

ASP.NET动态网站 开发教程(第二版)

- ◆ 网页设计基础知识
- ◆ Visual Studio 2008开发动态网页的流程
- ◆ 主要的HTML标记的使用方法
- ◆ C#语言基础
- ◆ 主要的服务器控件
- ◆ 常用的内置对象
- ◆ 数据源和数据绑定控件
- ◆ ASP.NET AJAX控件



胡静 韩英杰 陶永才 编著



清华大学出版社

高等学校计算机应用规划教材

ASP.NET 动态网站开发教程

(第二版)

胡静 韩英杰 陶永才 编著

清华大学出版社

北 京

内 容 简 介

本书从初学者的角度出发,以通俗易懂的语言,丰富多彩的实例,详细介绍了使用 ASP.NET 3.5 进行 Web 程序开发应该掌握的主要技术。全书共分 10 章,主要内容包括 ASP.NET 3.5 概述, Visual Studio 2008 集成开发环境, HTML、XHTML 网页设计基础和 C# 2008 程序设计基础,使用 ASP.NET 编写网页的基础知识和相关的服务器控件,常用内置对象、数据源和数据绑定控件,以及 ASP.NET 3.5 中新增的 AJAX 控件及 LINQ 技术。

本书注重基础、讲究实用,内容丰富、结构合理、思路清晰、示例翔实,特别适合 Web 开发入门者。可作为高等院校计算机及相关专业的教材,也可供工程师和想利用 Visual Studio 2008 开发平台开发 Web 应用程序的人员参考阅读,并可作为各类培训班的培训教材。

本书的电子教案、实例源代码和习题答案可以到 <http://www.tupwk.com.cn/downpage/index.asp> 网站下载。

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 动态网站开发教程(第二版)/胡静,韩英杰,陶永才 编著. —北京:清华大学出版社,2009.10
(高等学校计算机应用规划教材)

ISBN 978-7-302-21158-7

I. A… II. ①胡… ②韩… ③陶… 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 174919 号

责任编辑:胡辰浩(huchenhao@263.net) 袁建华

装帧设计:孔祥丰

责任校对:成凤进

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:北京密云胶印厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×260 印 张:20.5 字 数:512 千字

版 次:2009 年 10 月第 2 版 印 次:2009 年 10 月第 1 次印刷

印 数:1~5000

定 价:30.00 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:031109-01

前 言

随着网络技术的飞速发展,人类的信息资源实现了高度共享,从根本上改变了人类进行信息交流的方式,展开了一场史无前例的信息革命。越来越多的人习惯从网上搜索自己需要的资料,越来越多的企业将应用系统发布成网站,供自己的用户实现快捷、方便地业务处理。浏览器/服务器(B/S)结构的应用程序随着用户的这种需求而被提升到更高的地位。

在实现 B/S 结构的技术中,最具代表性的就是.NET 框架下的 ASP.NET 技术和 J2EE 框架下的 JSP 技术。如今,随着 ASP.NET 技术的方便性逐渐提高,已经有越来越多的开发人员转入.NET 开发阵营,从而在这个技术领域内的初学者和急需提高的人员数量不断增加。2008 年 2 月,微软公司发布了 ASP.NET 3.5。为了使初学者和 ASP.NET 技术人员快速完整地了解 ASP.NET 3.5 的技术特性,本书从基础到提高,由浅入深地介绍了相关知识,使读者能够全面、轻松、深刻地了解书中介绍的技术。

目前市面上有不少介绍 ASP.NET 的图书,但是要找一本适合初学者的图书也不容易。有些图书起点太高,初学者难以理解基本概念,学习起来困难重重,容易产生厌倦心理而放弃学习;有的图书又过于简单,读者在学完之后还是不会做任何实际的事情,不能达到一定的高度。

概括起来,本书具有以下主要特点:

- ◆ 注重基础,讲究实用,力求从入门到精通。
- ◆ 充分体现案例教学。本书以易学易用为重点,精选大量实用的示例、知识丰富、步骤详细、学习效率高,特别适合入门者。
- ◆ 配有源代码,方便上机实践。本书的所有示例均在 Visual Studio 2008 开发环境下调试通过,读者可以直接下载所有例子的源程序,并通过书中介绍的步骤学习开发要点。

本书共分 10 章,各章的主要内容如下:

第 1 章简要介绍了 HTTP 协议、静态网页和动态网页等 Web 基础知识,并介绍了 ASP.NET 3.5 的发展历史以及主要特点,最后讲解了 Visual Studio 2008 的安装方法和开发 ASP.NET 应用程序的一般步骤。通过这些介绍使读者对 ASP.NET 有一个整体的了解,为以后章节的学习打下基础。

第 2 章主要介绍了超文本标记语言 HTML 的显示原理以及常用标记,这是进行页面设计的基础,最后比较了 XHTML 和 HTML 的不同。

第 3 章介绍了编写 ASP.NET 网页所用的语言 C# 的基础知识。在进行 ASP.NET 程序设计之前,首先要选择一种编程语言,而最合适的就是 C#。本章的内容包括 C# 2008 的数据类型、运算符、控制语句、类和对象等内容,最后一节介绍了 C# 2008 的新特性。通过

本章的学习,读者可以在后面的章节中使用 C#语言来设计程序。

第 4 章介绍了如何利用 ASP.NET 建立 Web 页面,及创建 ASP.NET Web 页面所需的基础知识。包括 ASPX 网页的基类、页面是如何组织和运行的、ASP.NET 的网页代码模型以及状态管理和配置管理的相关知识,对读者理解 ASP.NET 的工作模式非常重要。

第 5 章介绍了 HTML 服务器控件、Web 服务器控件和验证控件的使用方法,利用这些控件可以大大提高程序开发效率,是开发 ASP.NET 应用程序的基础。

第 6 章介绍了 ASP.NET 中常用的内置对象,包括 Request、Response、Session、Application 和 Server 的主要方法和属性,并讲解了 Cookie 对象的使用方法。熟练掌握这些内置对象,可以开发出功能强大的应用程序。

第 7 章介绍了使用 ADO.NET 进行数据库访问的方法。主要包括 ADO.NET 的数据提供者(Data Provider)和数据集(DataSet)的基础知识等。

第 8 章介绍了数据绑定技术、ASP.NET 3.5 提供的各种数据源控件和使用数据源控件连接到各种数据源的方法以及复杂数据绑定控件的功能和使用方法。

第 9 章介绍了 Ajax 的基础知识以及 ASP.NET AJAX 控件,这是微软的客户端异步无刷新页面技术,在新的 ASP.NET 3.5 中,已经包含了此技术框架。

第 10 章用一个综合的例子将有关知识贯穿在一起。

本书由胡静、韩英杰、石磊编著整理,在本书编写过程中,石磊教授给予了大力支持和帮助,同时胡连春、王文秀、侯垚森、景三东、何增辉、王战红、史海振、扬仕飞、刘杨、张艳等付出了辛勤的劳动,在此一并向他们表示诚挚的感谢。

本书从 ASP.NET 基础知识讲起,语言通俗易懂,并配有大量实例和插图,使读者对每一章所讲述的内容都能有深刻的理解并加以巩固,十分适合初学者和有一定 ASP.NET 基础的人员使用。

在编写本书的过程中参考了许多相关文献,在此向这些文献的作者深表感谢。由于时间较紧,书中难免有错误与不足之处,恳请专家和广大读者批评指正。我们的信箱是 huchenhao@263.net, 电话: 010-62796045。

作 者
2009 年 8

目 录

第 1 章 ASP.NET 3.5 概述	1	2.1.5 HTML 文档的结构	22
1.1 Web 基础知识	1	2.2 设置文字	24
1.1.1 HTTP 协议	1	2.2.1 段落标记<p>和换行 标记 	24
1.1.2 Web 服务器和浏览器	2	2.2.2 文字修饰标记	25
1.1.3 C/S 模式与 B/S 模式	2	2.2.3 注释标记	26
1.1.4 Web 的访问原理	2	2.3 超链接	26
1.1.5 常见的网页程序设计技术	4	2.3.1 什么是 URL	27
1.2 ASP.NET 3.5 简介	6	2.3.2 超链接的路径	27
1.2.1 .NET 框架	6	2.3.3 超链接标记<a>	27
1.2.2 ASP.NET 3.5 概述	9	2.4 插入图像	29
1.2.3 ASP.NET 开发工具	10	2.4.1 图像的格式	29
1.3 安装 Visual Studio 2008	10	2.4.2 图像标记	29
1.3.1 安装 Visual Studio 2008	11	2.5 表格	31
1.3.2 主窗口	13	2.5.1 基本的表格标记	31
1.3.3 文档窗口	13	2.5.2 设置表格的属性	31
1.3.4 工具箱	15	2.5.3 设置跨行、跨列的表格	33
1.3.5 解决方案资源管理器	16	2.6 表单	34
1.3.6 属性窗口	16	2.6.1 表单标记<form>	35
1.3.7 错误列表窗口	16	2.6.2 <input>标记	36
1.4 开发 ASP.NET Web 应用程序 的一般流程	17	2.6.3 <select>标记	37
1.4.1 创建 ASP.NET 应用程序	17	2.6.4 <textarea>标记	37
1.4.2 运行 ASP.NET 应用程序	18	2.7 XHTML	39
1.5 本章小结	18	2.7.1 XHTML 简介	39
1.6 思考和练习	18	2.7.2 XHTML 文档的结构	40
第 2 章 HTML 和 XHTML	20	2.7.3 XHTML 与 HTML 之间 的语法差异	41
2.1 HTML 语言简介	20	2.8 本章小结	42
2.1.1 Web 标准	20	2.9 思考和练习	42
2.1.2 HTML 的起源和演变	21	第 3 章 C# 2008 基础	45
2.1.3 HTML 语言的特点	21	3.1 C#语言简介	45
2.1.4 HTML 标记基础	21		

3.2 数据类型.....46	3.8.5 自动属性.....97
3.2.1 值类型.....46	3.9 本章小结.....98
3.2.2 引用类型.....51	3.10 思考和练习.....98
3.2.3 语句.....53	
3.2.4 标识符与关键字.....54	第4章 用ASP.NET建立Web页面.....103
3.2.5 常量和变量.....55	4.1 ASPX 网页的基类.....103
3.2.6 装箱和拆箱.....57	4.2 页面管理.....104
3.3 运算符.....57	4.2.1 页面的往返与处理机制.....105
3.3.1 算术运算符.....58	4.2.2 页面的生命周期.....105
3.3.2 赋值运算符.....58	4.2.3 ASP.NET 页生命周期事件.....106
3.3.3 关系运算符.....59	4.2.4 ASP.NET 页面指令.....107
3.3.4 逻辑运算符.....59	4.3 ASP.NET 的网页代码模型.....108
3.3.5 条件运算符.....60	4.3.1 单文件页模型.....108
3.3.6 对象创建运算符.....60	4.3.2 代码隐藏页模型.....110
3.3.7 运算符的优先级.....61	4.3.3 用 Visual Studio 2008 创建 ASP.NET 网页.....113
3.4 控制语句.....62	4.4 状态管理.....115
3.4.1 分支语句.....62	4.4.1 页面状态概述.....115
3.4.2 循环语句.....64	4.4.2 视图状态.....115
3.4.3 跳转语句.....68	4.4.3 控件状态.....117
3.4.4 异常处理语句.....69	4.4.4 隐藏域.....118
3.5 字符串处理.....75	4.5 ASP.NET 配置管理.....118
3.5.1 使用 String 和 StringBuilder.....75	4.5.1 配置文件的规则.....119
3.5.2 格式化字符串.....78	4.5.2 配置 Web.config 文件.....120
3.6 类和对象.....80	4.6 本章小结.....121
3.6.1 定义类.....81	4.7 思考和练习.....122
3.6.2 定义成员.....82	
3.6.3 对象的创建与回收.....86	第5章 服务器控件.....124
3.6.4 封装性、继承性和多态性.....87	5.1 控件简介.....124
3.6.5 定义接口和抽象类.....89	5.1.1 控件的分类.....124
3.7 委托与事件.....91	5.1.2 在页面中添加 HTML 服务器控件.....125
3.7.1 委托.....92	5.1.3 在页面中添加 Web 服务器控件.....125
3.7.2 事件.....93	5.1.4 以编程方式添加 服务器控件.....126
3.8 C#2008 的新特性.....94	5.1.5 设置服务器控件属性.....126
3.8.1 扩展方法.....94	5.2 HTML 服务器控件.....127
3.8.2 对象初始化器.....95	
3.8.3 匿名类型.....97	
3.8.4 隐式类型局部变量.....97	

5.2.1 HTML 服务器控件的 公共属性	129	6.5.2 利用 Session 存储信息	183
5.2.2 HTML 服务器控件的使用	130	6.5.3 Timeout 属性	184
5.3 Web 服务器控件	139	6.5.4 Abandon 方法	184
5.3.1 基本的 Web 服务器控件	141	6.5.5 使用 Session 对象的 注意事项	184
5.3.2 ASP.NET 列表控件	149	6.6 Application 对象	185
5.4 验证控件	155	6.6.1 Application 对象简介	185
5.4.1 验证控件及其作用	155	6.6.2 利用 Application 对象 存储信息	185
5.4.2 验证控件的公有成员	156	6.6.3 Global.asax 文件	186
5.4.3 验证控件的用法	157	6.7 Server 对象	188
5.4.4 禁用数据验证	168	6.7.1 Server 对象简介	188
5.5 本章小结	169	6.7.2 ScriptTimeout 属性	189
5.6 思考和练习	169	6.7.3 CreateObject 方法	189
第 6 章 ASP.NET 常用对象	171	6.7.4 HTMLEncode 方法	189
6.1 对象简介	171	6.7.5 URLEncode 方法	190
6.2 Request 对象	172	6.7.6 MapPath 方法	191
6.2.1 Request 对象简介	172	6.7.7 Execute 方法	192
6.2.2 使用 QueryString 属性	172	6.7.8 Transfer 方法	192
6.2.3 ServerVariables 属性	174	6.8 本章小结	192
6.2.4 ClientCertificate 属性	174	6.9 思考和练习	193
6.2.5 Browser 属性	175	第 7 章 ADO.NET 数据库编程	196
6.3 Response 对象	175	7.1 ADO.NET 基础	196
6.3.1 Response 对象简介	175	7.1.1 什么是 ADO.NET	196
6.3.2 利用 Write 方法输出信息	176	7.1.2 与数据有关的命名空间	197
6.3.3 使用 Redirect 方法引导客户 至另一个 URL 位置	176	7.2 ADO.NET 数据提供者	198
6.3.4 关于 BufferOutput 属性	177	7.2.1 Connection 对象	198
6.3.5 输出缓存资料	178	7.2.2 Command 对象	200
6.3.6 WriteFile 方法	178	7.2.3 DataReader 对象	203
6.4 Cookie 对象	179	7.2.4 DataAdapter 对象	205
6.4.1 什么是 Cookie	179	7.3 使用 ODBC.NET Data Provider	206
6.4.2 如何读写 Cookie	180	7.3.1 ODBC .NET Data Provider 简介	206
6.4.3 检测用户是否启用了 Cookie	182	7.3.2 连接 ODBC 数据源	207
6.5 Session 对象	182	7.4 DataSet 对象	209
6.5.1 Session 对象简介	182		

7.4.1 DataSet 对象简介.....	209	9.1.5 ASP.NET 3.5 AJAX	289
7.4.2 插入记录	213	9.1.6 ASP.NET 3.5 AJAX 简单示例.....	290
7.4.3 删除记录	215	9.2 ASP.NET 3.5 AJAX 控件.....	292
7.4.4 更新记录	216	9.2.1 ScriptManger(脚本管理员) 控件.....	292
7.5 ADO.NET 3.5 的新特性	218	9.2.2 Timer(时间)控件	294
7.6 本章小结.....	218	9.2.3 UpdatePanel(更新区域) 控件.....	296
7.7 思考与练习	218	9.2.4 UpdateProgress(更新进度) 控件.....	298
第 8 章 ADO.NET 数据库高级操作	222	9.3 本章小结	299
8.1 数据源控件.....	222	9.4 思考和练习	300
8.1.1 SqlDataSource 控件	223	第 10 章 网站会员登录与注册系统.....	302
8.1.2 AccessDataSource 控件	229	10.1 设计思路.....	302
8.1.3 ObjectDataSource 控件	230	10.1.1 功能概述	302
8.1.4 SiteMapDataSource 控件	236	10.1.2 程序业务流程图	302
8.1.5 XmlDataSource 控件.....	238	10.1.3 数据库设计	303
8.1.6 LinqDataSource 控件	243	10.1.4 文件组织结构	304
8.2 数据绑定技术	250	10.2 创建数据库.....	304
8.2.1 简单数据绑定技术	250	10.3 会员注册.....	305
8.2.2 复杂的数据绑定技术	252	10.3.1 实现关键技术	305
8.2.3 Eval()和 Bind()方法.....	253	10.3.2 功能实现	307
8.3 数据绑定控件	254	10.4 会员登录.....	310
8.3.1 Repeater 控件	254	10.4.1 关键技术实现方法	310
8.3.2 DataList 控件.....	257	10.4.2 功能实现	313
8.3.3 GridView 控件	260	10.5 本章小结.....	315
8.3.4 DetailsView 控件	263	10.6 思考和练习	316
8.3.5 FormView 控件.....	267	参考文献.....	319
8.3.6 DataPager 控件.....	273		
8.3.7 ListView 控件	274		
8.4 本章小结.....	282		
8.5 思考与练习	283		
第 9 章 ASP.NET 3.5 AJAX.....	286		
9.1 Ajax 简介	286		
9.1.1 什么是 Ajax	286		
9.1.2 Ajax 与传统 Web 技术 的区别.....	287		
9.1.3 Ajax 的优点	288		
9.1.4 Ajax 使用的技术.....	289		

第1章 ASP.NET 3.5概述

早期网站的网页一般都是静态的，当数据发生变化时，必须重新编制网页。逐个修改网页需要耗费大量的时间和人力，如何使用已有数据库的数据资料？如何实现人机交互？为了解决这一问题，CGI 技术应运而生，如今已有多种实现交互式网页的方法，如 Perl、ASP、PHP、JSP 等。微软公司推出的 ASP.NET，更是把 Web 编程提高到了一个新的境界。

本章的学习目标：

- 理解静态网页与动态网页的概念及其工作原理
- 了解常见的动态网络程序设计语言
- 理解.NET 框架的组成
- 掌握 Visual Studio 2008 的安装方法
- 掌握开发 ASP.NET Web 应用程序的一般流程

1.1 Web 基础知识

1.1.1 HTTP 协议

WWW(World Wide Web)又称万维网，起源于 1989 年欧洲粒子物理研究所(CERN)，当时是为了研究人员互相传递文献资料用的。在 WWW 出现之前，Internet 主要用于科学研究和军事方面。自从 WWW 问世以后，Internet 迅速进入千家万户，成为人们学习、工作、交流、娱乐的一个非常重要的手段。HTTP(Hyper Text Transfer Protocol)，即超文本传输协议是在 Internet 中进行信息传送的协议，浏览器默认使用该协议。

从浏览器向 Web 服务器发出的访问某个 Web 网页的请求叫做 HTTP 请求。Web 服务器收到 HTTP 请求后，就会按照请求的要求，寻找相应的网页。如果找到，就把网页的 HTML(Hypertext Markup Language，超文本标记语言)代码通过 Internet 传回浏览器；如果没有找到，就发送一个错误信息给浏览器，后面的这些操作叫做 HTTP 响应。

HTTP 协议是一个无状态协议，也就是说，使用该协议时，不同的请求之间不会保存任何信息。每个请求都是独立的，它不知道现在的请求是第一次发出还是第二次或第三次发出，也不知道这个请求的发送来源。当用户请求到所要的网页后，就会断开与 Web 服务器的连接。

从程序设计角度来看，无状态的特点对于 HTTP 来说是一个缺点，因为这使得某些功能很难实现。但是，由于网络本身的特点，这也是没有办法改变的。可以假设一下，如果 HTTP 协议是一个有状态的协议，那么，就需要在 Web 服务器上保存用户的每一个连接，这样可能

会导致服务器瘫痪。

1.1.2 Web 服务器和浏览器

Web 服务器就是一台安装了 Web 服务器软件的计算机,它可以为提出 HTTP 请求的浏览器提供 HTTP 响应。常见的 Web 服务器软件有 Apache 和 IIS。Apache 是一个开放源码、采用模块化设计的 Web 服务器软件,具有很强的安全性和稳定性。IIS 是微软公司的产品,最大的特点是图形化的管理界面,使用方便,易于维护。

浏览器是运行在客户机上的程序,用户可以通过它来浏览服务器上的可用资源,因此称为浏览器。当客户进行网页浏览时,由客户的浏览器执行来自服务器的 HTML 代码,并将其内容显示给客户。最初的浏览器是基于文本的,不能显示任何图形信息。1993 年早期,随着 Mosaic 的出现,这一情况发生了改变, Mosaic 是第一个具有图形用户界面的浏览器。目前,最常用的浏览器是 Microsoft Internet Explorer(IE)和 Firefox 浏览器。

1.1.3 C/S 模式与 B/S 模式

C/S 和 B/S 是目前开发模式技术架构的两大主流技术。C/S 模式最早是由美国 Borland 公司研发的,而 B/S 模式则是由美国微软公司研发的。

(1) C/S 模式

C/S(Client/Server, 客户机/服务器)模式是一种软件系统体系结构。这种结构是建立在局域网基础之上的,它需要针对不同的操作系统开发不同版本的软件。同时,它不依赖于外网环境,即无论是否能够上网都不会影响应用。

(2) B/S 模式

B/S(Browser/Server, 浏览器/服务器)模式是随着 Internet 技术的兴起,对 C/S 模式的一种变化或改进。在这种模式下,用户工作界面是通过 Web 浏览器来实现的。B/S 模式的最大好处是能够实现不同人员、从不同地点、以不同的接入方式访问和操作共同的数据,这就大大减轻了系统维护与升级的成本和工作量,降低了用户的总体成本;其最大的缺点是对外网依赖性太强。

1.1.4 Web 的访问原理

Web 应用程序是基于 B/S 结构的。下面首先介绍客户端和服务端的概念,然后详述静态网页和动态网页的工作原理。

1. 客户端和服务端

一般来说,提供服务的一方称为服务器端,而接受服务的一方称为客户端。例如,当用户浏览搜狐主页的时候,搜狐网站所在的服务器就称为服务器端,而用户自己的计算机就称为客户端,如图 1-1 所示。

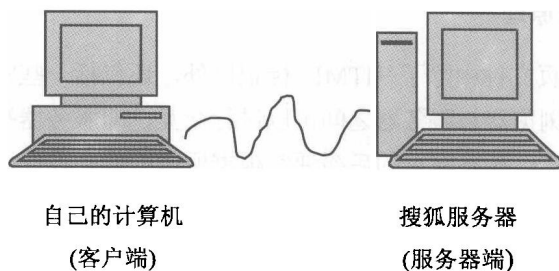


图 1-1 服务器端和客户端示例图

如果在自己的计算机上安装了 Web 服务器软件, 其他浏览者通过网络就可以访问该计算机, 那么它就是服务器端。对于很多初学者, 在调试程序时, 往往把自己的计算机既作为服务器端, 又作为客户端。

2. 静态网页的工作原理

静态网页也称为普通网页, 是相对动态网页而言的。静态并不是指网页中的元素都是静止不动的, 而是指网页文件中没有程序代码, 只有 HTML(超文本标记语言)标记, 一般后缀为 .htm、.html、.shtml 或 .xml 等。在静态网页中, 可以包括 GIF 动画, 鼠标经过 Flash 按钮时, 按钮可能会发生变化。静态网页一经制成, 内容就不会再变化, 不管何人何时访问, 显示的内容都是一样的。如果要修改网页的内容, 就必须修改其源代码, 然后重新上传到服务器上。

对于静态网页, 用户可以直接双击打开, 看到的效果与访问服务器是相同的。这是因为在用户访问该网页之前, 网页的内容就已经确定, 无论用户何时访问, 以怎样的方式访问, 网页的内容都不会再改变。静态网页的工作流程可以分为以下 4 个步骤。

- (1) 编写一个静态网页文件, 并在 Web 服务器上发布。
- (2) 用户在浏览器的地址栏中输入该静态网页的 URL(统一资源定位符)并按回车键, 浏览器发送访问请求到 Web 服务器。
- (3) Web 服务器找到此静态网页文件的位置, 并将它转换为 HTML 流传送到用户的浏览器。
- (4) 浏览器收到 HTML 流, 显示此网页的内容。

在步骤(2)~(4)中, 静态网页的内容不会发生任何变化, 其原理如图 1-2 所示。

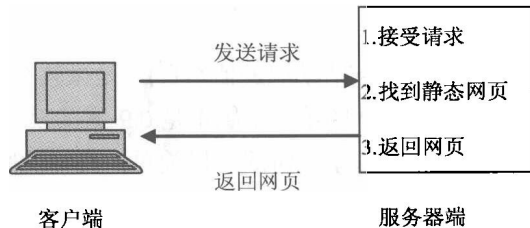


图 1-2 静态网页的工作原理

3. 动态网页的工作原理

动态网页是指在网页文件中除了 HTML 标记以外,还包括一些实现特定功能的程序代码,这些程序代码使得浏览器与服务器之间可以进行交互,即服务器端可以根据客户端的不同请求动态产生网页内容。动态网页的后缀通常根据所用的程序设计语言的不同而不同,一般为 .asp、.aspx、.cgi、.php、.perl、.jsp 等。动态网页可以根据不同的时间、不同的浏览者显示不同的信息。常见的留言板、论坛、聊天室都是用动态网页实现的。

动态网页相对复杂,不能直接双击打开,动态网页的工作流程分为以下 4 个步骤。

(1) 编写一个动态网页文件,其中包括程序代码,并在 Web 服务器上发布。

(2) 用户在浏览器的地址栏中输入该动态网页的 URL 并按回车键,浏览器发送访问请求到 Web 服务器。

(3) Web 服务器找到此动态网页的位置,并根据其中的程序代码动态创建 HTML 流传送到用户的浏览器。

(4) 浏览器收到 HTML 流,显示此网页的内容。

从整个工作流程中可以看出,用户浏览动态网页时,需要在服务器上动态执行该网页文件,将含有程序代码的动态网页转化为标准的静态网页,最后把静态网页发送给用户,其原理如图 1-3 所示。

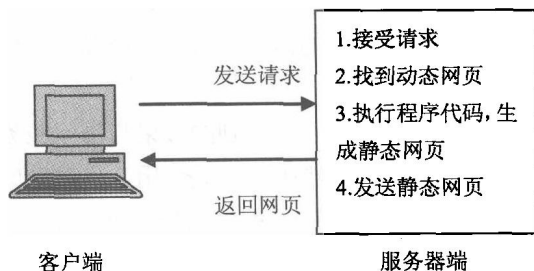


图 1-3 动态网页的工作原理

1.1.5 常见的网页程序设计技术

目前,常见的动态网页程序设计技术有 ASP、PHP、JSP 和 ASP.NET 等,下面分别介绍各自的特点。

1. ASP

ASP 的全称是 Active Server Pages(活动服务器网页),是微软在 1996 年底推出的一种用于取代 CGI 的动态网页开发技术。ASP 的第一个版本是 0.9 测试版。它给 Web 开发带来一阵风暴,使得建立一个动态页面就如小孩子玩游戏一样简单,目前的版本为 3.0。

ASP 既不是一种语言,也不是一种开发工具。ASP 文件就是在普通的 HTML 文件中嵌入 VBScript 或 JavaScript 脚本语言程序,其后缀名是 .asp。当客户请求一个 ASP 文件时,服务器端就会运行 ASP 文件中的脚本代码,并转化为标准的 HTML 文件,然后发送到客户端。ASP 提供了几个非常有用的内部对象和内部组件,利用它们可以轻松地实现表单上传、存取

数据库等功能。

ASP 的最大优点是简单好学,又有微软的强大支持,所以一经推出就备受欢迎,很多网站都是用 ASP 开发的。

ASP 的缺点主要是不能跨平台,一般只能在 Windows 系列的操作系统上运行。

2. PHP

PHP 是 Rasmus Lerdorf 于 1994 年开发的,其最初目的是帮助 Lerdorf 记录他个人网站的访问者。1995 年,他开发了一个名为个人主页工具(Personal Home Page Tool)的包,也就是 PHP 第一个公开发布版本。后来,人们开始使用一个递归式的名字 PHP: Hypertext Preprocessor(超文本预处理器),这使得它原来的名字逐渐被人们所遗忘。PHP 是一个开放源码的产品,其官方网站是 <http://www.php.net>,用户可以自由下载。

PHP 程序可以运行在 UNIX、Linux 及 Windows 操作系统上,对客户端浏览器也没有特殊要求。PHP、MySQL 数据库和 Apache Web 服务器是一个比较好的组合。

PHP 也是将脚本语言嵌入 HTML 文档中,大量采用了 Perl、C++ 和 Java 的一些特性,其文件的扩展名是 .php、.php3、.phtml 等。PHP 程序在服务器端执行,转化为标准的 HTML 文件发送到客户端。

PHP 的主要优点是免费和开放源码,这对于许多要考虑成本的商业网站,尤为重要。

3. JSP

JSP 的全称是 Java Server Pages,是由 Sun 公司于 1999 年 6 月推出的一种全新的动态页面技术。JSP 是 Java 开发阵营中最具代表性的解决方案,它不仅拥有与 Java 一样的面向对象、便利、跨平台等优点和特性,而且还拥有 Java Servlet 的稳定性,并且可以使用 Servlet 提供的 API、JavaBean 及 Web 开发框架技术,使页面代码与后台处理代码分离,提高工作效率。在目前流行的 Web 程序开发技术中,JSP 是比较热门的一种。

JSP 其实就是将 Java 程序片段(Scriptlet)和 JSP 标记(Tag)嵌入到普通的 HTML 网页中。当客户端访问一个 JSP 网页时,由 JSP 引擎解释 JSP 标记和其中的程序片段,生成所请求的内容,然后将结果以 HTML 格式返回给客户端。

JSP 的主要优点是开放的、跨平台的结构,可以运行在几乎所有的操作系统上。而且采用先编译后运行的方式,能够提高执行效率。

4. ASP.NET

ASP.NET 是对 ASP 的继承,如 Request、Response、Application、Session、Server 等对象依然存在。但 ASP.NET 又不是 ASP 的简单升级,而是微软发展的新的体系结构.NET 的一部分。ASP.NET 完全是一个新体系,是一个由类和对象(组件)组成的完全面向对象的系统,更加安全、容易配置和部署。

相对于 ASP 来说,ASP.NET 具有如下特点:

(1) 程序结构清晰

过去的 Web 页面将代码和标记混合在同一个文件中,更新和维护都很费力。ASP.NET

改变了这一点,它采用代码隐藏技术将代码与标记分离,代码采用 VB.NET、C#等面向对象语言编写,简单明了。

(2) 效率增强

ASP 属于一种解释型的编程框架,它的核心是 VBScript 和 JavaScript,无法进行像传统编程语言那样的底层操作,并且由于它是解释执行的,所以在运行效率上大打折扣。而 ASP.NET 是一种编译型的编程框架,大大提高了执行速度,而且可以用 Visual Basic.NET 和 C#等语言来编写,不必借助于其他编程语言就可以进行很多底层操作。

(3) 顶级开发工具支持

之前的 Web 窗体需要编程人员一行一行地输入代码,运行之后才能看到效果。现在的 Web 窗体则简单多了,微软提供了一个功能丰富的开发工具 Visual Studio.NET,编程者只需将控件拖放到页面上,然后编写代码实现控件的效果即可完成以前需要大量代码才能实现的功能。

(4) 开发简单

ASP.NET 中的很多特性使得开发变得更简单。如它提供的验证控件可以方便地验证客户是否正确填写表单内容,不需要编写大段的验证代码。另外,提交表单后,可以自动保留客户在表单中输入的内容。

(5) 支持 Web 服务

Web 服务,其实就是一种特殊的 Web 组件,该组件有一些属性和方法,其他网上应用程序或传统应用程序可以远程调用这些属性和方法,并返回一个结果,这是 ASP.NET 的关键部分。Web 服务基于 SOAP(简单对象访问协议),可以跨平台进行操作。

1.2 ASP.NET 3.5 简介

1.2.1 .NET 框架

2000 年 6 月,微软公司提出了“Microsoft.NET 下一代互联网软件和服务战略”,在全球 IT 行业中引起了广泛关注。微软公司把 Microsoft.NET 描述为一个建立在下一代 Internet 上的网络平台。更详细的解释是 Microsoft.NET 是一种分布式的运算框架,以 XML 为基础,以 Web 为核心,并结合其他多种技术,最大限度地利用 Internet 上丰富的资源来提高工作效率。

Microsoft.NET 的最终目的就是让用户无论在任何地方、任何时间,利用任何设备都能够访问他们所需要的信息、文件和程序,而无需知道这些东西存在什么地方,甚至连如何获得等具体细节都不知道。只需发出所需要的请求,然后接收就可以了,所有后台的复杂性是完全屏蔽起来的,也就是透明的。

Visual Studio.NET 是微软公司开发的可用于企业规模级 Web 开发的应用程序,又称为 .NET 框架(.NET Framework)。.NET 框架主要由以下 3 部分组成,如图 1-4 所示。

- 公共语言运行时(Common Language Runtime, CLR)
- .NET Framework 类库(.NET Framework Class Library)

- VB.NET、C#等语言运行库

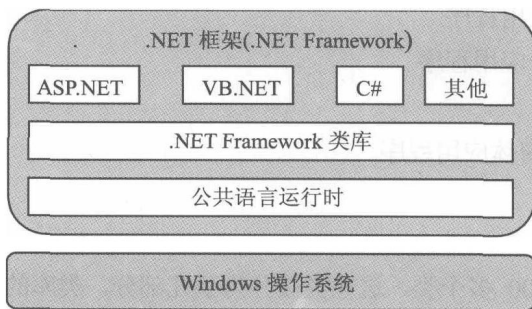


图 1-4 .NET 框架示意图

(1) 公共语言运行库(CLR)

公共语言运行时是.NET 框架的运行环境，负责运行和维护用户编写的任何程序代码，包括内存管理、线程执行、代码执行、代码安全验证、编译及其他系统服务等。

CLR 为每一种.NET 语言提供了一个编译器，当网页第一次被访问时，将会编译生成一种通用的中间语言，最初被命名为微软中间语言(Microsoft Intermediate Language, MSIL)，现在通常称为中间语言(Intermediate Language, IL)。中间语言是一种类似于汇编的程序语言，并不是可以直接执行的机器代码，进行了一系列优化。不论最初使用的是 VB.NET 还是 C# 或其他任何语言，在编译之后，所有的 IL 程序都具有相同的形式。然后 IL 程序利用即时编译器(Just In Time, JIT)在本机上进一步编译成机器代码以便执行。当网页再次被访问时，就只需由 IL 语言生成机器代码即可，所以这是一种高效的编译方法。如果源代码有改变，则需要重新编译为 IL 语言，然后再生成机器代码。.NET 语言的编译过程如图 1-5 所示。

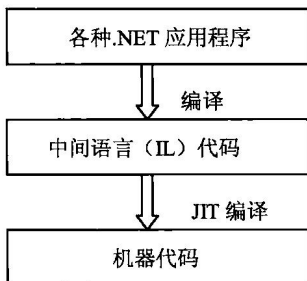


图 1-5 .NET 语言的编译过程

采用这种方式的好处在于可以满足跨平台的需要：由于中间语言与机器无关，所以它可以在任何一个可以运行 CLR 的机器上运行。并且由于所有关于机器的优化都是由 CLR 进行的，所以不存在早期编译所产生的中间语言由于机器不同而产生不兼容问题。

(2) .NET Framework 类库(.NET Framework Class Library)

.NET 提供了强大的类库，使我们可以轻松地构建程序。关于类库的概念一直就存在，VC 有 MFC 类库、Delphi 有类库 VCL、Java 有 Swing、AWT 等类库。这些类库封装了系统底层的功能并提供更好的操作方式。.NET 中的类库封装了对 Windows、网络、文件、多媒体的处理功能，是所有.NET 语言都必须使用的核心类库。使用.NET 基本类库(FCL)可以开

发以下 6 种应用程序:

- Windows 窗体应用程序
- Windows 控制台应用程序
- XML Web 服务
- ASP.NET Web 窗体应用程序
- Windows 服务
- .NET 组件

FCL 类库大约有 7000 多个类, 这些类被分为了几部分, 类库的每一部分都被包含在一个命名空间下。 .NET 的命名空间是指功能相近的类的集合。命名空间呈层次结构, 即命名空间下又可分为子命名空间。每个命名空间都包含一组按照功能划分的相关的类。

在开发程序时, 可以将命名空间引入到代码中, 然后使用该命名空间下的类, 完成开发的需要。最常用的类库如表 1-1 所示。

表 1-1 .NET 常用类库

命 名 空 间	说 明
System	包含最基本的类, 这些类提供数据类型、事件和事件处理、接口、属性、异常处理等操作
System.Windows.Forms	包含用于创建基于 Windows 的窗体应用程序的类
System.Collections	其中的类用于管理对象集合, 包括常用的集合类型: 堆栈、队列、散列表等
System.Drawing	其中的类用于操作二维图形, 用于在 Windows 窗体应用程序及 Web 窗体应用程序中显示图像或进行打印
System.IO	提供用于操作 I/O 流、遍历目录和读写文件的类
System.NET	提供用于网络通信的类, 如 P2P 通信等
System.Reflection	操作反射, 提供用于查看元数据以及延迟绑定类型和它们的成员的类
System.Text	提供用于以不同的编码方式(如 ASCII 或者 Unicode)来操作文本的类
System.Threading	操作线程, 用于异步操作以及同步访问资源
System.Xml	提供用于处理 XML 架构和数据的类
System.Web.Services	提供用于创建和操作 Web 服务的类
System.Web.UI	提供用于操作 Web 窗体创建基于 Web 的应用程序的类

(3) .NET 语言

.NET 框架支持多种开发语言, 最初包含 5 种: Visual Basic. NET(VB. NET)、Managed C++. NET、Jscript. NET、J#. NET 和一种新的语言 C#。VB. NET 基于 VB 6.0, 它广泛用于 Web 编程和包括图形用户界面在内的软件开发。VB. NET 与 VB 在许多方面都有所不同, 最重要的区别就是 VB. NET 是成熟的面向对象语言, 而 VB 不是。Managed C++. NET 是 C++ 语言的垃圾收集版本。Jscript. NET 基于 Javascript, 但也完全支持面向对象编程。J#. NET 是微软的 Java 版本, 使原来的 Java 程序员可以很容易地转向 .NET 应用程序的开发。C# 是一种新型