



新 编 计 算 机 类 本 科 规 划 教 材

ASP.NET

数据库网站设计教程 (C#版)

孙士保 张 瑾 张 鸣 主编
崔 淼 徐雅静 尹彦彬 等编



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

新编计算机类本科规划教材

ASP.NET数据库网站设计教程 (C#版)

孙士保 张 瑾 张 鸣 主编
崔 森 徐雅静 尹彦彬 等编

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

微软公司的 ASP.NET+C#组合是网站开发采用的主流技术之一。本教材以实际应用为目的,全面系统地介绍了使用 Visual C#语言开发 ASP.NET 3.5 数据库网站的方法及相关知识,包括 ASP.NET 基础,Web 窗体与 Page 对象,ASP.NET 的常用控件,验证控件,主题、母版页和站点导航,ASP.NET 常用对象和状态管理,Access 和 SQL Server 数据库基础,使用数据源控件访问数据库,数据绑定与数据绑定控件,使用.NET 数据提供程序访问数据库,使用 DataSet 访问数据库,使用 ASP.NET AJAX,使用 LINQ to SQL 访问数据库,数据库应用程序开发综合实训等内容。每章均有实训,供学生练习。

本教材在 Visual Studio 2008 和 SQL Server 2008 环境下讲解,除 AJAX 和 LINQ 两章外,也完全适用于 Visual Studio 2005 和 SQL Server 2005 环境。本教材概念清晰、重点突出、实例丰富,符合教师教学和学生学习习惯,非常适合课堂教学使用。

本书适合作为高等院校计算机或相关专业的教材,也可作为自学人员和网站开发人员的技术参考书。

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 数据库网站设计教程: C#版 / 孙士保, 张瑾, 张鸣主编. —北京: 电子工业出版社, 2010.3

新编计算机类本科规划教材

ISBN 978-7-121-10368-1

I. A… II. ①孙…②张…③张… III. ①主页制作—程序设计—高等学校—教材②C 语言—程序设计—高等学校—教材 IV. TP393.092 TP312

中国版本图书馆 CIP 数据核字 (2010) 第 024803 号

策划编辑: 冉 哲

责任编辑: 冉 哲

印 刷: 北京东光印刷厂

装 订: 三河市皇庄路通装订厂

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

开 本: 787×1092 1/16 印张: 26.75 字数: 680 千字

印 次: 2010 年 3 月第 1 次印刷

印 数: 4 000 册 定价: 39.00 元

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。

前 言

微软公司的 ASP.NET+C#组合是网站开发采用的主流技术之一。本教材以 Visual Studio 2008 和 SQL Server 2008 为运行环境,采用 ASP.NET 3.5 技术,以 Visual C#为开发语言,比较完整地介绍了开发 ASP.NET 数据库网站的方法和相关知识,主要内容包括:ASP.NET 基础,Web 窗体与 Page 对象,ASP.NET 的常用控件,验证控件,ASP.NET 主题、母版页和站点导航,ASP.NET 常用对象和状态管理,Access 和 SQL Server 数据库基础,使用数据源控件访问数据库,数据绑定与数据绑定控件,使用 .NET 数据提供程序访问数据库,使用 DataSet 访问数据库,使用 ASP.NET AJAX,使用 LINQ to SQL 访问数据库,数据库应用程序开发综合实训等内容。每章均配有实训,供学生练习。考虑到有些学校计算机配置较低,除 AJAX 和 LINQ 两章外,其他章节的内容完全适用于 Visual Studio 2005 和 SQL Server 2005 环境。

在学习本教材前,要求学生具有静态网站的基础知识和程序设计语言基础。本教材特别增加了一章“Access 和 SQL Server 数据库基础”,介绍了本教材所必需的数据库方面的知识。

在网站设计技术和技巧方面,本教材大量采用微软及业界推荐的方法,使本教材介绍的方法更贴近实际应用。值得一提的是,本教材中所有变量的命名,均采用业界提倡采用的 Pascal 命名法;本教材中的许多源代码都来自富有经验的程序员,有的经过简化,阅读这样的代码,有利于读者养成良好的编程习惯。本书在编写风格上,力求深入浅出,尽量将知识融于浅显的案例之中,让读者更容易掌握知识。

本教材在知识内容介绍上,采用符合认知规律的形式,先引出概念,再介绍语法格式,然后介绍方法步骤,最后给出应用实例。因为 ASP.NET 及 C#都是人工语言,我们必须按照业界及微软所采用的形式和方法、步骤来设计教材,而且在 MSDN 和相关手册中都采用这种编写形式。只有掌握了这种形式,才能很好地从 MSDN 等帮助中取得需要的知识和方法。也就是说,我们必须按照业界和 MSDN 提供的语法格式来进行“套用”。这种“套用”的方法,是学生必须掌握的。相反,有些所谓的基于工作过程或项目驱动的教材,只给出一段程序,省略了最重要的语法格式,读者只能看懂这段程序,而不知道为何要这样编写这段程序,变换一项要求就不会了。这类教材舍本而逐末,违反了认知规律。

本教材的另一个特点是合理取舍,因为受到课时的限制,课堂没有过多的时间讲授全部内容,本教材选取 ASP.NET 中应用最多的知识来介绍,舍去很少使用的内容。在学习本教材后,即便没有介绍的内容,按照本教材的思路和方法,通过查询 MSDN 等帮助,也可以很快掌握。

本教材的主要作者是具有丰富教学经验的教师和具有丰富开发经验的企业级应用程序开发工程师,他们优势互补,保证了教材的质量,使教材更贴近实际,是校企结合的结晶、范例。

本书由孙士保、张瑾、张鸣主编,崔淼、徐雅静、尹彦彬等编,其中:第 1、3、11 章由孙士保编写,第 2、7 章由张鸣编写,第 4、6 章由徐雅静编写,第 5、13 章由崔淼编写,第 8、14 章由尹彦彬编写,第 9、10 章由张瑾编写,第 12 章、附录 A 由宾春清编写,附录

B、C 由沈宇峰、王峰、臧顺娟、崔瑛瑛、李晓娟、刘克纯、孙洪玲、庄建新、彭守旺编写，教材中的许多代码由沈宇峰设计、编写，全书由孙士保统稿，刘瑞新教授主审。由于编写者水平有限，书中难免存在错误与疏漏之处，敬请读者批评指正。

本书适合作为高等院校计算机或相关专业的教材，也可作为自学人员和网站开发人员的技术参考书。

为了方便教师授课及读者学习，本书提供电子教案、源代码等，可登录华信教育资源网（<http://www.hxedu.com.cn>）免费注册下载。

作 者

目 录

第 1 章 ASP.NET 基础	1
1.1 C/S 和 B/S 架构体系	1
1.2 静态网页与动态网页	2
1.2.1 静态网页技术	2
1.2.2 动态网页技术	2
1.3 .NET Framework 简介	5
1.4 Visual Studio 简介	7
1.5 ASP.NET 网站的创建过程	7
1.5.1 新建和运行 ASP.NET 网站	7
1.5.2 打开 ASP.NET 网站	11
1.5.3 使用解决方案资源管理器	12
1.5.4 使用“工具箱”	13
1.5.5 使用“属性”窗口	14
1.5.6 使用代码和文本编辑器	15
1.5.7 子窗口的操作	17
1.5.8 保存 ASP.NET 网站	17
1.6 实训	18
第 2 章 Web 窗体与 Page 对象	20
2.1 ASP.NET Web 窗体模型	20
2.1.1 ASP.NET Web 窗体的概念	20
2.1.2 ASP.NET Web 窗体的模型	20
2.1.3 ASP.NET 网页的代码模型	21
2.2 ASP.NET 网站的组成文件	23
2.3 Page 指令	25
2.3.1 Page 指令概述	25
2.3.2 @Page 指令	26
2.4 Page 对象	27
2.4.1 ASP.NET 页的生命周期	27
2.4.2 Page 对象的常用事件	30
2.4.3 Page 对象的属性	30
2.4.4 Page 对象的常用方法	31
2.5 实训	32

第 3 章 ASP.NET 的常用控件	34
3.1 ASP.NET 控件概述	34
3.1.1 控件的类型	34
3.1.2 服务器控件基础	34
3.1.3 添加服务器控件	35
3.1.4 添加服务器控件的事件	37
3.1.5 服务器控件的基本属性	38
3.1.6 ASP.NET 页面的处理过程	39
3.2 标准控件	41
3.2.1 文本输入/输出控件	41
3.2.2 按钮控件	43
3.2.3 超链接控件	47
3.2.4 图像控件	48
3.2.5 选择控件	53
3.2.6 容器控件	66
3.3.7 其他专用控件	69
3.3 多个控件共享一个事件	83
3.4 ASP.NET 网站中资源的路径	85
3.5 实训	86
第 4 章 验证控件	95
4.1 数据验证控件概述	95
4.2 必须项验证控件	98
4.3 比较验证控件	100
4.4 范围验证控件	103
4.5 正则表达式验证控件	104
4.6 自定义验证控件	106
4.7 验证摘要控件	111
4.8 指定验证组	113
4.9 禁用验证控件	113
4.10 实训	114
第 5 章 ASP.NET 主题、母版页和站点导航	116
5.1 ASP.NET 主题和外观	116
5.1.1 概述	116
5.1.2 定义 ASP.NET 页面主题	118
5.1.3 应用 ASP.NET 主题	121
5.1.4 禁用 ASP.NET 主题	124
5.2 ASP.NET 母版页	124

5.2.1	概述	124
5.2.2	ASP.NET 母版页的创建	126
5.3	ASP.NET 站点导航	131
5.3.1	概述	131
5.3.2	ASP.NET 站点地图	132
5.3.3	SiteMapPath 控件	133
5.3.4	SiteMapDataSource 控件	136
5.3.5	TreeView 控件	136
5.3.6	Menu 控件	137
5.4	实训	140
第 6 章	ASP.NET 常用对象和状态管理	148
6.1	ASP.NET 常用对象	148
6.1.1	Response 对象	148
6.1.2	Request 对象	151
6.1.3	Server 对象	158
6.2	ASP.NET 的状态管理	164
6.2.1	使用 Cookie 实现状态管理	164
6.2.2	使用 Session 对象	168
6.2.3	使用 Application 对象共享数据	173
6.3	实训	178
第 7 章	Access 和 SQL Server 数据库基础	181
7.1	关系型数据库	181
7.2	常用数据库产品简介	182
7.3	Access 2003 使用基础	183
7.3.1	创建 Access 数据库	183
7.3.2	Access 表	184
7.4	SQL Server 2008 使用基础	192
7.4.1	创建 SQL Server 数据库	192
7.4.2	表的操作	197
7.4.3	记录的操作	200
7.4.4	查询的操作	204
7.4.5	存储过程的操作	207
7.4.6	数据库的分离和附加	214
7.5	在 Visual Studio 中创建 SQL Server 数据库	216
7.6	实训	220
第 8 章	使用数据源控件访问数据库	221
8.1	ADO.NET 简介	221

8.2	数据控件概述	223
8.2.1	数据源控件概述	223
8.2.2	数据绑定控件概述	224
8.3	使用 SqlDataSource 控件连接数据库	224
8.3.1	SqlDataSource Web 服务器控件概述	224
8.3.2	将 SqlDataSource 控件连接至数据源	225
8.3.3	使用 SqlDataSource 控件选择数据	229
8.3.4	使用 SqlDataSource 控件修改数据	235
8.4	实训	237
第 9 章	数据绑定与数据绑定控件	238
9.1	数据绑定	238
9.1.1	数据绑定概述	238
9.1.2	简单数据绑定和复杂数据绑定	239
9.1.3	数据绑定控件概述	239
9.2	GridView 控件	240
9.2.1	GridView 控件简介	240
9.2.2	使用 GridView 控件显示数据	240
9.2.3	启用 GridView 控件的默认功能	244
9.2.4	GridView 控件的列	247
9.3	DetailsView 控件	251
9.3.1	DetailsView 控件简介和语法	251
9.3.2	使用 DetailsView 控件编辑、插入、删除记录	252
9.3.3	在主/详细方案中使用 DetailsView 控件显示详细记录	253
9.4	FormView 控件	255
9.4.1	FormView 控件简介和语法	255
9.4.2	使用 FormView 控件显示数据	256
9.4.3	使用 FormView 控件编辑数据	259
9.5	使用数据绑定表达式实现数据绑定	260
9.5.1	数据绑定语法	260
9.5.2	使用 Eval 方法	261
9.5.3	使用 Bind 方法	262
9.6	调用 DataBind()方法实现数据绑定	263
9.7	实训	264
第 10 章	使用 .NET 数据提供程序访问数据库	269
10.1	.NET 数据提供程序概述	269
10.1.1	.NET 数据提供程序简介	269
10.1.2	使用 .NET 对象访问数据库的步骤	270

10.2	数据库的连接字符串	270
10.2.1	数据库连接字符串的常用参数	270
10.2.2	连接到 SQL Server 的连接字符串	271
10.2.3	连接到 OLE DB 数据源的连接字符串	272
10.2.4	连接字符串的存放位置	272
10.3	连接数据库——Connection 对象	274
10.3.1	Connection 对象概述	274
10.3.2	创建 Connection 对象	275
10.3.3	Connection 对象的属性和方法	275
10.3.4	连接到数据库的基本步骤	277
10.3.5	关闭连接	278
10.4	执行数据库命令——Command 对象	278
10.4.1	Command 对象概述	278
10.4.2	创建 Command 对象	279
10.4.3	Command 对象的属性和方法	280
10.5	读取数据——DataReader 对象	286
10.5.1	DataReader 对象概述	286
10.5.2	创建 DataReader 对象	287
10.5.3	DataReader 对象的属性和方法	287
10.5.4	关闭 DataReader 对象	291
10.6	数据适配器——DataAdapter 对象	291
10.6.1	DataAdapter 对象概述	291
10.6.2	创建 SqlDataAdapter 对象	292
10.6.3	DataAdapter 对象的属性和方法	292
10.7	实训	293
第 11 章	使用 DataSet 访问数据库	300
11.1	DataSet 的基本构成	300
11.1.1	DataSet、DataAdapter 和数据源之间的关系	300
11.1.2	DataSet 的组成结构和工作过程	301
11.1.3	DataSet 中的常用子对象	302
11.1.4	DataSet 对象常用属性和方法	302
11.1.5	构建 DataSet 的方法	303
11.2	访问和填充 DataSet	305
11.2.1	创建 DataSet	305
11.2.2	填充 DataSet	305
11.3	修改 DataSet 及数据更新	307
11.3.1	添加新记录	307
11.3.2	修改数据库记录	309

11.3.3 删除记录	310
11.4 实训	310
第 12 章 使用 ASP.NET AJAX	321
12.1 Ajax 和 ASP.NET AJAX	321
12.1.1 Ajax 概述	321
12.1.2 了解 ASP.NET AJAX	322
12.2 ScriptManager 控件	323
12.2.1 ScriptManager 的主要属性和方法	323
12.2.2 注册客户端脚本	324
12.3 UpdatePanel 控件	325
12.3.1 UpdatePanel 控件的常用属性	325
12.3.2 实现页面局部更新	326
12.4 UpdateProgress 控件	330
12.4.1 UpdateProgress 控件的常用属性	330
12.4.2 使用 UpdateProgress 控件	331
12.5 Timer 控件	333
12.5.1 Timer 控件的常用属性和事件	333
12.5.2 使用 Timer 控件	334
12.6 ASP.NET AJAX 控件工具包	335
12.6.1 安装 ASP.NET AJAX 控件工具包	335
12.6.2 使用 ConfirmButtonExtender 控件弹出确认对话框	337
12.6.3 使用 FilteredTextExtender 实现输入字符过滤	339
12.6.4 使用 SlideShowExtender 制作图片播放器	341
12.7 实训	343
第 13 章 使用 LINQ to SQL 访问数据库	345
13.1 LINQ 简介	345
13.1.1 LINQ 的组成结构	345
13.1.2 与 LINQ 相关的几个概念	346
13.2 使用 LinqDataSource 访问数据库	350
13.2.1 创建 DataContext 类	350
13.2.2 配置 LinqDataSource 控件	353
13.2.3 配合 GridView 控件访问数据库	354
13.3 使用 LINQ to SQL 查询数据	357
13.3.1 返回数据表中的全部记录 (Select 语句)	357
13.3.2 返回数据表中符合条件的部分记录 (Where 语句)	359
13.3.3 返回数据集合的排序 (OrderBy 方法)	360
13.3.4 连接不同数据表 (Join 语句)	361

13.4	使用 LINQ to SQL 操作数据库	363
13.4.1	插入新记录	363
13.4.2	修改记录	365
13.4.3	删除记录	367
13.4.4	使用 LINQ to SQL 直接执行 SQL 语句	368
13.5	实训	369
第 14 章	数据库应用程序开发综合实训——设计一个简易留言板程序	371
14.1	实训目的和要求	371
14.2	程序功能的实现	374
14.2.1	用户登录功能的实现	374
14.2.2	查看和发布留言功能的实现	377
14.2.3	新用户注册功能的实现	379
14.2.4	重设被遗忘的密码功能的实现	382
14.2.5	用户管理功能的实现	385
14.3	实训	388
附录 A	SQL Server 2008 的安装与配置	389
附录 B	Visual Studio 2008 的安装与配置	403
附录 C	变量的命名规范	414

第 1 章 ASP.NET 基础

本章内容: C/S 和 B/S 架构体系, 静态网页与动态网页, .NET Framework 简介, Visual Studio 简介, ASP.NET 网站的创建过程。

本章重点: ASP.NET 网站的创建过程。

1.1 C/S 和 B/S 架构体系

目前在程序开发领域中, 主要有两大编程体系: 一是基于操作系统平台的 C/S 结构, 二是基于浏览器的 B/S 结构。

1. C/S 架构体系

在 2000 年以前, C/S (Client/Server, 客户机-服务器) 架构占据着开发领域的主流地位, 其结构如图 1-1 所示。通常, 程序员将开发完成的软件安装在某台微机(客户机)中, 将数据库安装在专用的服务器(数据库服务器)中, 这样可以利用两端的硬件资源, 将任务合理分配到客户端和服务端, 降低了系统的通信开销。这种架构要求客户机中必须安装客户端程序, 否则无法工作。再有, 在 C/S 架构中, 主要的数据分析处理工作需要在客户机中完成, 这就要求客户机有较高的硬件配置。C/S 架构的应用程序有: QQ、MSN、Foxmail、Outlook、浏览器及一些网络游戏等。

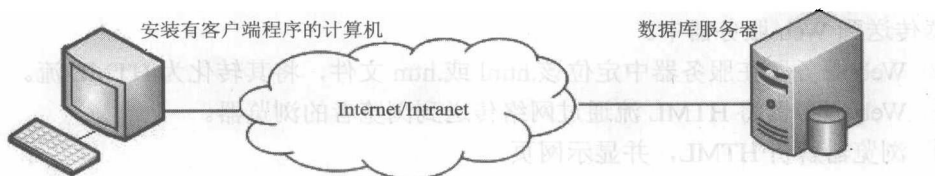


图 1-1 C/S 架构体系

2. B/S 架构体系

B/S (Browser/Server, 浏览器/服务器) 架构体系如图 1-2 所示, 由客户机、Web 服务器和数据库服务器三部分组成。在中小型应用系统中, Web 服务器可以与数据库服务器安装在同一台服务器中。与 C/S 架构相比, 它不需要在客户机上安装专门的客户端软件, 用户在使用程序时仅需要通过安装在客户机上的浏览器访问指定的 Web 服务器即可。目前, 绝大多数微机都在使用集成了 Internet Explorer (IE 浏览器) 的 Windows 操作系统, 也就是说, 只要客户机能够通过网络访问指定的 Web 服务器, 即可使用 B/S 架构的应用程序。此外, 在 B/S 架构中, 主要的数据分析处理工作是在应用服务器中完成的, 客户端主要用来下达指令和接收结果, 所以客户机的配置要求不高。B/S 架构非常适合“瘦客户端”的运行环境。

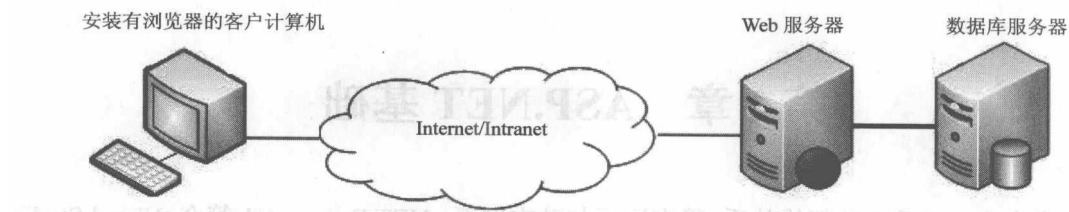


图 1-2 B/S 架构体系

1.2 静态网页与动态网页

随着 Internet 技术的发展，基于 Internet 的 Web 应用程序开发已经成为当今软件技术发展最快的应用领域，B/S 结构的应用程序已经成为应用软件的开发主流。

1.2.1 静态网页技术

在动态网页技术出现之前，所有的网页都是静态的。静态网页是指由网页编写者用纯 HTML 代码编写的网页，以.html 或.htm 文件的形式保存。静态网页制作完成并发布后，网页的内容（文本、图像、声音、超链接等）和外观是保持不变的，无论哪个浏览者，在何时，以何种方式访问这个网页，它的外观总是保持不变。静态网页中不包含任何客户交互的动态内容，其优点是访问效率很高，网站的开发和架设相当容易。目前仍然有很多网站使用静态网页技术。

静态网页技术的工作过程（如图 1-3 所示）如下。

① 浏览者在浏览器地址栏中输入 HTTP 请求或链接到该网页地址，该请求通过网络从浏览器传送到 Web 服务器中。

② Web 服务器在服务器中定位该.html 或.htm 文件，将其转化为 HTML 流。

③ Web 服务器将 HTML 流通过网络传送到浏览者的浏览器。

④ 浏览器解析 HTML，并显示网页。



图 1-3 静态网页技术的工作过程

静态网页的主要缺点是，当网页中的内容需要改变时，必须重新制作网页，所以静态网页不适合需要频繁改变内容的网页。

1.2.2 动态网页技术

动态网页技术主要分为两种：客户端动态网页技术和服务器端动态网页技术。

1. 客户端动态网页技术

客户端动态网页技术是指 Web 服务器把原始的 HTML 页面及一组包含了页面逻辑的脚本、组件等一起发送到客户端，这些脚本和组件包含了如何与浏览器交互并产生动态内容的指令，由客户端的浏览器及其插件解析 HTML 页面并执行这些指令。典型的客户端动态网页技术包括 JavaScript、VBScript、ActiveX 控件、Java Applet、Ajax 等。

客户端动态网页由网页制作者用 HTML 语言编写，并将其以.html 或.htm 文件的形式保存。同时用其他语言编写指令，这些指令嵌入 HTML 语言中，或者以单独的文件保存。

客户端动态网页技术的工作过程（如图 1-4 所示）如下。

- ① 用户在客户端浏览器中输入一个 HTTP 网页请求，通过网络传送到 Web 服务器。
- ② Web 服务器在服务器中定位该.html 或.htm 文件，以及 HTML 文件指令中包含的其他文件，将其转化为 HTML 流。
- ③ Web 服务器将 HTML 流和其他指令，通过网络传送到浏览者的浏览器。
- ④ 浏览器插件解析指令，并将其转换为 HTML 文件。
- ⑤ 浏览器解析 HTML，显示网页。

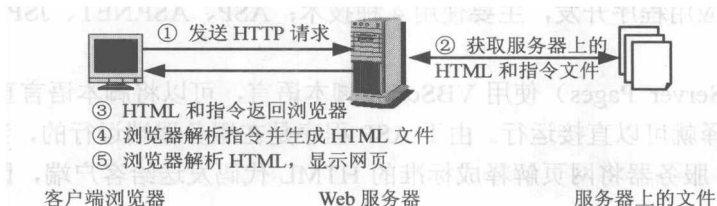


图 1-4 客户端动态网页技术的工作过程

客户端动态网页技术的主要优点是，充分利用了客户端的计算机资源，减轻了服务器和网络上的计算机压力，同时可以方便地实现基于图形的用户交互界面。然而，客户端动态网页技术需要把语言脚本和组件下载到客户端的计算机中，如果脚本或者组件较大，下载速度就会变慢。其次，现在的每种客户端浏览器可能存在兼容问题，不能完整地解析代码。还有，脚本和组件下载到浏览者的计算机中后，源代码不便于保密。另外，有些脚本和组件可能含有恶意代码。所以，客户端网页技术在 Web 应用程序上的应用一般局限在显示动画、验证用户输入等方面。

2. 服务器端动态网页技术

服务器端动态网页技术是指在 Web 服务器端根据客户端浏览器的不同请求，动态地生成相应的内容，然后发送给客户端浏览器。

服务器端动态网页技术的工作过程（如图 1-5 所示）如下。

- ① 用户在客户端浏览器中输入一个 HTTP 网页请求，通过网络传送到 Web 服务器中。
- ② Web 服务器在服务器中定位指令文件。
- ③ Web 服务器根据指令生成 HTML 流。
- ④ Web 服务器将生成的 HTML 流通过网络传送到浏览者的浏览器中。
- ⑤ 浏览器解析 HTML，显示网页。

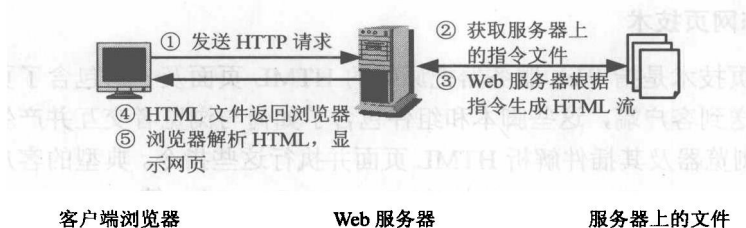


图 1-5 客户端动态网页技术的工作过程

使用服务器端动态网页技术，所有指令都先在服务器中进行处理，并根据不同浏览者的请求生成不同的 HTML 静态网页，然后把静态网页传送到客户端的浏览器中，再由浏览器解析并显示出来。服务器端动态网页技术把原始页面代码始终隐藏在服务器中，浏览者无法看到原始代码，起到了保密作用。缺点是，由于页面是在浏览者请求时临时生成的，因此，首次显示网页时速度较慢。

3. B/S 架构编程技术

目前 B/S 架构应用程序开发，主要使用 4 种技术：ASP、ASP.NET、JSP 和 PHP。

(1) ASP

ASP (Active Server Pages) 使用 VBScript 脚本语言，可以将脚本语言直接嵌入 HTML 文档中，不需要编译就可以直接运行。由于 ASP 程序是在服务器端运行的，当客户端浏览器访问 ASP 网页时，服务器将网页解释成标准的 HTML 代码发送给客户端，因此，不存在浏览器兼容的问题。

但是，每当客户端打开一个 ASP 页面时，服务器都会将该 ASP 程序解释一遍，最后生成标准的 HTML 代码发送到客户端，从而影响了 ASP 程序的运行速度。再有，ASP 不能运行在 Linux、FreeBSD 等操作系统中。在一般情况下，用 ASP 开发的程序只能运行在 Windows 操作系统的 IIS 环境中。目前新开发的系统已经不再使用 ASP。

(2) ASP.NET

ASP.NET 是一种用于创建动态 Web 页的强大的服务器端新技术，它可为 WWW 站点或企业内部互联网创建动态的可进行交互的 HTML 页面。

ASP.NET 是微软 .NET 体系结构的一部分，并不是 ASP 的升级版本，所以在学习 ASP.NET 前并不需要先学习 ASP。ASP.NET 的主要优点如下。

① 使用 .NET 提供的所有类库，全面支持面向对象的程序设计，可以实现以往 ASP 所不能实现的许多功能。

② 引入了服务器端控件的概念，这样使开发交互式网站更加方便。

③ 引入了 ADO.NET 数据访问接口，大大提高了数据库访问效率。

④ 在 Visual Studio 可视化开发环境中创建 ASP.NET 应用程序，可以采用 C#、Visual Basic、C++、J# 等语言，进一步提高了编程效率。

⑤ 因为 ASP.NET 应用程序的核心部分在发布到 IIS 网站前就已被编译成 .dll 文件，所以执行速度更快。但是，ASP.NET 目前只能运行在 Windows 操作系统的 IIS 环境下。

(3) JSP

JSP 页面由 HTML 代码和嵌入其中的 Java 代码组成，具有良好的跨平台性。在页面被客

户端请求时，服务器对其中的 Java 代码进行处理，然后将生成的标准 HTML 页面发送到客户端。与 ASP 不同的是，JSP 页面第一次被访问时，服务器将 JSP 编译成二进制代码，并保存起来，以后当客户机再次访问该页面时，这些二进制代码将被直接调用，所以 JSP 较 ASP 具有更高的执行效率。

(4) PHP

PHP 程序最初是用 Perl 语言编写的简单程序，后来经其他程序员不断完善，于 1997 年发布了功能基本完善的 PHP3。PHP 程序可以运行在 UNIX、Linux 和 Windows 操作系统中，对客户端浏览器也没有特殊的要求。PHP 也是将脚本语言嵌入到 HTML 文档中，它大量采用了 C、Java 和 Perl 语言的语法，并加入了 PHP 自己的特征。

PHP 在 1999—2000 年期间应用较为普遍，由 Linux + PHP + MySQL 构成的完全开源而且非常稳定的应用平台曾经风靡一时，但由于 PHP 语言更新较慢再加上没有很好的技术支持，目前 PHP 正在逐步退出 B/S 架构的开发领域。

1.3 .NET Framework 简介

Microsoft .NET Framework（简称.NET）是一种新的开发平台，是美国微软公司为适应 Internet 发展的需要而推出的特别适合网络编程和网络服务开发的平台。对于软件开发人员来说，.NET 是继 DOS 开发平台（如 BASIC、FORTRAN、PASCAL 等）、Windows 开发平台（Visual Basic、Visual FoxPro 等）之后，以计算机网络为背景的新一代软件开发平台。

1. .NET Framework 结构

Microsoft .NET Framework 是一个用于 Windows 应用程序、Web 应用程序、控制台应用程序和智能设备应用程序的平台。.NET Framework 还提供丰富的类库，程序员可以使用类库来减少需要编写、测试和维护的代码量。

.NET 技术的核心是 .NET Framework，它是构建于计算机网络基础上的开发工具。.NET Framework 的基本结构如图 1-6 所示。

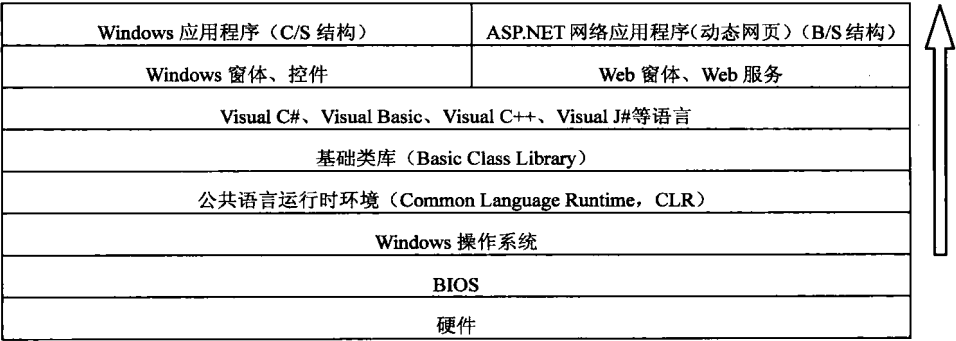


图 1-6 .NET Framework 结构

从图 1-6 中可以看出，.NET Framework 的最上层是开发完成的应用程序，分为基于 ASP.NET 的 Web 应用程序和基于 Windows 系统的应用程序。前者由 Web 窗体和 Web 服务（Web Service）组成，用户通过浏览器访问存放在服务器上的应用程序；后者由窗体和控件