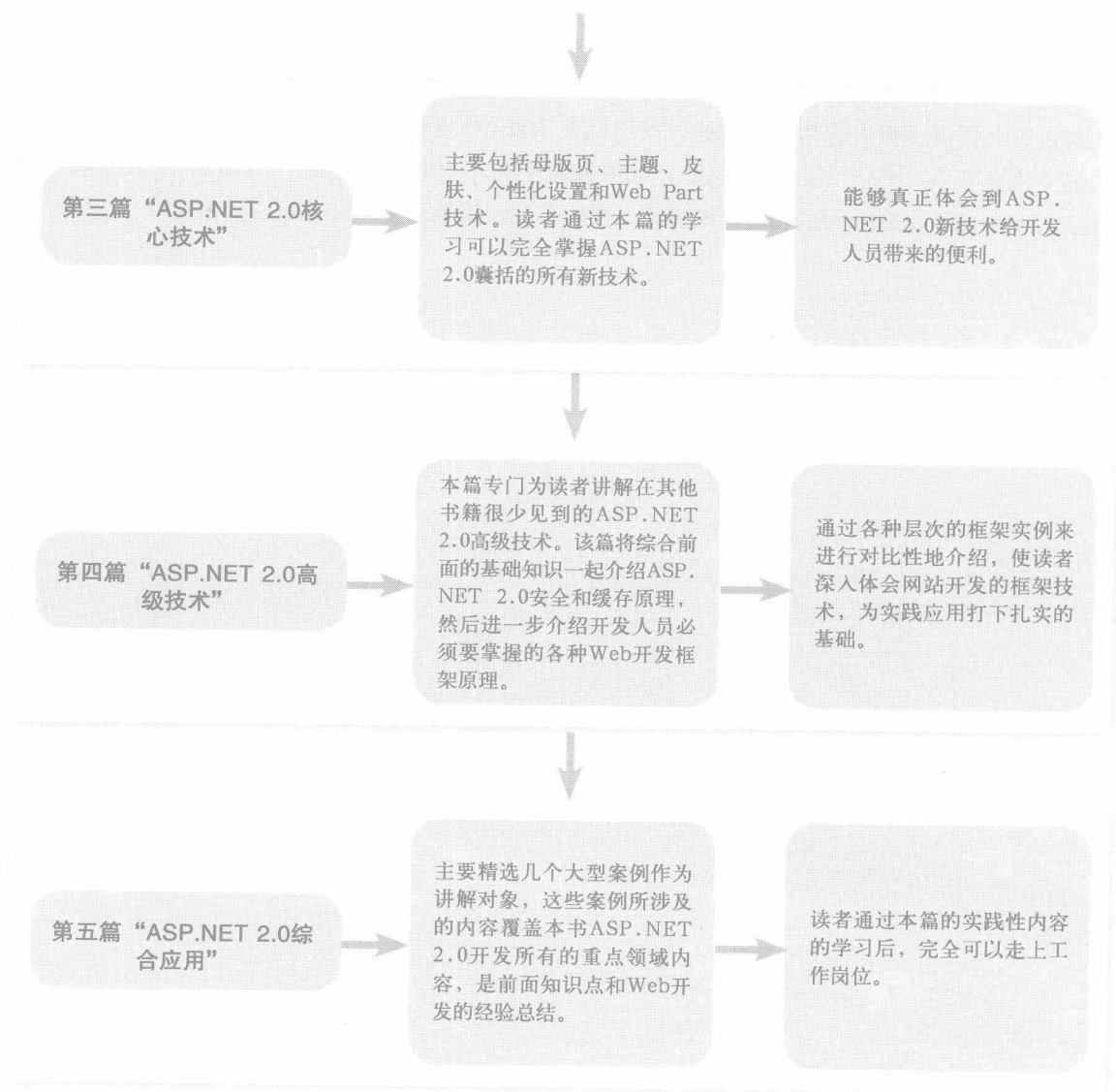
计算机软件技术的日新月异推动了社会信息化的进程，这也使得开发人员不得不学习新的技术以适应这种局面。在Web应用领域，同样历经几代变迁，从最初的Mosaic到Netscape，再到目前的Web Service时代，其中涌现出很多优秀的Web开发技术，CGI、ASP、JSP、PHP都曾经是潮流。但ASP.NET向世界展现了一种更为领先的技术，这正是它推出不久就深受开发人员青睐的原因所在。应业界广大开发人员的意见和建议，微软公司于2005年底正式推出了新的ASP.NET 2.0版本，和原来的ASP.NET 1.x版本相比，在方便性和性能方面都作了很大的提升。

ASP.NET 2.0新技术大大提高了编程人员的开发效率，可以说是一项革命性的创新，特别是在设计思想、编程模式和开发效率方面给开发人员带来了前所未有的便利。为了帮助国内众多开发人员迅速有效地掌握ASP.NET 2.0技术，本书从介绍最基本的ASP.NET基础知识开始，以实践为导向，带领读者慢慢步入ASP.NET 2.0新技术的殿堂，了解ASP.NET 2.0技术的奥秘。

本书内容结构

本书以ASP.NET 2.0的版本作为讲解对象，从基本的语法和规范入手，以恰当的实例为导向，从浅入深地介绍ASP.NET 2.0开发的基本理论知识及Web应用系统开发的实践经验。本书一共21章分为5篇。分别如下：





本书读者对象

本书在内容安排上由浅入深, 写作上层层剥洋葱式的分解, 实例举证, 非常适合于ASP.NET 2.0技术的初学者, 同时也适合具有一定ASP.NET 2.0基础, 欲对ASP.NET 2.0开发技术进一步了解和掌握的中级学者。如果您是以下类型的学者, 此书会带领您迅速进入ASP.NET 2.0开发领域:

- 1 没有任何ASP.NET基础, 欲直接学习ASP.NET 2.0的初学者。
- 2 有其他Web编程语言如ASP、JSP开发经验, 欲快速转向ASP.NET 2.0开发的程序员。
- 3 有一定的ASP.NET基础, 欲快速学会ASP.NET 2.0新技术的读者。
- 4 有一定ASP.NET开发基础, 需要加深对ASP.NET 2.0技术核心进一步了解和掌握的程序员。

致 谢

本书由张英男，高喆，林丁报和靳海江编写，在编写过程中，温才燚，李振捷，王大平，宗杰，刘冉等人提供了很大的帮助，在此，对他们表示衷心的感谢。由于时间仓促，加上作者水平有限，书中难免存在一些不足和错误之处，如果读者使用本书时遇到问题，可以到本书的服务网站 <http://www.cx-pub.com> 的相关论坛提问，我们会及时给您回复。

目录 Content



第一篇 ASP.NET 2.0技术基础

第1章 ASP.NET技术概述 01

- 1.1 Web工作原理 02
 - 1.1.1 Web服务器 02
 - 1.1.2 Web客户 03
- 1.2 HTML与HTTP 04
 - 1.2.1 HTML 标记语言 04
 - 1.2.2 CSS样式表 05
 - 1.2.3 脚本语言 06
 - 1.2.4 什么是HTTP协议 07
 - 1.2.5 何谓URL 08
 - 1.2.6 HTML Form与Web请求 09
 - 1.2.7 GET与POST请求 10
- 1.3 动态页面 11
 - 1.3.1 静态页面和动态页面 11
 - 1.3.2 ASP.NET扮演的角色 12
- 1.4 小结 14

第2章 搭建开发环境 15

- 2.1 IIS安装和配置 16
- 2.2 Visual Studio .NET2005 集成开发环境 21
 - 2.2.1 安装 Visual Studio 2005 21
 - 2.2.2 常用功能窗口介绍 23
 - 2.2.3 配置设置 27
 - 2.2.4 类设计器 29
 - 2.2.5 多种测试功能 30
- 2.3 SQL Server 2005安装与配置 31
 - 2.3.1 安装、配置SQL Server 2005 31
 - 2.3.2 测试配置是否成功 32
- 2.4 熟练使用开发环境 33
 - 2.4.1 创建Web站点 33
 - 2.4.2 编写ASP.NET 2.0 应用程序 34
 - 2.4.3 测试SQL Server数据库连接 35
 - 2.4.4 编译和运行应用程序 36
 - 2.4.5 部署应用程序 37
- 2.5 小结 38

第3章 C# 2.0语言基础 39

- 3.1 NET及C#语言概述 40

- 3.2 数据类型 43
 - 3.2.1 值类型 43
 - 3.2.2 引用类型 44
 - 3.2.3 装箱和拆箱 45
 - 3.2.4 命名空间 46
- 3.3 流程控制 48
 - 3.3.1 条件语句 48
 - 3.3.2 循环语句 50
 - 3.3.3 异常处理 52
- 3.4 面向对象编程简介 53
 - 3.4.1 概述 53
 - 3.4.2 类 54
 - 3.4.3 字段 57
 - 3.4.4 属性 57
 - 3.4.5 方法 58
 - 3.4.6 委托与事件 59
 - 3.4.7 抽象类 62
 - 3.4.8 接口 63
 - 3.4.9 多态 64
 - 3.4.10 集合 65
- 3.5 小结 66

第4章 ASP.NET 页面的结构 67

- 4.1 一个简单的.aspx文件 68
 - 4.1.1 一个简单的.aspx文件 68
 - 4.1.2 aspx文件中各个元素 69
 - 4.1.3 ASP.NET页面的布局 69
 - 4.1.4 ASP页面与ASP.NET页面的异同 70
- 4.2 页面指令 72
 - 4.2.1 Page指令 73
 - 4.2.2 Import指令 75
 - 4.2.3 Assembly指令 76
 - 4.2.4 Reference指令 77
 - 4.2.5 Register指令 79
 - 4.2.6 Implements指令 80
 - 4.2.7 其他指令 80
- 4.3 小结 80

第5章 ASP.NET内置对象及实例详解 81

- 5.1 内建对象的介绍 82
 - 5.1.1 Response对象 82
 - 5.1.2 Request对象 82

5.1.3 Application对象	83
5.1.4 Session对象	83
5.1.5 Server对象	84
5.1.6 Cache对象	85
5.1.7 Cookie对象	85
5.2 内建对象的使用实例	86
5.2.1 Response对象	86
5.2.2 Request对象	87
5.2.3 Application对象	90
5.2.4 Session对象	91
5.2.5 Server对象	92
5.2.6 Cookie对象	94
5.3 小结	95

第二篇 ASP.NET 2.0技术进阶

第6章 ASP.NET服务器控件 96

6.1 HTML服务器控件	97
6.2 Web服务器控件	101
6.2.1 简单控件	102
6.2.2 按钮控件	106
6.2.3 列表控件	110
6.2.4 数据绑定控件	114
6.2.5 验证控件	116
6.3 用户控件	120
6.3.1 用户控件基础	120
6.3.2 使用属性和事件	121
6.3.3 动态加载	123
6.3.4 片段缓存	124
6.4 ASP.NET 2.0新控件	125
6.4.1 HiddenField控件	126
6.4.2 FileUpload控件	127
6.4.3 ImageMap控件	128
6.4.4 MultiView、View控件	129
6.4.5 其他的新控件	130
6.5 小结	130

第7章 Web自定义控件 132

7.1 一个简单的ASP.NET控件	133
7.2 子控件	137
7.2.1 控件的子控件	137
7.2.2 建立分析器	139
7.3 复合控件	140
7.4 小结	143

第8章 数据源控件 144

8.1 数据源控件的原理与本质	145
-----------------	-----

8.2 SqlDataSource控件	146
---------------------	-----

8.2.1 SqlDataSource控件简介	146
8.2.2 使用SqlDataSource连接数据库	147
8.2.3 使用SqlDataSource连接多种数据源	156

8.3 AccessDataSource控件	158
------------------------	-----

8.3.1 AccessDataSource控件简介	158
8.3.2 使用AccessDataSource控件访问Access数据库	159

8.4 XmlDataSource控件	161
---------------------	-----

8.4.1 XmlDataSource控件简介	161
8.4.2 使用XmlDataSource控件显示RSS数据	162

8.5 SiteMapDataSource控件	165
-------------------------	-----

8.5.1 SiteMapDataSource控件简介	165
8.5.2 SiteMapDataSource控件与绑定控件的结合使用	165

8.6 ObjectDataSource控件	168
------------------------	-----

8.6.1 ObjectDataSource控件简介	168
8.6.2 使用ObjectDataSource绑定数据访问层	168
8.6.3 使用ObjectDataSource绑定业务逻辑层	176
8.6.4 使用ObjectDataSource绑定类型化数据集	180

8.7 小结	185
--------	-----

第9章 数据绑定控件 186

9.1 DropDownList控件	187
--------------------	-----

9.1.1 DropDownList控件简介	187
9.1.2 绑定到DropDownList控件	188

9.2 DataList控件	190
----------------	-----

9.2.1 DataList控件简介	190
9.2.2 绑定到DataList控件	193

9.3 GridView控件	196
----------------	-----

9.3.1 GridView控件对象模型	196
9.3.2 典型应用1—绑定到GridView控件	202
9.3.3 典型应用2—实现数据分页功能	203
9.3.4 典型应用3—实现编辑数据功能	205

9.4 DetailsView控件	206
-------------------	-----

9.4.1 DetailsView控件简介	206
9.4.2 使用DetailsView操作数据	206

9.5 FormView控件	210
----------------	-----

9.5.1 控件简介	210
9.5.2 使用FormView操作数据	211

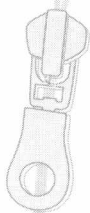
9.6 小结	216
--------	-----

第10章 ADO.NET 对象模型 217

10.1 ADO.NET简介	218
----------------	-----

10.1.1 ADO.NET的体系结构	218
10.1.2 托管提供程序的数据模型	225

10.1.3 SQL Server .NET和OLE DB .NET数据模型	228	13.1 个性化设置简介	285
10.1.4 .NET数据集	230	13.2 个性化用户配置	286
10.2 访问数据库	231	13.2.1 <profile>的详细配置	286
10.2.1 使用Connection对象连接数据库	232	13.2.2 使用API配置自定义用户属性	288
10.2.2 使用Command对象操作数据库	234	13.2.3 SQL Server数据库配置	290
10.2.3 使用DataReader对象查询数据库	237	13.3 实现个性化用户配置	294
10.2.4 Connection、Command和DataReader对象的 综合使用实例	240	13.3.1 为匿名用户实现简单数据类型的个性化用户 配置	294
10.3 存储过程	245	13.3.2 为注册用户实现复杂数据类型的个性化用户 配置	299
10.3.1 存储过程原理	245	13.4 小结	307
10.3.2 创建存储过程	246	第14章 用Web Part 创建门户	308
10.3.3 使用存储过程的实例详解	248	14.1 Web Part概述	309
10.4 小结	252	14.1.1 WebPart的基本控件	309
第三篇 ASP.NET2.0核心技术		14.1.2 配置WebPart环境	312
第11章 母版页	254	14.1.3 WebPart的生命周期	316
11.1 母版页概述	255	14.2 创建WebPart页面	317
11.2 创建母版页和内容页	256	14.2.1 使用WebPartZone控件创建一个简单的 WebPart页面	317
11.2.1 创建母版页	256	14.2.2 使用CatalogZone控件添加向页面中添加 WebPart	319
11.2.2 创建内容页	258	14.2.3 使用EditorZone控件编辑页面中WebPart	322
11.3 母版页的嵌套	259	14.2.4 管理WebPart页面	326
11.4 动态访问母版页	261	14.2.5 WebPart页面间的通讯	326
11.4.1 访问母版页上的成员	261	14.2.6 自定义WebPart	336
11.4.2 获取母版页上的控件的值	261	14.3 小结	337
11.5 母版页的应用范围	264	第四篇 ASP.NET2.0高级技术	
11.6 缓存母版页	265	第15章 ASP.NET缓存的使用	338
11.7 小结	266	15.1 ASP.NET缓存概述	339
第12章 主题和皮肤	267	15.2 页面输出缓存	340
12.1 主题概述	268	15.2.1 使用@OutputCache指令	341
12.1.1 组成元素	268	15.2.2 使用API	348
12.1.2 文件的存储和组织方式	269	15.2.3 页面输出缓存应用	350
12.1.3 全局主题和应用程序主题	270	15.3 页面部分缓存	352
12.2 创建主题	270	15.4 应用程序数据缓存	354
12.2.1 创建并使用皮肤文件	270	15.5 SQL 数据依赖缓存	356
12.2.2 为主题添加CSS文件	278	15.6 配置缓存	360
12.2.3 在主题中使用图片	280	15.7 小结	361
12.3 应用主题	281	第16章 ASP.NET的安全性	362
12.3.1 将主题应用于整个应用程序	281	16.1 ASP.NET2.0安全性概述	363
12.3.2 使用StyleSheetTheme为页面指定和禁用主题	282	16.2 窗体验证	364
12.4 小结	283	16.3 应用实例	366
第13章 个性化设置	284	16.4 高级设置	373



16.4.1 修改数据提供程序	373
16.4.2 控件调整	375
16.4.3 成员和角色编程	376
16.4.4 无cookie的窗体身份验证	376
16.4.5 一些预防措施	377
16.5 小结	378
第17章 ASP.NET网站开发架构	379
17.1 B/S架构设计理念	380
17.1.1 单层设计架构	380
17.1.2 二层设计架构	387
17.1.3 三层设计架构	392
17.1.4 使用Visual Studio 2005建立第一个三层架构 应用	395
17.1.5 三种设计架构的比较	409
17.2 三层架构进阶	410
17.2.1 理解三层架构——从一个小餐馆说起 ..	410
17.2.2 引入实体项目——在土豆炖牛肉盖饭中一 定能发现土豆和牛肉	412
17.2.3 跨数据库的实现——小餐馆新来了一位大 厨师傅	419
17.2.4 引入行为项目——让服务生知道大厨师傅的 手艺	425
17.2.5 三层架构完整的执行过程	431
17.3 N层架构体系	433
17.3.1 N层架构体系中的商业逻辑——三碗不过岗	433
17.3.2 N层架构体系与Web服务	436
17.4 小结	440
第18章 AJAX.NET和ATLAS简介	441
18.1 AJAX.NET概述	442
18.2 AJAX.NET常用技术	445
18.3 Atlas概述	453
18.3.1 什么是Atlas	453
18.3.2 如何对Atlas进行部署	454
18.3.3 如何在现有工程中添加Atlas	454
18.3.4 Atlas有哪些优点	456
18.4 Atlas常用技术	456
18.5 小结	463

第五篇 ASP.NET 2.0综合应用

第19章 构建个人网站系统	464
19.1 如何总揽全局（设计）	465
19.2 几个关键技术	466
19.2.1 如何组织数据	466

19.2.2 面向对象编程	468
19.2.3 使用服务器端控件	482
19.2.4 建立站点导航	491
19.2.5 定义全局参数	492
19.3 美化程序页面	494
19.3.1 母版页	494
19.3.2 主题和皮肤	495
19.4 用户管理	497
19.5 安装并使用	501
19.6 小结	504

第20章 小型企业网站系统

20.1 系统设计	506
20.1.1 系统流程	506
20.1.2 设计数据库表	506
20.1.3 设计功能模块	509
20.1.4 设计程序外观	509
20.2 系统实现	510
20.2.1 实现主要功能模块	510
20.2.2 实现其他辅助模块	530
20.2.3 设计站点导航文件	534
20.2.4 定义全局参数	534
20.2.5 设计母版页	536
20.2.6 设计主题和皮肤	538
20.3 系统扩展	539
20.4 小结	539

第21章 网上商城购物系统

21.1 需求分析	541
21.2 设计方案	541
21.2.1 系统流程	541
21.2.2 设计数据库表	543
21.2.3 设计数据层	550
21.2.4 设计应用层	555
21.2.5 设计表示层	555
21.3 解决方案	555
21.3.1 数据库	555
21.3.2 全局变量（Web.config文件）	556
21.3.3 母版页	559
21.3.4 主题和皮肤	561
21.3.5 对象类	562
21.3.6 页面代码	564
21.4 小结	584

第1章 ASP.NET 技术概述

拉开崭新的学习帷幕

一些读者可能对Web工作原理及ASP.NET技术有些许了解,但是也有部分读者可能原本是编写Windows应用程序的,对Web编程所需要的一些基本知识还不甚了解。通过本章的学习,读者将了解到Web的基本工作原理,进一步认识Web通讯的本质,从而为今后编写Web应用程序打下基础。同时,本章将向读者介绍ASP.NET相关的基本内容,使读者对微软的网页编程技术有个基本的认识,在后续章节里,将向读者详细介绍其详细内容。

本章内容包括:

- ★Web工作原理。
- ★HTML与HTTP。
- ★动态页面。



1.1 Web工作原理

万维网 (World Wide Web, 简称WWW) 是一种基于超文本方式工作的信息服务系统。它作为一个能够处理文字、图像、音频、视频等多媒体信息的综合系统, 提供了大量内容, 丰富了信息资源。这些信息资源以Web页面的形式, 分门别类地存放在各个Internet服务器上。用户可以根据个人的兴趣爱好, 通过Web浏览器选择所需的内容。通过万维网, 全世界的网民可以在世界各地浏览全球的资源。如图1.1所示:

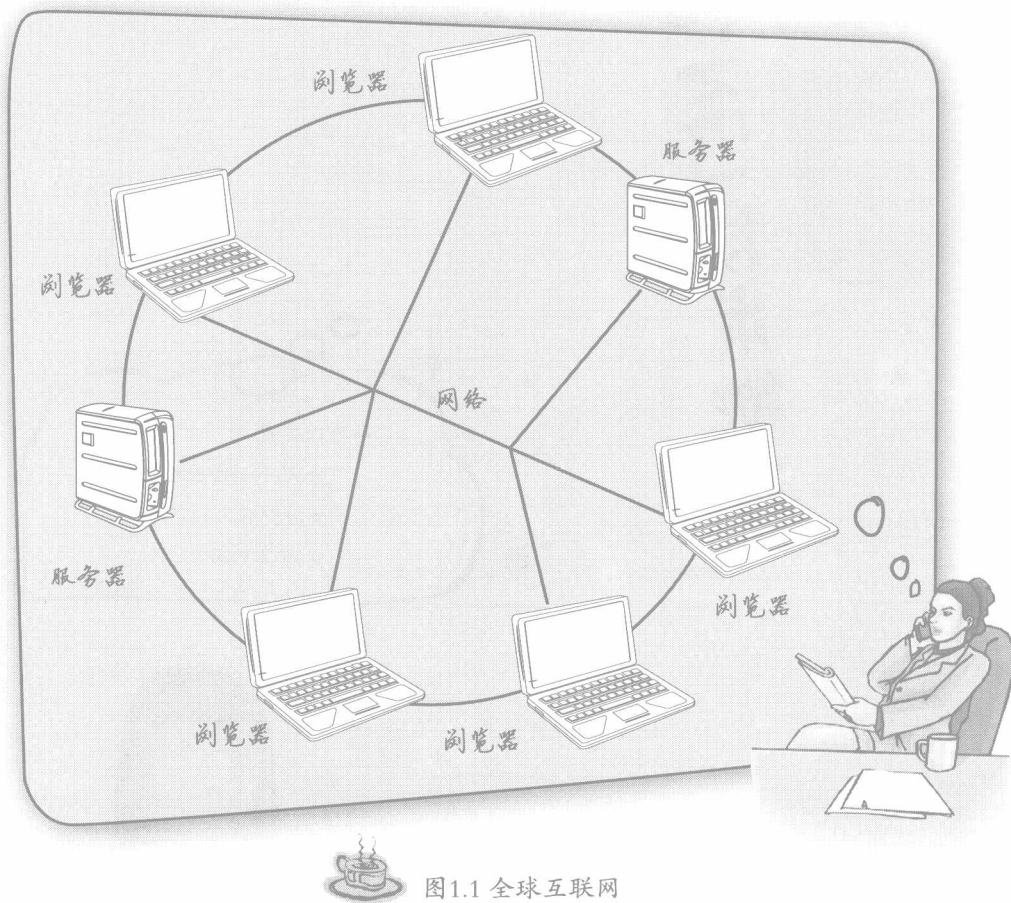


图1.1 全球互联网

虽然大家每天都在使用Web资源, 但是可能并不了解其内部工作原理。接下来将向读者详细介绍Web浏览的各个组成要素, 并透析其工作原理。

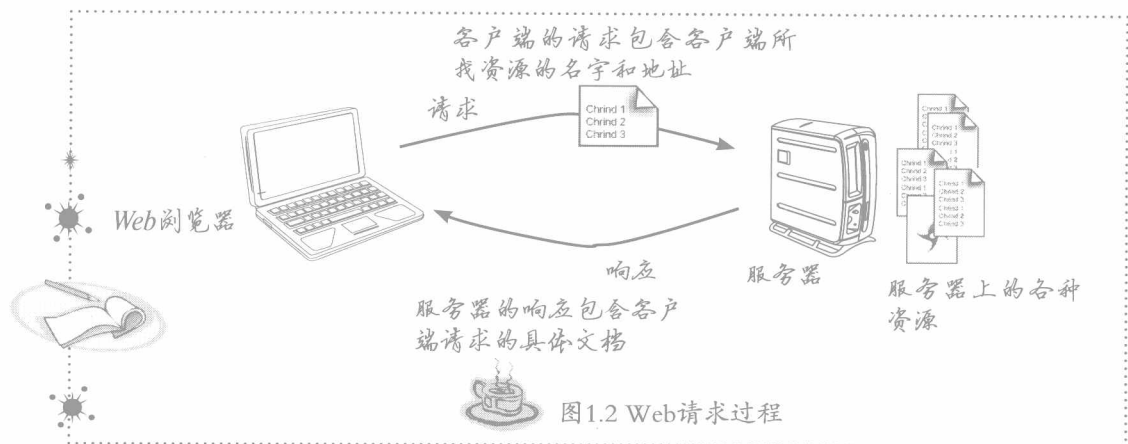
1.1.1 Web服务器

提及服务器, 读者通常联想到的是硬件设备。在这里, 我们所谓的Web服务器除了硬件设备之外, 更主要的是指计算机硬件上运行的软件。这个Web服务器软件对外提供Web服务, 供客户访问浏览。Windows Server 2003操作系统自带的IIS 6.0便是一款功能强大的Web服务器软件 (Windows XP Professional版本中也带有一个低版本的IIS 5.1, 供开发人员使用)。当服务器安装完该软件

后,便可以对外提供Web服务。

Web服务器软件的本质和其他各种提供网络服务的软件一样,接收客户端的请求,然后将特定的内容返回给客户端。

用户通过Web浏览器软件,向服务器请求一个资源。当Web服务器收到这个请求后,将替用户查找这个资源,然后将查询的结果返回给客户端。资源的内容可以多种多样,可能是一个普通的HTML页面,也可能是一个音频文件,也可能是一张图片,也可能是一个视频文件。总之,只要是客户端请求的资源,并且Web服务器能找到,就会将其返回给用户。如图1.2所示:



当然,由于请求的随意性,用户所请求的资源并不一定存在。如果服务器并没有搜寻到这个资源,则会向客户端返回一个“HTTP 404 Not Found”的错误,告知用户资源未找到。上网的人经常会碰到这样的错误提示,例如原先收藏的某个网址由于服务器更新不再存在,那么当用户再次浏览时,服务器便会返回此错误。

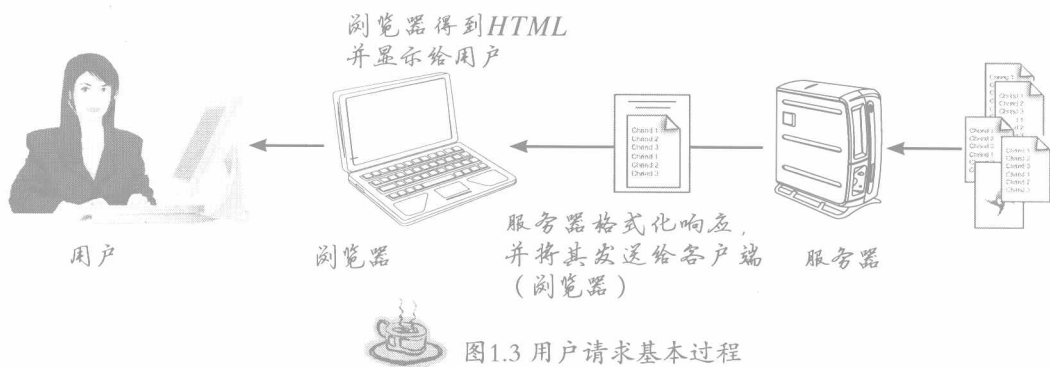
1.1.2 Web客户端

Web客户端是个比较宽泛的概念,通常我们将那些向Web服务器发送请求以获取资源的软件称为Web客户端。大家经常使用的Web客户端便是Web浏览器。微软的Windows操作系统同样自带了一个功能完善的Web浏览器Internet Explorer,简称IE。Windows XP下IE的版本是6.0。

Web客户端的功能是,根据用户的要求,发送特定的资源请求给Web服务器。当接收到Web服务器的响应后,将内容按预先定义的形式展现给用户。

如图1.3所示显示了一个基本的请求过程。





用户在浏览器地址栏内输入一个URL地址或者点击某个链接。此时，浏览器将该信息通过标准的HTTP请求发送至服务器。当服务器接收到浏览器发出的HTTP请求后，根据请求内容，从资源库中查找请求中所需要的资源信息。找到相应的资源后，服务器将该部分资源通过标准的HTTP响应发送至浏览器。浏览器在接收到响应后，将HTML文档显示给用户，整个流程如图1.3所示。

1.2 HTML与HTTP

前两个小节向读者介绍了Web访问过程中两个重要组成部分各自的功能。接下来，本书将向读者介绍Web客户端与Web服务器之间传输的内容及协议。

1.2.1 HTML 标记语言

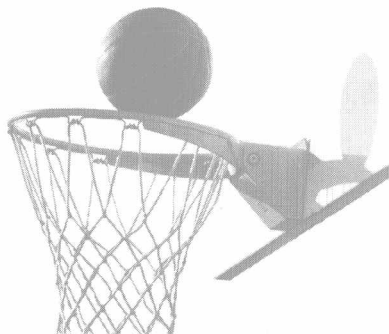
HTML全名为超文本标记语言（HyperText Markup Language），Web服务器返回给客户最常见的内容便是HTML文档。其最基本的元素是“标记（Tags）”。用户可以先输入纯文本，然后在文件中加入适当的标记，最后以htm或者html为扩展名保存，便成为一个静态的Web文件。通过浏览器浏览时，浏览器便会依照文本文件内的不同标记产生指定的效果。HTML中的标记通常是成对出现的。

较为常见的HTML标记有：

<code><html></html></code>	文档的起始和结束标记
<code><title></title></code>	HTML文档的标题
<code><table></table></code>	表格标记
<code><tr></tr></code>	表格里的行标记
<code><td></td></code>	行里的列标记
<code><a></code>	超链接标记
<code><p></p></code>	段落标记

下面是一个简单的HTML文件的例子：

```
<html>
<head>
```



```
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
```

```
<title>HTML示例</title>
```

```
</head>
```

```
<body>
```

```
<table width="100%" border="1px">
```

```
<tr>
```

```
<td align="center"><p>示例</p></td>
```

```
<td><a href="#">链接示例</a></td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

页面效果如图1.4所示:

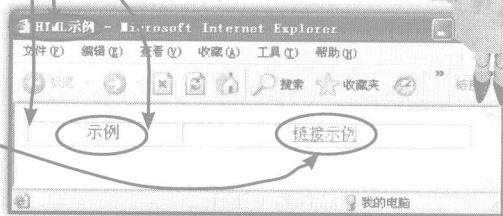


图1.4 HTML示例

1.2.2 CSS样式表

在网页中另一个重要的表现技术就是CSS层叠样式表。在当今的网页制作中,几乎所有漂亮的网页都用了CSS。CSS是Cascading Style Sheet的缩写,有些书上把它译为“层叠样式表”。有了CSS的控制,我们的网页便会给人一种赏心悦目、工工整整的感觉,同时字体的色彩变化也使主页变得更加生动活泼。虽然只短短的十几行代码,得到的效果却不同凡响。

下面以一段比较典型的CSS代码为例介绍其具体用法。

```
<html>
<head>
  <title>CSS层叠样式示例网页</title>
  <style>
    p {font-family: 宋体; font-size: 9pt; line-height: 12pt; color:red}
    a:hover {background-color: #ffccff; color: #0080ff; text-decoration: underline }
    a:link {color:#000000;text-decoration: none}
    a:visited {color:gray;text-decoration: none}
    a:active {color:green;text-decoration: none}
  </style>
</head>
<body>
<table width="100%" border="1px">
  <tr>
    <td align="center"><p>示例</p></td>
    <td><a href="#">链接示例</a></td>
  </tr>
```

```
</table>
</body>
</html>
```

上述源文件中，<style></style>标签对括起来的部分就是CSS层叠样式表。页面浏览的效果如图1.5所示：



图1.5 CSS示例页面

下面，我们逐一解释一下5行CSS代码所代表的含义。

第一行："p"代表该行样式是针对<p>标签的。font-family表示字体，font-size表示字体大小，line-height表示行高，color则表示字体颜色；

第二行："a:hover"代表鼠标移动到链接上方时的样式。background-color表示链接文字的背景色，color表示链接文字的前景色；

第三行："a:link"代表链接未被访问时的样式。color表示链接文字前景色，text-decoration表示下划线状态（none为没有下划线）；

第四行："a:visited"代表链接被访问后的样式；

第五行："a:active"代表鼠标点击该链接后的样式。

以上只是介绍了CSS样式的简单用法，更为复杂的效果请参阅其他专题书籍。

1.2.3 脚本语言

HTML语言本身的内容是有限的，通过它只能显示一些基本的元素，网页内容显得枯燥乏味。于是，Netscape公司与Sun公司合作开发了JavaScript脚本语言，使得网页的可交互性得到了显著提高。在此之后，微软公司也发布了自己的脚本语言VBScript。

无论是哪种脚本语言，其运行的环境都只局限于Web浏览器。它们只能通过Web浏览器去完成某种操作而不是像普遍意义上的程序那样可以独立运行。因为需要由Web浏览器进行解释和执行，所以