

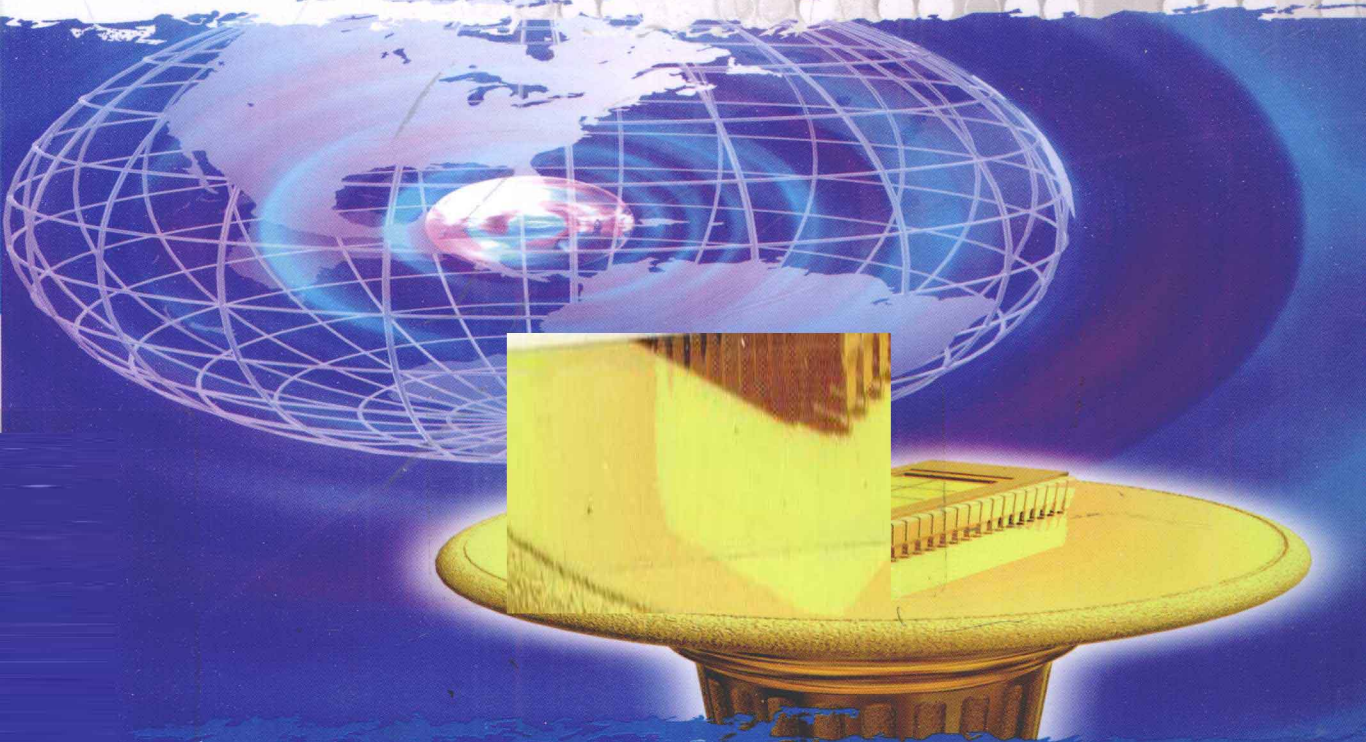


新世纪应用型高等教育
计算机类课程规划教材

ASP.NET 程序设计

新世纪应用型高等教育教材编审委员会 组编

主编 张海春



大连理工大学出版社

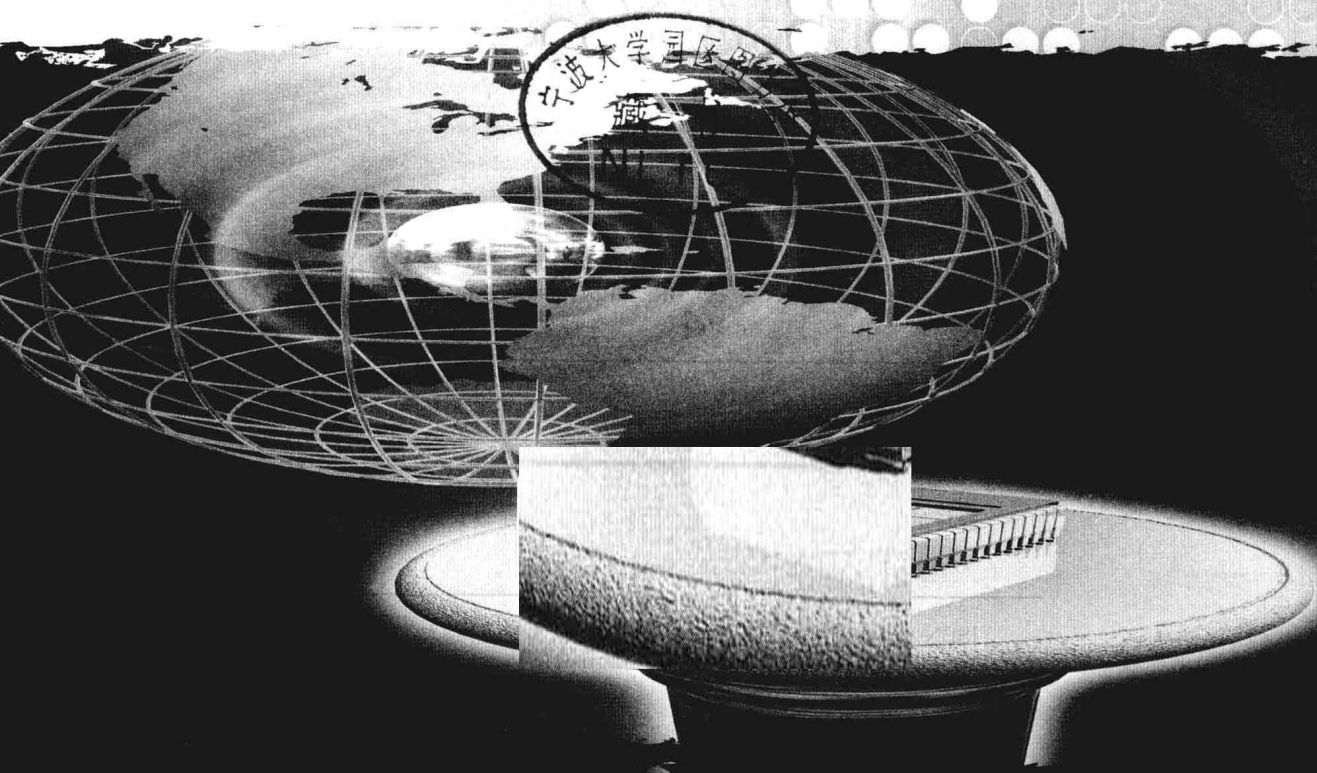


新世纪应用型高等教育
计算机类课程规划教材

ASP.NET 程序设计

新世纪应用型高等教育教材编审委员会 组编

主编 张海春
副主编 何东彬 呼洪强
朱艳红 宋宏伟



大连理工大学出版社

图书在版编目(CIP)数据

ASP.NET 程序设计 / 张海春主编. —大连: 大连理工大学出版社, 2010. 4

新世纪应用型高等教育计算机类课程规划教材

ISBN 978-7-5611-5501-1

I. ①A… II. ①张… III. ①主页制作—程序设计—高等学校—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2010)第 065138 号

大连理工大学出版社出版

地址:大连市软件园路 80 号 邮政编码:116023

发行:0411-84708842 邮购:0411-84703636 传真:0411-84701466

E-mail:dutp@dutp.cn URL:http://www.dutp.cn

大连理工印刷有限公司印刷 大连理工大学出版社发行

幅面尺寸:185mm×260mm 印张:17 字数:393 千字
印数:1~2000

2010 年 4 月第 1 版 2010 年 4 月第 1 次印刷

责任编辑:潘弘喆

责任校对:潘素君

封面设计:张莹

ISBN 978-7-5611-5501-1

定 价:32.00 元

前 言

ASP. NET 3.5 是微软公司目前推出的最先进、特征最丰富、功能最强大、且最易用的 Web 开发平台,具有方便、灵活、高效、安全、完整等特性,是当前主流的网络应用系统开发环境之一。

现代教育改革的核心是使学习者变被动型学习为投入型学习,让他们在真实的环境中学习和接受挑战性的学习任务。创设与当前学习主题、与实际相关的学习情景和任务,引导学习者带着与自己切身相关的任务进行学习,才能使抽象的理论知识直观化、形象化。“案例加项目”是全书的主线,书中的所有知识点都通过案例与项目的形式体现。本书以项目开发作为学习的总目标,以典型案例划分单元教学内容,以案例引导读者对基本知识的掌握。让读者(特别是在校学生)在教中学,在学中做,在做中理解与巩固所学的知识和技术,实现“理论与实践一体化”,提高学生的动手能力和学习积极性,使读者在尽可能短的时间内,掌握 ASP. NET 的基本开发方法,能轻松建立功能强大的动态网站。如果在学习过程中能够划分小组,并将“案例学习—知识归纳—案例实践—总结讲评”循环进行,可以更加行之有效地提高学习效果。

本教材从实用的角度出发,结合当前企业对 Web 开发人员的技能要求,详细讲解 ASP. NET 应用程序开发技术。全书共分为 10 章,第 1 章介绍了 ASP. NET 的基础知识,主要包括:Web 技术的相关概念、静态与动态网页的工作原理、ASP. NET 的工作原理。第 2 章介绍了 C# 的基本语法和 C# 面向对象程序设计。第 3 章介绍了 ASP. NET 网页代码模型、网页代码模型视图、ASP. NET 网页、ASP. NET 页面的客户端状态维护及 ASP. NET 页面生命周期等。第 4 章介绍了 HTML、HTML 服务器与 Web 服务器控件的基本概念及用法,并通过实例为读者演示了部分 ASP. NET 常用控件的使用方法。第 5 章介绍了 ASP. NET 的内置对象。第 6 章介绍了 SQL Server 2005 Express 的使用,如数据库、数据表、视图的创建,使用存储过程,数据库关系视图的建立等。第 7 章介绍了通过 ADO. NET 访问数据库及使用数据展现控件与用户进行交互的基本方法。第 8 章进一步对通过 ADO. NET 访



问数据库进行深入阐释,例如利用数据集访问数据库,熟悉和了解 DataSet、DataTable、DataColumn、DataRow 对象及 TableAdapter、ObjectDataSource 控件等,了解数据绑定功能的实现方法,并在 DropDownList、DataList 和 GridView 等数据展现控件中轻松实现数据绑定。第 9 章介绍了 ASP.NET 的容错技术与策略。第 10 章带领读者进入项目实战阶段,通过项目设计与实践来学习和巩固基础知识,并随着项目的不断展开,深入学习更多的 ASP.NET 建站技术。

本教材具有如下特点:

1. 内容新颖、结构合理

所讲内容避开晦涩难懂的理论知识,又覆盖了 Web 开发所需的各方面技术,抓住读者心理特点,循循善诱,难度适宜。

2. 结构清晰、循序渐进

本书是编者多年实践和教学工作的总结,在教材的组织和编写上,注重知识讲解的清晰与完整,采用由浅入深、前后衔接、逐步推进的方式来组织内容,并特别针对在校学生从零起步的特点,在轻松的学习过程中全面提高学生学、练、用的能力。

3. 突出实用、贴近实际

掌握必需的 ASP.NET 知识的同时,提高实际应用技能和解决问题的综合能力。书中大量的经典实例来自企业实际开发项目,一般只需以此为基础,做少量修改甚至不做修改,即可用于实际项目的开发。

4. 易于学习,配套齐全

书中精选的实践项目,业务逻辑虽然简单,但麻雀虽小,五脏俱全。通过引导读者实现最基本的功能,轻松学习和掌握建立一个三层架构的 ASP.NET 网站所需的基本技术。另外,除提供电子课件及源代码文件以外,为方便读者学习,还提供项目操作录像,可以帮助读者快速入门。

本书由张海春任主编,何东彬、呼洪强、朱艳红、宋宏伟任副主编。具体分工为:第 1 章由张海春、宋宏伟、何东彬编写,第 2、3、4、5 章由呼洪强编写,第 6、7、8、9、10 章由何东彬、朱艳红编写,全书由张海春统稿,宋宏伟审阅。李国静与陈森为本书提供了许多宝贵的意见与建议,岳秀蕾为项目操作录像的制作付出了辛勤劳动,谨此表示感谢!

尽管我们在本书的编写方面做了许多努力,但由于作者水平有限,加之时间紧迫,不当之处在所难免,恳请各位批评指正,并将意见和建议及时反馈给我们,以便下次修订时改进。

所有意见和建议请发往:gzjckfb@163.com

欢迎访问我们的网站:<http://www.dutpgz.cn>

联系电话:0411-84707492 84706104

编者

2010 年 4 月



第 1 章 ASP.NET 基础	1
1.1 Web 技术相关概念	1
1.2 WWW 的工作原理	2
1.2.1 静态网页工作原理	2
1.2.2 动态网页工作原理	3
1.3 ASP.NET 的工作原理及相关技术	3
1.4 ASP.NET 的安装、运行与调试环境	5
1.4.1 ASP.NET 的运行环境	5
1.4.2 安装 IIS 及 .NET Framework	5
1.4.3 Microsoft Visual Web Developer 2008 Express Edition	6
思考练习与上机操作	10
第 2 章 C# 编程基础	12
2.1 C# 程序	12
2.2 数据类型	13
2.3 常量和变量	15
2.3.1 常量	15
2.3.2 变量	16
2.3.3 类型转换	21
2.4 编写表达式	22
2.4.1 表达式、运算符和操作数	22
2.4.2 确定运算符的优先级	23
2.5 条件语句	24
2.5.1 if 语句	24
2.5.2 switch 语句	26
2.6 循环语句	28
2.6.1 for 语句	28
2.6.2 while 语句	30
2.6.3 do... while 语句	31
2.6.4 foreach 语句	32
2.7 C# 面向对象程序设计	34
思考练习与上机操作	37
第 3 章 ASP.NET 的网页代码模型及生命周期	40
3.1 ASP.NET 的网页代码模型	40
3.1.1 单文件页模型	40
3.1.2 代码隐藏页模型	41
3.1.3 如何选择代码模型	44
3.2 网页设计器的“源”视图	44
3.3 网页设计器的“设计”视图	45
3.3.1 在“设计”视图中定位元素	45
3.3.2 如何在“设计”视图中定位元素	46
3.4 ASP.NET 网页概述	47

3.4.1 ASP.NET 网页的组件	47
3.4.2 ASP.NET 网页介绍	47
3.5 ASP.NET 网页的客户端状态维护	52
3.5.1 视图状态	52
3.5.2 控件状态	53
3.5.3 隐藏域	53
3.5.4 Cookie	54
3.5.5 查询字符串	55
3.5.6 选择何种客户端状态维护方法	56
3.6 Page 类	56
3.6.1 Page 类的常用属性、方法和事件	57
3.6.2 Page 类的使用	58
3.7 ASP.NET 页生命周期	59
3.8 ASP.NET 网站布局	61
思考练习与上机操作	62
第 4 章 Web 窗体标准控件	64
4.1 HTML 控件、HTML 服务器控件与 Web 服务器控件的异同	64
4.1.1 HTML 服务器控件	64
4.1.2 Web 服务器控件	65
4.1.3 HTML 控件、HTML 服务器控件和 Web 服务器控件的选用	65
4.2 Web 服务器控件的共同属性	66
4.3 简单控件	66
4.3.1 标签控件(Label)	66
4.3.2 超链接控件(HyperLink)	67
4.3.3 图像控件(Image)	68
4.4 文本框控件(TextBox)	70
4.5 按钮控件(Button、LinkButton 和 ImageButton)	72
4.6 单选按钮控件和单选按钮组控件(RadioButton 和 RadioButtonList)	75
4.7 复选框控件和复选框组控件(CheckBox 和 CheckBoxList)	79
4.8 列表控件(DropDownList、ListBox 和 BulletedList)	82
思考练习与上机操作	87
第 5 章 ASP.NET 的内置对象	89
5.1 Response 对象	89
5.1.1 向浏览器发送消息	90
5.1.2 重定向	90
5.1.3 输出文本文件	90
5.2 Request 对象	91
5.2.1 获取表单数据	91
5.2.2 获取客户端浏览器的信息	93
5.2.3 获取服务器端环境变量	94
5.2.4 获取文件的路径	96
5.3 Application 对象	96
5.3.1 存取 Application 对象变量值	97
5.3.2 Application 对象的锁定与解锁	98
5.3.3 Application 对象的事件	98
5.4 Session 对象	99
5.4.1 Session 对象的常用属性	99
5.4.2 Session 对象的常用方法	101
5.4.3 Session 对象的事件	102

5.5 Server 对象	103
5.5.1 利用 HtmlEncode 和 HtmlDecode 方法对网页内容编码	104
5.5.2 URL 编码和解码	105
5.5.3 建立虚拟路径与服务器物理目录间映射	106
5.5.4 执行指定的程序	107
5.6 Cookie 对象	108
思考练习与上机操作	110
第 6 章 使用 SQL Server 2005 Express	112
6.1 SQL Server Management Studio Express 的安装与使用	112
6.2 创建数据库及数据表	115
6.3 使用存储过程	117
6.4 视图	122
6.5 数据库关系图	125
思考练习与上机操作	127
第 7 章 ADO.NET 的使用	130
7.1 ADO.NET 概述	130
7.1.1 ADO.NET 的特点与优势	130
7.1.2 ADO.NET 的结构	131
7.2 使用 ADO.NET 访问数据库的方式	132
7.2.1 创建数据库连接	132
7.2.2 数据查询与操作	133
7.2.3 Visual Web Developer 2008 数据库资源管理器	136
7.3 使用数据控件通过 ADO.NET 访问数据库	138
7.3.1 用户界面的建立	138
7.3.2 添加验证控件	142
7.3.3 主从表的设计与实现	145
思考练习与上机操作	150
第 8 章 数据访问技术	153
8.1 利用数据集访问数据库	153
8.1.1 DataSet 简介	153
8.1.2 TableAdapter 类简介	158
8.1.3 TableAdapter 类的建立与配置	160
8.2 ASP.NET 的数据绑定	166
8.2.1 数据绑定语法与实现	166
8.2.2 Repeater 控件实现数据绑定	171
8.2.3 DataList 控件实现数据绑定	175
8.2.4 GridView 控件中的数据绑定	181
思考练习与上机操作	185
第 9 章 ASP.NET 的错误处理	187
9.1 容错环节与实现策略	187
9.1.1 容错代码被忽略的原因	187
9.1.2 容错实现途径	188
9.2 防御性编码	189
9.2.1 参数检查	189
9.2.2 输入验证	190
9.3 异常处理	193
9.3.1 异常简介	193
9.3.2 捕获异常	195

9.3.3 自定义错误页面	197
思考练习与上机操作	200
第 10 章 用户留言簿系统项目实践	202
10.1 需求分析	203
10.1.1 项目定义	203
10.1.2 需求描述	203
10.1.3 用户角色分析	203
10.1.4 系统用例图	204
10.1.5 模块划分	204
10.2 系统设计	205
10.2.1 架构分析	205
10.2.2 三层架构带来的变化和好处	205
10.2.3 系统架构设计	206
10.3 数据库设计	207
10.4 系统开发与编码实现	208
10.4.1 使用网站管理工具建立用户及角色	209
10.4.2 尝试建立登录页面	212
10.4.3 母版页的建立	213
10.4.4 使用母版页建立登录页面	219
10.4.5 使用层及添加登录相关控件	221
10.4.6 添加站点地图	224
10.4.7 使用母版页建立用户注册页面	225
10.4.8 用户登录设置	227
10.4.9 使用母版页建立用户留言页面	227
10.5 安全管理中的目录权限设置	229
10.6 三层架构的建立	231
10.6.1 建立数据访问层(DAL)	231
10.6.2 建立业务逻辑层(BLL)	233
10.6.3 修改表现层的用户留言页面	236
10.7 查看用户留言列表	239
10.7.1 建立查看用户留言列表页面	239
10.7.2 利用视图进行多表查询	240
10.7.3 在数据访问层中添加对象	240
10.7.4 在业务逻辑层中添加对象	244
10.7.5 修改表现层的查看用户留言页面	245
10.8 选择用户留言内容及回复	249
10.8.1 建立查看用户留言内容及回复页面	249
10.8.2 在数据访问层中添加对象	251
10.8.3 在业务逻辑层中添加对象	252
10.8.4 使用 HiddenField 保持变量值	254
10.8.5 修改表现层的页面	255
10.9 用户管理	259
10.9.1 建立用户管理页面	259
10.9.2 在数据访问层中添加对象	260
10.9.3 在业务逻辑层中添加对象	260
10.9.4 修改表现层的页面	261
10.10 任务总结	263
10.11 项目后记	263
参考文献	264

● 本章要点

- Web 技术相关概念
- 静态与动态网页的工作原理
- ASP.NET 的工作原理
- ASP.NET 的安装、运行与调试环境

本章主要介绍开发 Web 程序所必备的一些基础知识,ASP.NET 的工作原理,以及如何搭建一个 ASP.NET 程序的运行和开发平台。

1.1 Web 技术相关概念

Web(World Wide Web,万维网)技术是电子商务的核心技术。Web 的体系结构包含下列基本元素:Web 服务器、Web 浏览器、HTTP(HyperText Transfer Protocol,超文本传输协议)、HTML(HyperText Markup Language,超文本标记语言)及 URL 等。

1. Web 服务器

由于 Web 服务器提供 HTTP 服务,因此也被称为 HTTP 服务器,它是一种管理各种 Web 文件的系统软件,当客户端浏览器提出文件请求(以 URL 形式)时,它将指定文件请求结果返回给客户端。返回结果通常包含两种形式的内容,“静态”内容和“动态”内容。

大多数情况下,Web 服务器和浏览器位于不同的机器,但也可同时存在于一台机器上。比较常见的 Web 服务器有 Apache 和 IIS(Internet Information Server,因特网信息服务器),其中 IIS 是微软公司的产品,ASP.NET 应用程序只能运行在 IIS 上。

2. Web 浏览器

Web 浏览器通过 HTTP 协议与 Web 服务器交互,并获取用户所输入的 URL 地址中指定的网页,为用户在浏览器中显示,并提供用户与网页间的交互。

3. HTTP

HTTP 是 Web 上最常用也是最重要的协议,它是 Web 服务器和浏览器之间传输 Web 页面与数据的基础。通常用户在浏览器地址栏中输入网址时不必指明 HTTP 协议,例如输入 www.dutp.com.cn 时,浏览器会自动使用 HTTP 作为默认传输协议,并形成最终的 URL 地址: http://www.dutp.com.cn 。

HTTP 是一种无状态协议。当使用这种协议的时候,所有的请求都是为了某一个特定的 Web 网页而发出的,它不知道现在的请求是第几次发出,也不知道以前发出请求的

内容和状态,当获得所请求的网页后,就自动断开与服务器的连接。

4. HTML

HTML不是一种程序设计语言,而是一种标记语言。它提供了一种将数据内容与显示形式分离开来的数据表示方法,并使得HTML文档独立于任何机器平台和处理程序。

HTML提供的链接机制是Web的本质特征之一,但它更多地是关注如何在Web页面上安排文本、图像和用户交互等,缺乏对结构化数据的表示能力。目前,使用严格规则的XHTML(可扩展超文本标记语言)是W3C计划中的HTML的接替者。虽然很多人认为它已经成为当前的HTML标准,但是它实际上是一个独立的、与HTML平行发展的标准。W3C目前的建议是使用XHTML 1.1、XHTML 1.0或者HTML 4.01进行网络出版。

5. URL

URL也被称为网页地址,用于唯一标识因特网上的资源。一个URL地址中包含四个基本元素:

- (1) 传送协议;
- (2) 服务器;
- (3) 端口号;
- (4) 路径(包含文件名在内)。

例如下面这样一个典型的URL地址: `http://www.asp.net:80/downloads/Default.aspx`。

其中:

- (1) `http`,是协议名称;
- (2) `www.asp.net`,是服务器域名;
- (3) `80`,是服务器上的网络端口号,在地址中可省略;
- (4) `downloads/Default.aspx`是路径名称(包含文件名)。

在URL地址中也可以使用其他协议,例如:HTTPS(安全的超文本传输协议)和FTP,其中FTP协议默认的端口号为21。

1.2 WWW的工作原理

1.2.1 静态网页工作原理

早期的Web服务器只支持静态网页的传送服务,虽然网页中包含文字、图片及多媒体内容,但是只要不修改设计,网页的内容不会发生变化,即不论何时、何地、何人浏览,所看到的网页信息都是一致的。对静态网页的访问过程如下:

- (1) 客户端向Web服务器提交URL请求页面;
- (2) 服务器向客户端回送被申请页面文件;
- (3) 客户端得到页面并在浏览器内显示;

(4)断开客户端与服务器的联系。

1.2.2 动态网页工作原理

从网站浏览者的角度来看,无论是动态网页还是静态网页,都可以展示基本的文字和图片信息,但从网站开发、管理和维护的角度来看就有很大的差别。动态网页一般具有如下特点:

(1)动态网页一般以数据库技术为基础,可以大大降低网站维护的工作量;

(2)采用动态网页技术可以实现更多的功能,如用户注册、用户登录、在线查询、用户管理、订单管理等;

(3)动态网页实际上并不是独立存在于服务器上的网页文件,只有当用户请求时服务器才生成并返回一个完整的 HTML 网页;

(4)动态网页中除包含标准的 HTML 标签外,还包含 Web 服务器控件以及在服务器上运行的代码。

动态网页文件的后缀一般为:.aspx、.asp、.jsp、.php、.perl、.cgi 等形式,下面以访问 ASP.NET 动态网页为例,分六个步骤来说明对动态网页的访问过程。第一步,用户提交 HTTP 请求到 IIS 服务器。第二步,由 IIS 将用户请求的 .aspx 网页文件发送到 ASP.NET 引擎。第三步,在服务器处理模块 HttpHandler(HTTP 处理程序组件)处理之前,还需要一些 HttpModule 模块的处理(这些模块都是系统默认的),这样做的好处是:在提高效率的基础上保证了安全性,并能让开发者在更多的环节上进行控制。第四步,由服务器处理模块 HttpHandler 对用户请求网页文件进行处理后返回。第五步,用户请求网页被处理后返回 IIS。第六步,用户浏览器接收到返回的网页文件,并呈现给用户浏览。如图 1-1 所示。

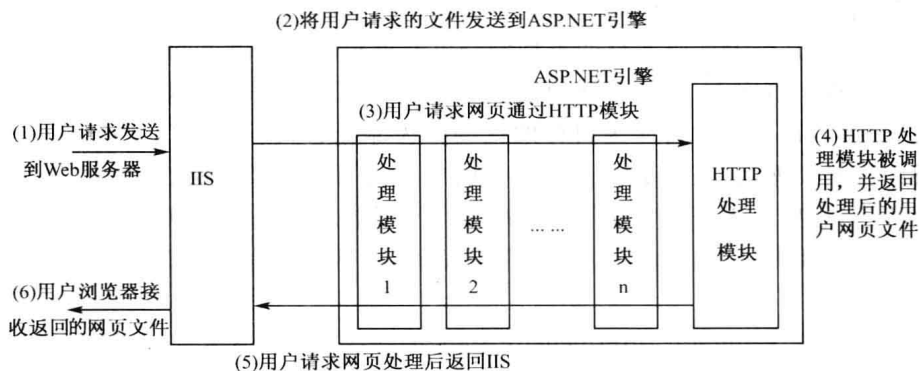


图 1-1 动态网页的访问过程

1.3 ASP.NET 的工作原理及相关技术

ASP.NET 是微软公司为了快速构建 Web 应用程序而提出的一个统一的 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。要理解 ASP.NET 的工作原理,首先要了解什么是 .NET,什么是 .NET Framework 以及什么是 CIL。

1. .NET

.NET 是 Microsoft 建立在开放体系结构基础上的面向 XML Web 服务的平台,XML Web Services 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、设备或编程语言。该平台的推出是为了简化 Web 开发,它包括如下一些核心技术:.NET Framework、.NET 企业服务器、构建模块服务及 .NET 开发工具 (Visual Studio .NET)。

其中,.NET Framework 是为 .NET 应用程序提供的全新运行环境和基础类库,所有 .NET 应用程序都必须基于 .NET Framework 才能运行。.NET 企业服务器是指那些可以集成、运行、操作管理 .NET 应用的应用系统程序,如 SQL Server 2005 等。而构建模块服务是以用户为中心的 XML Web Service 集合,它可以将对用户数据的控制由应用程序转移到用户,使得集成各种应用变得更加容易。

2. .NET Framework

.NET Framework 是 .NET 的核心组件,也是支持生成和运行下一代应用程序和 XML Web Services 的内部 Windows 组件。.NET Framework 为开发者提供了一个一致的面向对象的编程环境,而不管对象代码是在本地存储和执行,还是在本地执行但在 Internet 上分布,或者是在远程执行的;同时也提供了一个将软件部署和版本控制冲突最小化的代码执行环境,并使得开发人员的经验在面对类型不相同的应用程序(如基于 Windows 的应用程序和基于 Web 的应用程序)时保持一致。

.NET Framework 具有两个主要组件:公共语言运行库(Common Language Runtime,CLR)和 .NET Framework 类库。公共语言运行库是 .NET Framework 的基础,可以将它看做一个在执行时管理代码的代理,它提供内存管理、线程管理和远程处理等核心服务,并且还强制实施严格的类型安全以及可提高安全性和可靠性的其他形式的代码准确性检查。事实上,代码管理的概念是运行库的基本原则。以运行库为目标的代码称为托管代码,而不以运行库为目标的代码称为非托管代码。.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 reusable 类型集合,使用它可以开发多种应用程序,这些应用程序包括传统的命令行和图形用户界面(GUI)应用程序,也包括基于 ASP.NET 所提供的最新创新的应用程序(如 Web 窗体和 XML Web Services)。

3. CIL

公共中间语言(Common Intermediate Language,CIL)并不是一种可以直接执行的程序语言,它介于高级语言与机器语言之间,可读性较差,但是由于是从高级语言优化而来的,可以在计算机上高效地运行。

MSIL(Microsoft Defined Intermediate Language)是微软公司定义的中间语言,它是介于高级语言和基于 Intel 的汇编语言间的伪汇编语言。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。从理论上来说,MSIL 将消除多年以来业界不同语言之间的纷争。在 .NET 中可能出现下面的情况:一部分代码由 C# 实现,而另一部分代码使用 VB.NET 完成,但是最后这些代码都将被转换为中间语言。MSIL 给程序员提供了极大的灵活性,程序员可以选择自己熟悉的语言,并且再也不用为学习不断推出的新语言而烦恼了。

ASP.NET 是一种建立在通用语言上的程序构架,可以在 Web 服务器上建立强大的 Web 应用程序。一般可以任意选取支持 .NET 的开发语言,例如 VC++ .NET、C#、VB.NET 等来建立 ASP.NET 的 Web 应用系统。如图 1-2 所示。

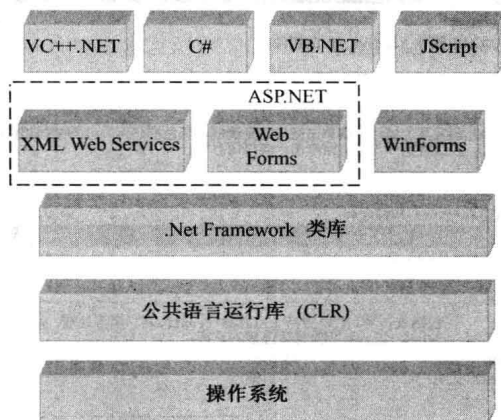


图 1-2 ASP.NET 的 Web 应用系统架构

1.4 ASP.NET 的安装、运行与调试环境

1.4.1 ASP.NET 的运行环境

在本书中,所涉及的 ASP.NET 程序运行对软硬件环境需求如下:

(1) 硬件要求

- CPU: Pentium 系列, 600 MHz 以上;
- 内存: 256 M 以上;
- 硬盘: 大于 800 M。

(2) 软件要求

- 操作系统: Windows XP、Windows 2003 Server 等;
- 浏览器: IE 6.0 及以上版本;
- .NET Framework 3.5;
- Web 服务器: IIS;
- 数据库: Microsoft Access、SQL Server 2005 等。

1.4.2 安装 IIS 及 .NET Framework

如果当前操作系统是 Windows 2003 Server, 则 IIS 已默认安装。如果使用的是

Windows XP 操作系统,则需要打开“控制面板”的“管理工具”查找“Internet 信息服务”,如果没有找到,则需要使用 Windows XP 安装光盘来添加 IIS 组件。安装步骤如下:

首先打开“控制面板”,双击“添加或删除程序”项,打开“添加或删除程序”对话框,单击对话框左侧的“添加/删除 Windows 组件”按钮,打开“Windows 组件向导”对话框,如图 1-3 所示。选中列表中的“Internet 信息服务(IIS)”复选框,并单击【下一步】按钮,将安装光盘放入光驱,按照安装提示进行安装。

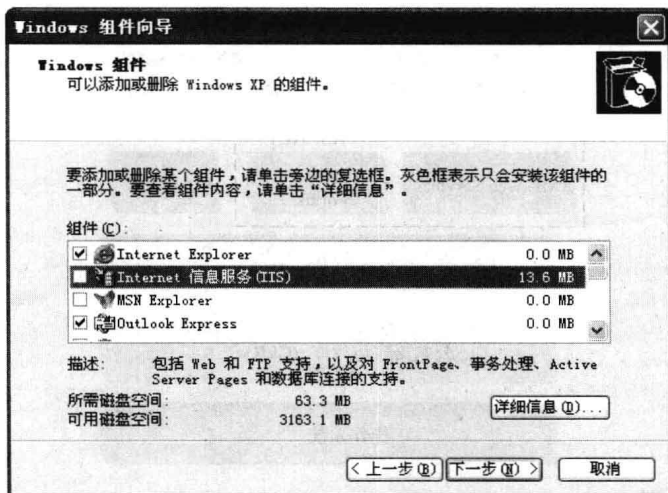


图 1-3 “Windows 组件向导”对话框

安装成功之后,重新启动 Windows,IIS 服务便会自动启动。IIS 的大部分程序都安装在 C:\WINDOWS\system32\inetrv 目录下,同时在操作系统所在磁盘驱动器下创建了一个 Inetpub 目录用于存放 Web 网页文件。

为了测试 IIS 是否能够正常工作,可以打开浏览器,并在地址栏内输入: <http://localhost/localstart.asp>,如果能够正常显示欢迎网页 localstart.asp,则表明 IIS 安装成功,可以正常运行。

为了使 ASP.NET 的 Web 系统能够正确运行,还需要安装一个名为 .NET Framework SDK 的软件包,读者可以到微软公司的网站 <http://www.microsoft.com/download> 进行下载,注意需要选择 3.5 以上版本。安装结束后,ASP.NET 的运行环境就基本搭建完毕了。

1.4.3 Microsoft Visual Web Developer 2008 Express Edition

Visual Web Developer 2008 Express Edition(以下简称为 VWD 2008)是微软公司最新开发的一款集成开发环境(IDE),它极大地简化了通过创建网页来使用 Web 服务、ASP.NET 控件和数据的全过程。使用 Visual Web Developer,可以在 .NET Framework 3.5 平台的基础上构建动态的、用数据来驱动的网页。Web 服务是 Web 2.0 的一项技术,它本质上是使用了开放式 Internet 标准的 API,使用 Visual Web Developer,几乎不用写代码,就可以完成 Web 服务的集成与创建。

为了创建可靠的、用数据来驱动的网站,数据库是至关重要的一个部分。在 VWD 2008 中,可以使用免费的 SQL Server Express 来创建数据库,同时,可以利用 ADO.NET 中的数据控件,方便地操纵数据库,用以检索、插入、更新或删除其中的信息。

VWD 2008 是微软公司面向在校学生和编程爱好者提供的一款免费 Web 集成开发工具,相对 Visual Studio 2008 庞大的身躯来说,更适合初学者,特别是学生使用。因为它的安装文件只有 910 M,运行环境对硬件的需求更低,却保留了与 Visual Studio 2008 一致的界面和必要的功能。读者可以从微软公司的下载网站 <http://www.microsoft.com/express/download> 上免费获得 Visual Web Developer 2008 Express Edition 的最新安装版本。

1. 创建新网站

网站是 Web 文件的集合,是管理 Web 程序并向外发布的基本单位,也是网站迁移、安装部署的基本单元。使用 ASP.NET 来构建 Web 应用,一个网站就是一个独立和完整的应用系统。下面将演示如何使用 VWD 2008 来创建一个 ASP.NET 的本地网站。

运行 VWD 2008,在软件启动后,单击菜单栏中的“文件”,在出现的下拉菜单中选择“新建网站”菜单命令,如图 1-4 所示。

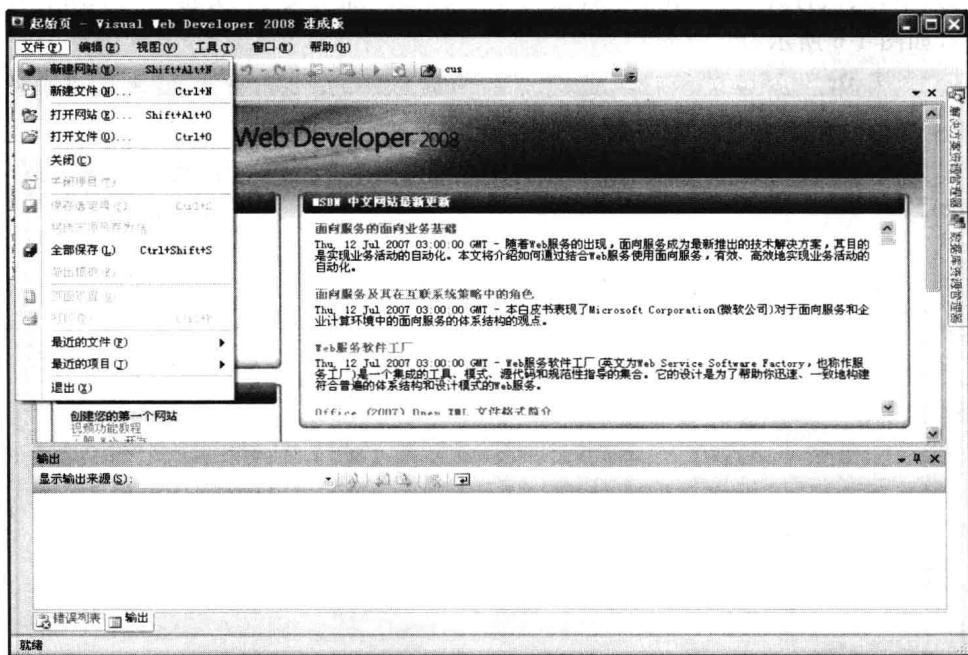


图 1-4 新建本地网站

【例 1.1】 在“新建网站”对话框中,“模板”选择“ASP.NET 网站”,“位置”选择“文件系统”,网站目录为“D:\ch1\WebSite”,“语言”选择“Visual C#”,如图 1-5 所示。

注意,在“位置”下拉列表框中包括三项:“文件系统”、“HTTP”和“FTP”。其中文件系统网站是一种在本地硬盘建立的开发者模式,便于开发和调试,却不能向外发布信息,其网站根目录可以放在任意物理目录下。使用文件系统建立网站时,并不需要在计算机上安装 IIS 服务器,在运行调试程序时,VWD 2008 将为开发者自动配置并运行一个“文

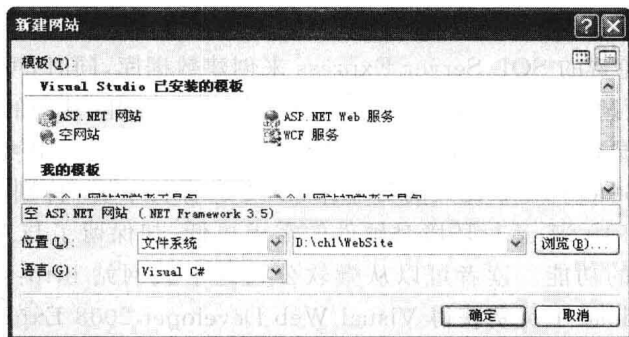


图 1-5 “新建网站”对话框

件型的轻量级开发服务器”:ASP.NET Development Server,用以模拟 IIS 对网站运行时的支持。

在图 1-5 所示“新建网站”对话框中,单击【确定】按钮,则进入 VWD 2008 的集成开发环境,在 D:\ch1\WebSite 网站根目录下,默认新建了 App_Data 文件夹(用于以后存放网站数据)、网站配置文件 Web.config 和 Default.aspx 默认主页文件。在编辑区域的“源”视图中,默认打开 Default.aspx,在<div>标签间加入文字“欢迎进入 ASP.NET 的世界!”,如图 1-6 所示。

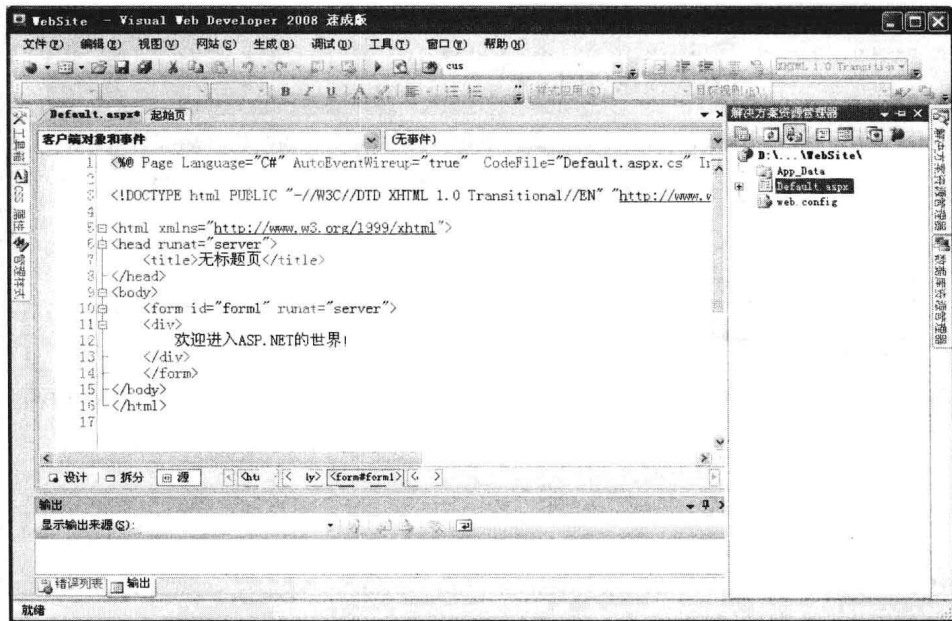


图 1-6 修改默认页

单击“调试”菜单,下拉菜单中包括下列命令:“启动调试”、“开始执行(不调试)”、“逐语句”、“逐过程”及“切换断点”。其中,“启动调试”方式可以在程序执行阶段中遇到断点或错误时暂停执行,并提供相关信息,方便开发人员分析代码和排除错误;“开始执行(不调试)”方式直接运行程序,不会因为遇到断点而停止运行;“逐语句”是指在调试模式下单语句执行;“逐过程”不进入过程而一次性执行过程体内全部代码并返回;“切换断点”