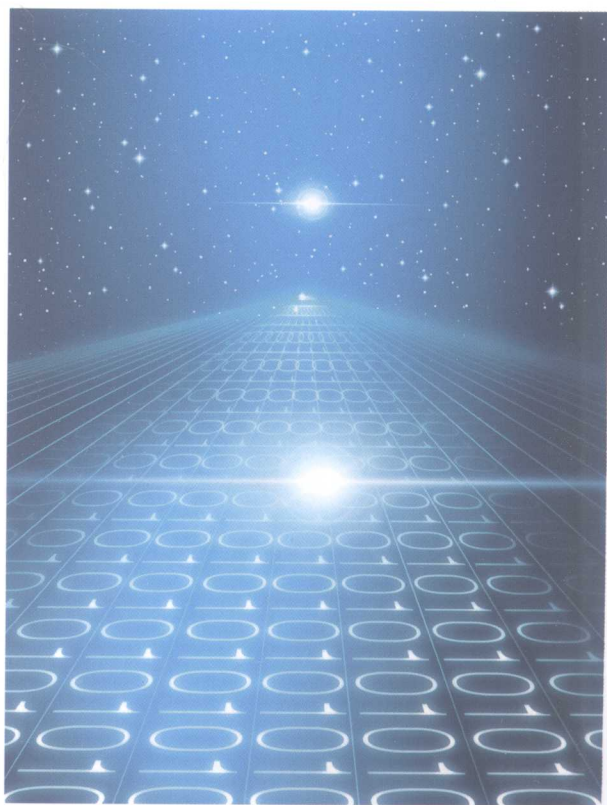


ASP.NET 2.0简明教程

(C# 2005篇)

- ◆ ASP.NET 2.0开发基础
- ◆ C#语言基础知识
- ◆ 服务器控件及验证控件
- ◆ ASP.NET常用对象
- ◆ C#面向对象程序设计
- ◆ ADO.NET数据库编程
- ◆ 数据绑定及相关控件
- ◆ 配置ASP.NET应用程序
- ◆ 母版页和站点导航
- ◆ XML数据管理
- ◆ 博客系统实例



金雪云 汪文彬 陈文臣 编著



清华大学出版社

高等学校计算机应用规划教材

ASP.NET 2.0 简明教程

(C# 2005 篇)

金雪云 汪文彬 陈文臣 编著

清华大学出版社

北 京

内 容 简 介

ASP.NET 2.0 是一种基于服务器的功能强大的技术,用于为 Internet 或企业的内部网(Intranet)创建动态的、交互式的 HTML 网页。它构成了 .NET Framework 的核心元素,为异常强大的 .NET 开发环境提供基于 Web 的访问。

本书包括 11 章,系统介绍了如何使用 ASP.NET 开发动态网站,具体包括 ASP.NET 的运行环境、C# 编程语言概述、ASP.NET 控件的使用、数据绑定、母版页、网站导航、XML 数据管理等内容,最后通过一个具体的动态网站开发项目为读者演示了用 ASP.NET 2.0 进行动态网站开发的方法和思路。

本书教学重点明确,结构合理,语言简明,书中实例均为作者在本领域工作中的真实案例,具有很强的实用性,适用对象为 ASP.NET 的初、中级读者。本书可以作为高等学校计算机相关专业的教材,也适合 ASP.NET 网站开发人员参考。

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。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

ASP.NET 2.0 简明教程(C# 2005 篇)/金雪云,汪文彬,陈文臣 编著. —北京:清华大学出版社,2009.3
(高等学校计算机应用规划教材)

ISBN 978-7-302-19609-9

I. A… II. ①金…②汪…③陈… III. 主页制作—程序设计—高等学校—教材 IV.TP393.092

中国版本图书馆 CIP 数据核字(2009)第 025330 号

责任编辑:王 定 刘金喜

装帧设计:康 博

责任校对:胡雁翎

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市世界知识印刷厂

装 订 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185×260 印 张:21.25 字 数:491 千字

版 次:2009 年 3 月第 1 版 印 次:2009 年 3 月第 1 次印刷

印 数:1~5000

定 价:32.00 元

本书如存在文字不清、漏印、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)62770177 转 3103 产品编号:027386-01

前 言

ASP.NET 2.0 是一种基于服务器的功能强大的技术,用于为 Internet 或企业的内部网 (Intranet)创建动态的、交互式的 HTML 网页。ASP.NET 2.0 构建在 .NET Framework 2.0 之上,扩展了 ASP.NET 1.0 和 ASP.NET 1.1 的功能,其内核是一个基于控件的、事件驱动的架构,这意味着只需要向页面添加少量的代码,就可以完成强大的功能。

本书面向初、中级用户,由浅入深、全面系统地介绍了使用 ASP.NET 和 C# 开发网站的基础知识、基本方法和具体应用。如果您是一名高手,这本书对您来说也将是一本极好的参考书。

本书共分为 11 章,包括 ASP.NET、C# 编程语言、HTML 和脚本语言的介绍,ASP.NET 的服务器控件和核心对象、数据访问、母版页、网站导航等技术的讲解,最后通过一个综合实例展示了如何使用多种技术来开发网站。

第 1 章讲解了 ASP.NET 的基础知识。首先对 ASP.NET 技术进行了概括介绍,然后介绍了如何构建 ASP.NET 的运行环境,最后通过一个实例介绍了如何使用 ASP.NET 创建 Web 程序。

第 2 章讲解了 C# 2005 编程语言。首先讲解了 C# 的数据类型和变量,然后讲解了流程控制的相关知识,最后讲解了 C# 面向对象的特征:类和对象。

第 3 章讲解了 ASP.NET 服务器控件。首先介绍了 HTML 服务器控件,以及 ASP.NET 控件的共有属性,接着介绍了常用的 Web 服务器控件,最后介绍了 ASP.NET 2.0 中的高级控件。

第 4 章讲解了 ASP.NET 2.0 验证控件。主要包括数据验证的方法、数据验证控件的分类、具体的数据验证控件的使用等内容。

第 5 章讲解了 ASP.NET 中的核心对象。主要包括 Response 对象、Request 对象、Server 对象、Session 对象、Cookie 对象以及 Application 对象。

第 6 章讲解了 ADO.NET 数据库编程。首先概括介绍了 ADO.NET,然后介绍了如何连接数据库和获取数据,最后介绍了如何访问和填充数据集。

第 7 章讲解了如何进行数据绑定。重点介绍的数据绑定控件包括 GridView、Repeater 和 DataList。

第 8 章介绍了 ASP.NET 程序的配置。主要包括使用 web.config 和使用 global.asax 进行配置。

第 9 章介绍了母版页和站点导航。主要介绍了如何创建、使用和动态加载母版页,以及如何使用站点地图、Menu 控件和 TreeView 控件进行站点导航。

第 10 章介绍了 XML 数据管理。首先介绍了 XML 的基础知识,然后介绍了如何使用流模型操作 XML 数据,接着介绍了使用 XmlDocument 类编辑 XML 数据的方法,最后介

绍了 DataSet 和 XML 的相互转换。

第 11 章通过开发一个“博客”系统演示如何综合使用多种技术来开发 Web 网站。本章除了介绍 ASP.NET 的具体技术之外,对于需求分析和系统设计、数据库设计以及功能模块的划分也有比较详细的介绍,有利于读者了解一个实际项目的开发流程。

本书对每个知识点都有实例演示。为了便于读者快速了解 ASP.NET 技术,本书的实例没有纠缠于具体而又繁琐的程序细节,而是简单明了地说明了如何运用知识点。本书最后一章是一个综合案例,取材于现实生活,希望读者通过这个案例,对如何使用 ASP.NET 编程有比较深刻的了解。

本书的实例源代码可通过 <http://www.tupwk.com.cn/downpage> 下载。

本书由金雪云、汪文彬、陈文臣、何昀峰编写,周明辉对本书进行了统稿,另外,参与本书编写和修改的还有李志伟、黄明、刘国防、黄波、王庆、李江波、杜杰、郑思普、杨军、叶明、崔宁等同志。在此,编者对以上人员致以诚挚的谢意!

由于本书涉及的范围比较广泛,作者的经验有限,时间仓促,书中难免有不足之处,敬请广大读者、专家提出宝贵意见。

编 者

2008 年 12 月

目 录

第 1 章	ASP.NET 2.0 开发基础	1
1.1	ASP.NET 简介	1
1.1.1	.NET 简介	1
1.1.2	ASP.NET 页面与 Web 服务器的交互过程	2
1.2	搭建 ASP.NET 2.0 开发环境	3
1.2.1	安装和配置 IIS	3
1.2.2	配置集成开发环境	8
1.3	创建第一个 ASP.NET 2.0 应用程序	12
1.3.1	创建 Web 站点	12
1.3.2	编写 ASP.NET 2.0 应用程序	14
1.3.3	编译和运行应用程序	15
1.4	习题	16
1.4.1	填空题	16
1.4.2	选择题	16
1.4.3	问答题	16
1.4.4	上机操作题	17
第 2 章	C#语言快速掌握	19
2.1	数据类型	19
2.1.1	值类型	20
2.1.2	引用类型	24
2.1.3	装箱和拆箱	27
2.1.4	指针类型	27
2.2	变量和常量	28
2.2.1	变量	28
2.2.2	常量	29
2.3	运算符	29
2.3.1	算术运算符	30
2.3.2	赋值运算符	31
2.3.3	关系运算符	32

2.3.4	逻辑运算符	32
2.3.5	条件运算符	33
2.3.6	位运算符	34
2.3.7	运算符的优先级	35
2.4	流程控制	36
2.4.1	条件语句	37
2.4.2	循环语句	43
2.4.3	跳转语句	45
2.5	类和对象	47
2.5.1	类	47
2.5.2	属性和方法	49
2.5.3	对象的创建和回收	49
2.5.4	继承和多态	52
2.6	委托与事件	58
2.6.1	委托和事件	58
2.6.2	使用委托进行回调	59
2.6.3	动态注册和移除事件	60
2.7	习题	62
2.7.1	填空题	62
2.7.2	选择题	63
2.7.3	问答题	63
2.7.4	上机操作题	63
第 3 章	ASP.NET 服务器控件	65
3.1	HTML 控件	65
3.1.1	表格	65
3.1.2	表单	68
3.1.3	图像	73
3.2	ASP.NET 控件的共有属性	75
3.3	基本 ASP.NET 控件	84
3.3.1	显示数据	84
3.3.2	显示图像	86

3.3.3 输入数据	88	4.8.4 上机操作题	142
3.3.4 按钮控件	91	第 5 章 ASP.NET 常用对象	145
3.3.5 超链接控件	93	5.1 基本输出对象 Response	145
3.3.6 列表控件	95	5.1.1 Response 对象的 属性和方法	145
3.3.7 多选操作	97	5.1.2 输出字符串和文件	146
3.3.8 单选操作	103	5.1.3 网页重定向	147
3.3.9 下拉列表框	107	5.2 基本输入对象 Request	150
3.3.10 控件的容器	109	5.2.1 Request 对象的属性和方法	150
3.3.11 表格控件	111	5.2.2 获取浏览器信息	150
3.4 ASP.NET 2.0 高级控件	114	5.2.3 获取 HTTP 中的信息	151
3.4.1 使用 BulletedList 控件 实现项目符合和编号	114	5.3 Server 对象	153
3.4.2 ImageMap 控件	116	5.3.1 Server 对象的属性和方法	153
3.5 ASP.NET 2.0 功能增强控件	118	5.3.2 利用 Server 对象进行 HTML 编码解码	155
3.5.1 使用 Calendar 控件 输入日期	118	5.3.3 利用 Server 对象进行 URL 编码和解码	157
3.5.2 使用 AdRotator 控件显示 页面广告	121	5.4 Session 对象	159
3.6 习题	126	5.4.1 Session 对象的属性和方法	159
3.6.1 填空题	126	5.4.2 Session 对象的唯一性和 有效时间	159
3.6.2 选择题	126	5.5 Cookie 对象	164
3.6.3 问答题	126	5.5.1 Cookie 对象的属性和方法	164
3.6.4 上机操作题	126	5.5.2 访问 Cookie	165
第 4 章 验证控件	129	5.6 Application 对象	165
4.1 验证控件简介	129	5.6.1 如何使用 Application 对象	166
4.2 非空验证	130	5.6.2 同步 Application 状态	166
4.3 范围验证	132	5.6.3 网站的访问计数	167
4.4 使用正则表达式进行验证	134	5.7 习题	168
4.5 验证两个控件的输入值 是否相同	136	5.7.1 填空题	168
4.6 使用自定义函数的验证	138	5.7.2 选择题	169
4.7 页面统一验证	140	5.7.3 问答题	169
4.8 习题	141	5.7.4 上机操作题	169
4.8.1 填空题	141	第 6 章 ADO.NET 数据库编程	171
4.8.2 选择题	141	6.1 ADO.NET 的基本对象	171
4.8.3 问答题	142	6.1.1 ADO.NET 简介	171

6.1.2 ADO.NET 组件结构	172	7.3.2 在 Repeater 控件中显示 数据编号	211
6.2 连接数据库	172	7.4 DataList 控件	214
6.2.1 建立 SQL Server 数据库	173	7.4.1 DataList 控件概述	214
6.2.2 连接 SQL Server 数据库	176	7.4.2 使用 SelectedItemTemplate 模板	214
6.2.3 连接 Microsoft Access 数据库	177	7.4.3 应用 DataKeyField 属性 获取主键	216
6.3 读取数据	179	7.4.4 在 DataList 控件中 编辑数据	218
6.3.1 使用 SqlCommand 类	179	7.5 ListBox 和 DropDownList 用于数据绑定	222
6.3.2 使用 OleDbCommand 类	181	7.6 习题	225
6.3.3 使用存储过程	182	7.6.1 填空题	225
6.4 使用 DataReader	184	7.6.2 选择题	225
6.5 填充数据集	187	7.6.3 问答题	226
6.5.1 使用 DataAdapter	188	7.6.4 上机操作题	226
6.5.2 使用 DataTable、 DataColumn 和 DataRow	190	第 8 章 配置 ASP.NET 应用程序	229
6.5.3 访问数据集	192	8.1 使用 web.config 进行配置	229
6.6 习题	195	8.1.1 身份验证和授权	230
6.6.1 填空题	195	8.1.2 web.config 文件的 其他设置	231
6.6.2 选择题	196	8.1.3 在代码中获取 Web.Config 应用程序设置	234
6.6.3 问答题	196	8.2 使用 global.asax 进行配置	237
6.6.4 上机操作题	196	8.2.1 编写 Application_Start 和 Application_End 事件 处理代码	238
第 7 章 数据绑定	199	8.2.2 编写 Session_Start 和 Session_End 事件代码	241
7.1 数据绑定简介	199	8.2.3 编写错误处理程序	243
7.1.1 简单数据绑定和复杂 数据绑定	199	8.3 习题	246
7.1.2 用于简单数据绑定的控件	200	8.3.1 填空题	246
7.2 GridView 控件	202	8.3.2 选择题	246
7.2.1 GridView 控件概述	202	8.3.3 问答题	246
7.2.2 在 GridView Web 服务器 控件中分页	203	8.3.4 上机操作题	247
7.2.3 对 GridView Web 服务器 控件中的数据进行排序	206		
7.2.4 控制 GridView 中的列	207		
7.3 Repeater 控件	211		
7.3.1 Repeater 控件概述	211		

第 9 章 母版页和站点导航.....	249
9.1 母版页概述.....	249
9.1.1 母版页的基础知识.....	249
9.1.2 母版页运行机制.....	251
9.1.3 母版页的优点.....	251
9.2 母版页的创建和使用.....	251
9.2.1 创建母版页.....	251
9.2.2 创建内容页.....	253
9.2.3 使用母版页.....	255
9.2.4 动态加载母版页.....	258
9.3 站点导航.....	264
9.3.1 基于 XML 的站点地图.....	264
9.3.2 SiteMapDataSource 服务器控件.....	266
9.4 TreeView 服务器控件.....	266
9.5 Menu 服务器控件.....	268
9.6 习题.....	270
9.6.1 填空题.....	270
9.6.2 选择题.....	271
9.6.3 问答题.....	271
9.6.4 上机操作题.....	271
第 10 章 ASP.NET 中的 XML	
数据管理.....	273
10.1 XML 基础.....	273
10.1.1 XML 概述.....	273
10.1.2 XML 文档结构.....	274
10.2 使用流模型操作 XML 数据.....	275
10.2.1 读取 XML 数据.....	275
10.2.2 写入 XML 数据.....	276
10.3 使用 XmlDocument 类编辑 XML 数据.....	278
10.4 使用 XML 数据填充 DataSet.....	286
10.4.1 DataSet 和 XML 之间 的关系.....	287
10.4.2 DataSet 和 XML 的 相互转换.....	287
10.4.3 DataSet 的 XML 架构.....	289
10.5 XmlDataSource 控件.....	291
10.6 习题.....	292
10.6.1 填空题.....	292
10.6.2 选择题.....	293
10.6.3 问答题.....	293
10.6.4 上机操作题.....	293
第 11 章 博客系统.....	295
11.1 系统设计.....	295
11.1.1 需求分析.....	295
11.1.2 总体设计.....	295
11.1.3 功能设计.....	296
11.2 系统数据库设计.....	297
11.2.1 数据库的需求分析.....	297
11.2.2 数据库概念结构设计.....	297
11.2.3 数据库的逻辑设计.....	299
11.3 配置文件 web.config.....	304
11.4 客户端.....	305
11.4.1 首页.....	305
11.4.2 日志分类列表页面.....	311
11.4.3 日志详细内容页面.....	312
11.4.4 我的相册.....	314
11.4.5 我的视频.....	316
11.4.6 留言页面.....	316
11.5 博客系统管理端.....	319
11.5.1 日志的管理和维护.....	319
11.5.2 评论管理页面.....	325
11.6 习题.....	327
11.6.1 填空题.....	327
11.6.2 选择题.....	327
11.6.3 问答题.....	328
11.6.4 上机操作题.....	328

第1章 ASP.NET 2.0开发基础

ASP.NET 是 Microsoft .NET Framework 中一套用于生成 Web 应用程序和 XML Web Services 的技术。ASP.NET 页面在服务器上执行,并生成发送到桌面或移动浏览器的标记(如 HTML、WML 或 XML),该页面使用一种已编译的、由事件驱动的编程模型,这种模型可以提高性能,并支持将应用程序逻辑同用户界面相隔离。

本章重点内容:

- Web 和 ASP.NET 的基本概念
- ASP.NET 的运行环境
- Visual Studio 2005 开发环境

1.1 ASP.NET 简介

ASP.NET 是微软公司为了迎接网络时代的来临,提出的一个统一的 Web 开发模型。ASP.NET 是建立在公共语言运行库上的编程框架,可用于在服务器上生成功能强大的 Web 应用程序。

1.1.1 .NET 简介

.NET 是微软公司发布的新一代的系统、服务和编程平台,主要由 .NET Framework 和 Microsoft Visual Studio .NET 开发工具组成。

.NET Framework 是一种新的计算平台,它包含了操作系统上软件开发的所有层,简化了在高度分布式 Internet 环境中的应用程序开发。.NET Framework 主要包括两个最基本的内核,即公共语言运行库(Common Language Runtime, CLR)和 .NET Framework 基本类库,它们为 .NET 平台的实现提供了底层技术支持。下面将分别做详细的介绍。

1. 公共语言运行库

公共语言运行库是 .NET Framework 的基础,是 .NET Framework 的运行时环境。公共语言运行库是一个在执行时管理代码的代理,以跨语言集成、自描述组件、简单配置和版本化及集成安全服务为特点,提供核心服务(如内存管理、线程管理和远程处理)。公共语言运行库还强制实施严格的类型安全以及可确保安全性和可靠性的其他形式的代码准确性。公共语言运行库遵循公共语言架构(CLI)标准,可以使 C++、C#、Visual Basic 以及 JScript 等多种语言能够深度集成。在 .NET Framework 中,用一种语言所写的代码能继承用另一种语言所写的类的实现,用一种语言所写的代码抛出的异常能被用另一种语言所写的代码捕获。

2. .NET 基类库

.NET Framework 的另一个主要组件是类库，它是一个综合性的面向对象的可重用类型集合，例如 ADO.NET、ASP.NET 等。.NET 基类库位于公共语言运行库的上层，与 .NET Framework 紧密集成在一起，可被 .NET 支持的任何语言使用。这也就是为什么 ASP.NET 中可以使用 C#、VB.NET、VC.NET 等语言进行开发。.NET 类库非常丰富，提供数据库访问、XML、网络通信，线程，图形图像、安全、加密等多种功能服务。类库中的基类提供了标准的功能，如输入输出、字符串操作、安全管理、网络通信、线程管理、文本管理和用户界面设计功能。这些类库使得开发人员更容易地建立应用程序和网络服务，从而提高开发效率。

1.1.2 ASP.NET 页面与 Web 服务器的交互过程

ASP.NET 是一个统一的 Web 开发模型，它包括使用尽可能少的代码生成企业级 Web 应用程序所必需的各种服务。ASP.NET 作为 .NET Framework 的一部分提供。

ASP.NET 网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。使用 ASP.NET 网页可以为网站创建动态内容。通过使用静态 HTML 页(.htm 或 .html 文件)，服务器读取文件并将该文件按原样发送到浏览器，以此来满足 Web 请求。相比之下，当用户请求 ASP.NET 网页(.aspx 文件)时，该页则作为程序在 Web 服务器上运行。该页运行时，可以执行网站要求的任何任务，包括计算值、读写数据库信息或者调用其他程序。该页动态地生成标记(HTML 或另一种标记语言中的元素)，并将该标记作为动态输出发送到浏览器。

ASP.NET 页面作为代码在服务器上运行。因此，要得到处理，页面必须在用户单击按钮(或者当用户选中复选框或与页面中的其他控件交互)时提交到服务器。每次页面都会提交回自身，以便它可以再次运行其服务器代码，然后向用户呈现其自身的新版本。传递 Web 页面的过程如下：

用户请求页面。使用 HTTP GET 方法请求页面，页面第一次运行，执行初步处理(如果已通过编程让它执行初步处理)。

页面将标记动态呈现到浏览器，用户看到的网页类似于其他任何网页。

(1) 用户输入信息或从可用选项中进行选择，然后单击按钮。如果用户单击链接而不是按钮，页面可能仅仅定位到另一页，而第一页不会被进一步处理。

(2) 页面被发送到 Web 服务器。浏览器执行 HTTP POST 方法，该方法在 ASP.NET 中称为“回发”。更明确地说，页面发送回其自身。例如，如果用户正在使用 Default.aspx 页面，则单击该页上的某个按钮可以将该页发送回服务器，发送的目标则是 Default.aspx。

(3) 在 Web 服务器上，该页被再次运行，并且可在页上使用用户输入或选择的信息。

(4) 页面执行通过编程所要实行的操作。

(5) 页面将其自身呈现回浏览器。

只要用户在该页面中工作，此循环就会继续。用户每次单击按钮时，页面中的信息会发送到 Web 服务器，然后该页面再次运行。每个循环称为一次“往返行程”。由于页面处理发生在 Web 服务器上，因此页面可以执行的每个操作都需要一次到服务器的往返行程。

此外，ASP.NET 网页是完全面向对象的。在 ASP.NET 网页中，可以使用属性、方法

和事件来处理 HTML 元素。ASP.NET 页框架为响应在服务器上运行的代码中的客户端事件提供统一的模型,从而我们不必考虑基于 Web 的应用程序中固有的客户端和服务端隔离的实现细节。该框架还会在页面处理生命周期中自动维护页及该页上控件的状态。

使用 ASP.NET 页和控件框架还可以将常用的 UI 功能封装成易于使用且可重用的控件。控件只需编写一次,即可用于许多页面并集成到 ASP.NET 网页中。这些控件在呈现期间放入 ASP.NET 网页中。

1.2 搭建 ASP.NET 2.0 开发环境

下面将介绍 ASP.NET 的基本运行环境和安装配置方法,其中包括 IIS 的安装、IIS 服务器和在 IIS 中创建虚拟目录。

1.2.1 安装和配置 IIS

1. 安装 IIS

在 Windows 2000 上安装的 IIS 的版本号是 5.0,在 Windows XP 上安装的 IIS 的版本号是 5.1,除此之外,这两种安装过程的区别不大。这里我们以 Windows XP 为例来介绍如何安装 IIS 5.1,需要注意的是,不能在 Windows XP Home 版上安装 IIS,因此不能在其上运行 ASP.NET。安装 IIS 的步骤如下:

(1) 选择“开始”|“控制面板”|“添加或删除程序”命令,显示如图 1-1 所示的对话框,该对话框显示计算机中当前已经安装的程序。

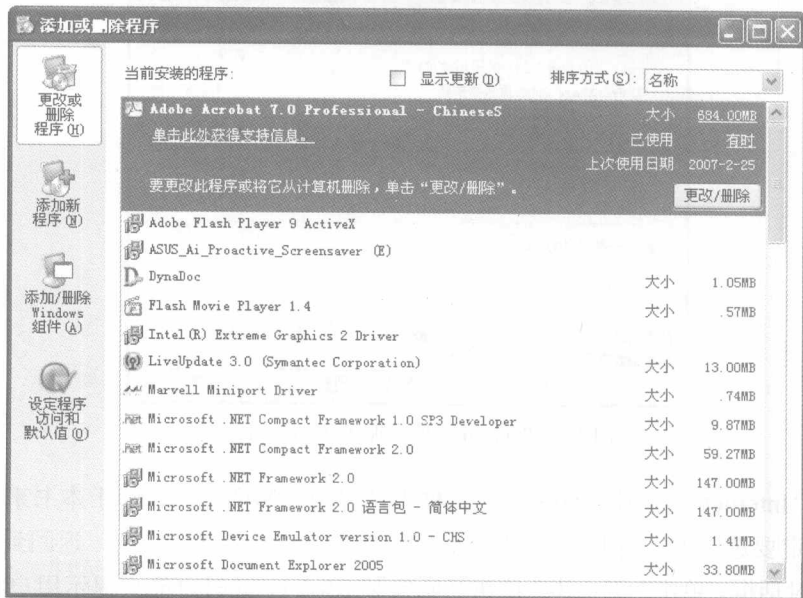


图 1-1 “添加和删除程序”对话框

(2) 在对话框的左侧单击“添加/删除 Windows 组件”图标，弹出“Windows 组件向导”对话框，如图 1-2 所示。

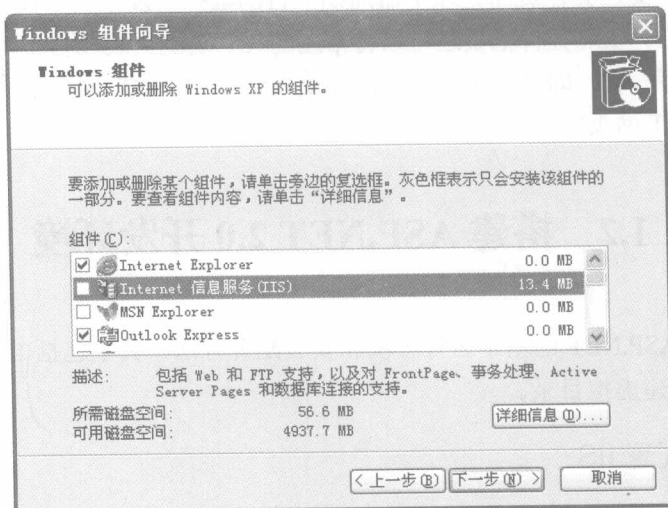


图 1-2 “Windows 组件向导”对话框

(3) 在“Windows 组件向导”对话框中找到“Internet 信息服务(IIS)”，如果尚未安装，则其左侧的复选框不会被选中；如果复选框是不可选状态，说明 IIS 的组件没有全部安装。否则说明 IIS 已经全部安装，退出安装过程。

(4) 如果复选框没有被选中，则选中该复选框。如果复选框是不可选状态，则选中该项，单击“详细信息”按钮，弹出如图 1-3 所示的对话框。

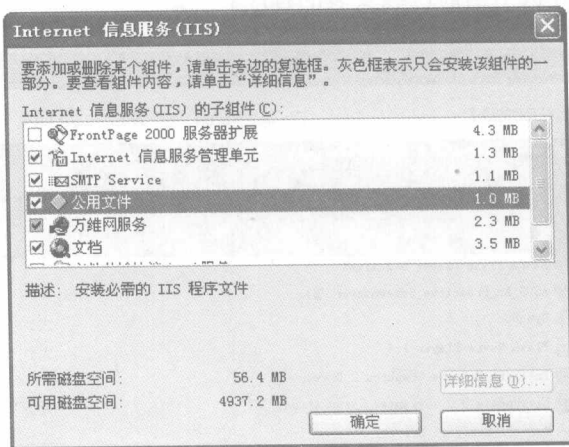


图 1-3 “Internet 信息服务(IIS)”对话框

(5) 在“Internet 信息服务(IIS)”对话框中选择要安装的选项，对于本书来说，“公用文件”是一定要选中的。选择完毕要安装的项后，单击“确定”按钮，返回到“Windows 组件向导”对话框。单击“下一步”按钮开始安装 IIS 5.1，此时可能会提示用户将 Windows XP 系统盘放入光驱。

(6) 安装完毕后, 返回到“添加或删除程序”对话框。

一旦安装完成, 系统会自动启动 IIS, 而且在此之后, 无论何时启动 Windows, 系统都会自动启动 IIS。因此, 用户不需要运行启动程序, 也不需要像启动 Word 等程序那样双击快捷方式。

我们可以通过“Internet 信息服务”对话框关闭 IIS, 步骤为如下:

(1) 选择“控制面板”|“管理工具”|“Internet 信息服务”命令, 弹出“Internet 信息服务”对话框,

(2) 依次展开根节点、“网站”节点、“默认网站”节点, 如图 1-4 所示。

(3) 右键单击“默认网站”节点, 弹出如图 1-5 所示的快捷菜单。

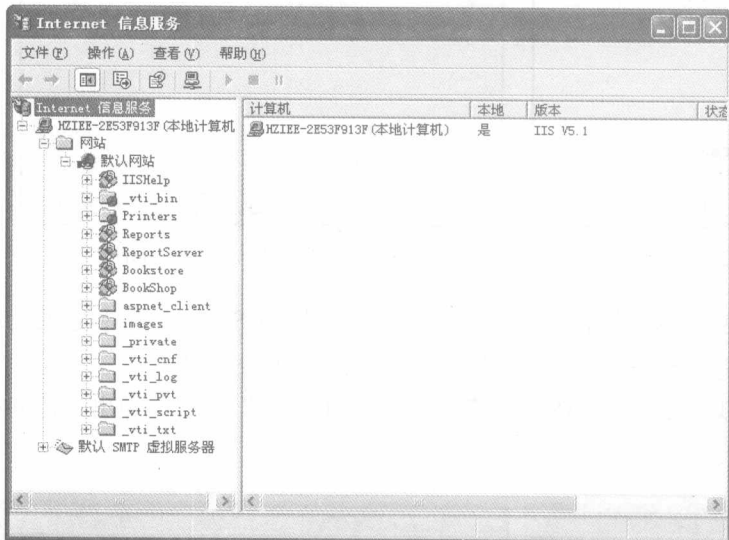


图 1-4 Internet 信息服务

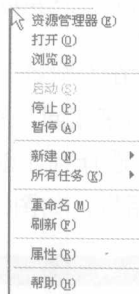


图 1-5 “默认网站”快捷菜单

(4) 我们可以选择“停止”关闭 IIS 服务, 也可以选择“暂停”关闭 IIS 服务。后面讲述管理 Web 服务器的目录时我们还会通过“新建”菜单来建立新的网站。

如果采用默认安装, IIS 在硬盘驱动器 C 的根目录中会创建 InetPub 目录, 该目录包含用于存放所创建的 Web 页面文件的子目录。我们创建的 Web 网站默认情况下都会保存到 InetPut 的子目录 wwwroot 中。fiproot、mailroot、nntproot 目录构成使用 FTP、邮件和新闻服务的站点目录树的顶级目录。

2. 管理 Web 服务器的目录

当用户通过 HTTP 浏览位于 Web 服务器上的一些 Web 页面时, Web 服务器需要确定与该页面对应的文件位于服务器硬盘上的什么位置。事实上, 在由 URL 给出的信息与包含页面的文件的物理位置(在 Web 服务器的文件系统中)之间有着重要的关系。这个关系是通过虚拟目录来实现的。

虚拟目录相当于物理目录在 Web 服务器机器上的别名, 它不仅使用户避免了冗长的 URL, 也是一种很好的安全措施, 因为虚拟目录对所有浏览者隐藏了物理目录结构。下面

通过一个具体的例子来说明如何创建虚拟目录。步骤如下：

(1) 在硬盘上创建一个物理目录，这里我们在 C 盘的根目录下创建一个目录，命名为 ASPTest。

(2) 启动 Internet 信息服务，右键单击“默认网站”节点，在如图 1-5 所示的菜单中选择“新建”|“虚拟目录”命令，启动虚拟目录创建向导，如图 1-6 所示。

(3) 单击“下一步”按钮，弹出“虚拟目录别名”对话框，如图 1-7 所示。

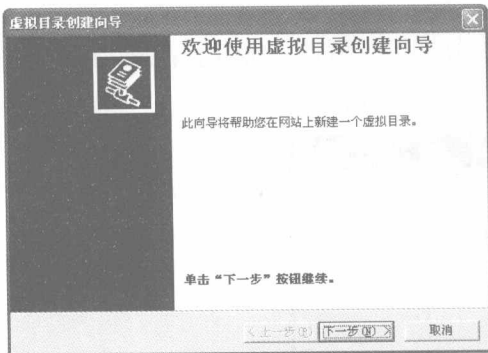


图 1-6 “虚拟目录创建向导”对话框

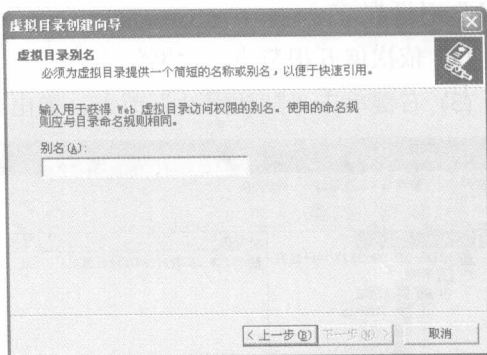


图 1-7 “虚拟目录别名”对话框

(4) 在“别名”文本框中输入虚拟目录的名字，这里命名为“ASPTest”，和它的物理目录的名字相同。然后单击“下一步”按钮，弹出如图 1-8 所示的对话框。

(5) 选择刚才创建的物理目录“C:\ASPTTest”，单击“下一步”按钮，弹出“访问权限”对话框，如图 1-9 所示。

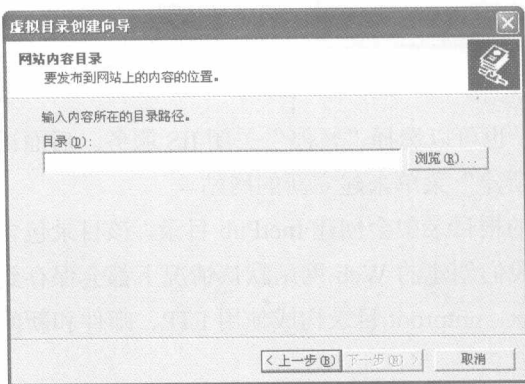


图 1-8 网站内容目录

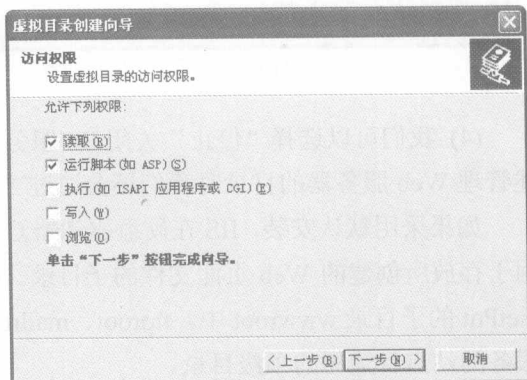


图 1-9 访问权限

(6) 在“访问权限”对话框中设置虚拟目录的访问权限，除非读者明白自己需要什么样的权限，否则不要改变创建时默认的权限。单击“下一步”按钮，弹出如图 1-10 所示的对话框。

(7) 单击“完成”按钮，完成虚拟目录的创建。此时，在“Internet 信息服务”窗口的目录树中将显示该 ASPTest 虚拟目录，如图 1-11 所示。

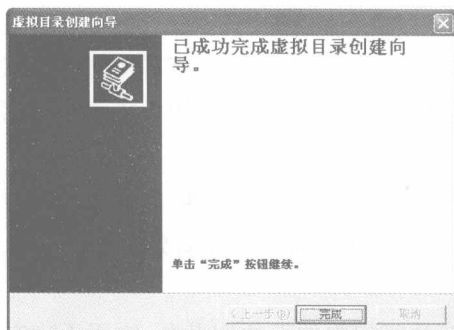


图 1-10 创建完成

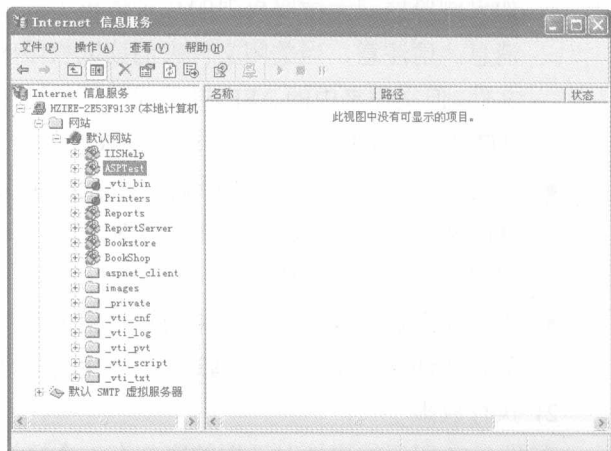


图 1-11 新创建的虚拟目录

当创建目录时，可以使用虚拟目录向导中提供的选项，将权限赋给目录，也可以在“Internet 信息服务”窗口的 ASPTest 虚拟目录中右击，从弹出的快捷菜单中选择“属性”命令，弹出如图 1-12 的对话框，来设置虚拟目录的访问权限。

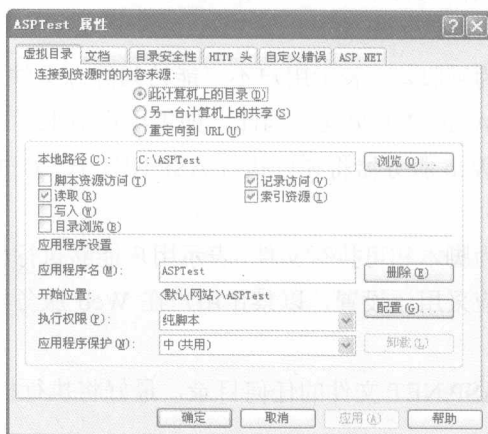


图 1-12 “ASPTest 属性”对话框

1) 访问权限

在图 1-12 所示的对话框中，我们感兴趣的是位于左边的 4 个复选框。它们用于确定给定目录的访问类型，并说明包含于目录中的文件的允许权限。下面分别介绍这 4 个复选框。

- “脚本资源访问”复选框：该权限允许用户访问 ASP.NET 页面的源代码。只有赋予了读取或写入权限后，才允许使用该权限。但通常不希望用户能够浏览 ASP.NET 源代码，因此，一般情况下包含 ASP.NET 页面的任何目录均不选中该复选框。默认情况下，在设置过程中所有目录均禁用脚本资源访问权限，应保留该默认设置。
- “读取”复选框：该权限允许浏览器读取或下载保存在主目录或虚拟目录中的文件。

如果浏览器向无读取权限的目录请求了一个文件, Web 服务器将只返回一个错误消息。请注意, 当关闭文件夹的读取权限时, 就不能读取位于文件夹中的 HTML 文件, 但文件夹中的 ASP.NET 代码仍能够运行。总的来说, 包含希望发布的信息(如 HTML 文件)的目录应具有读取权限。

- “写入”复选框: 如果启用虚拟目录中的写入权限, 用户就能够在该目录中创建或修改文件, 并修改这些文件的属性。出于安全性考虑, 通常并不启用该权限, 而且建议用户不要改变该设置。
- “目录浏览”复选框: 如果用户希望他人浏览目录的内容(即查看包含在目录中的所有文件列表), 那么可通过选中“目录浏览”复选框来使该权限有效。

2) 执行权限

在图 1-14 所示的属性对话框的底部有一个标记为执行权限的下拉列表框, 它用于确定在包含于指定目录中的页面上允许执行什么级别的程序。这里有 3 种可能的值: 无、纯脚本、脚本和可执行文件。

- 将执行权限设置为无, 表示用户只能访问静态文件, 如图像文件、HTML 文件。对用户来说, 包含于指定目录的其他任何基于脚本的可执行文件都是不可访问的。如果用户试图从权限被设置成“无”的文件夹中运行一个 ASP.NET 页面, 就会在页面上得到禁止执行访问的消息。
- 将执行权限设置为纯脚本, 表示用户不仅能访问静态文件, 也能访问任何一个基于脚本的页面, 如 ASP.NET 页面。因此, 如果用户请求了一个包含于指定目录中的 ASP.NET 页面, Web 服务器将允许执行 ASP.NET 代码。并且, 由此得到的 HTML 会传递回浏览器。
- 将执行权限设置为脚本和可执行文件, 表示用户能够执行包含在目录中的任何类型的文件。通常不应采用该设置, 以禁止用户在 Web 服务器上执行具有潜在破坏性的应用程序。

对于包含要发布的 ASP.NET 文件的任何目录, 最好将执行权限设置为纯脚本。

1.2.2 配置集成开发环境

Visual Studio 2005 是一个功能强大的集成开发环境, 在该开发环境中可以创建 Windows 应用程序、ASP.NET 应用程序、ASP.NET 服务和控制台程序等。

1. 集成开发环境简介

打开 Visual Studio 2005 集成开发环境, 如图 1-13 所示, 可以看到界面主要由几个不同的部分组成。

在进行页面设计时需要使用到“属性”对话框, 在此对话框中, 用户可以对页面的一些属性值进行设置, 这些设置的属性值会自动添加到源代码中, 属性值会随着标签值的改变而改变, 当没有打开任何工程时, 该对话框中一无所有, 如图 1-14 所示。