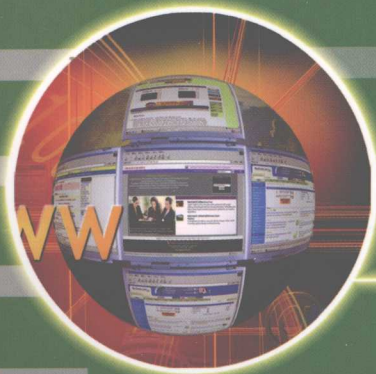


21世纪高等院校计算机科学规划教材

ASP.NET Web 程序设计

王祖俪 主 编
王 敏 吴 震 刘 泉 王力洪 等编著
甘 刚 主 审



中国水利水电出版社
www.waterpub.com.cn

21世纪高等院校计算机科学规划教材

ASP.NET Web 程序设计

王祖偲 主编

王敏 吴震 刘泉 王力洪 等编著

甘刚 主审

ISBN 7-118-04444-3	定价	58.00元
787mm×1092mm 1/16 32印张 2007年8月第1版第1次印刷	印次	0001-4000册
北京航天航空出版社印刷	编	
北京航空航天大学出版社	印	
地址：北京市海淀区学院路5号	特	
电话：(010) 62512666 (总机) 62511833 (营销中心) 62562818 (水)	售	
http://www.waterpub.com.cn	空	
网址：www.waterpub.com.cn	出	
中国水利水电出版社	版	
地址：北京西便门大街2号	行	
电话：(010) 64004444	发	
王敏 吴震 刘泉 王力洪 等编著	售	
王祖偲 主审	作	
ASP.NET Web 程序设计	书	

中国水利水电出版社

北京航空航天大学出版社

北京航空航天大学出版社

内 容 提 要

ASP.NET 技术是 Microsoft 公司推出的新一代基于 B/S 的动态 Web 开发工具,是目前电子商务、网站设计以及电子政务等 Web 应用开发的主流技术之一。本书从应用开发的角度介绍了 ASP.NET 程序设计技术,使读者学会使用 ASP.NET 技术开发 Web 应用程序。

全书共 11 章。第 1 章和第 2 章介绍了 ASP.NET 的编程基础知识、运行开发环境以及常用语言 C# 的语法结构;第 3 章至第 9 章重点介绍了 ASP.NET 的页面设计技术(控件的使用、组件对象的属性和方法)、数据库的访问技术、系统的配置、异常处理机制和网站设计中提高性能的页面缓存,以及文件操作等知识与实现技巧;第 10 章和第 11 章分别以两个实际项目——资产报修系统和在线产品预定系统为例,介绍了数据库技术、软件开发技术以及 ASP.NET 技术在一个实际开发项目中的综合利用。

本书内容全面、逻辑与结构合理、讲解由浅入深、示例典型实用、注释详尽。本书非常适合作为高等院校计算机类、软件开发类专业开展 Web 应用开发课程教学的教科书,同时也适合作为广大 Web 应用程序开发人员的自学用书。

本书实例的源代码和电子教案可以从中国水利水电出版社网站下载,网址为:
<http://www.waterpub.com.cn/softdown/>。

图书在版编目(CIP)数据

ASP.NET Web 程序设计 / 王祖俪主编. —北京: 中国水利水电出版社, 2007

21 世纪高等院校计算机科学规划教材

ISBN 978-7-5084-4605-9

I. A… II. 王… 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(2007)第 067990 号

书 名	ASP.NET Web 程序设计
作 者	王祖俪 主 编 王 敏 吴 震 刘 泉 王力洪 等编著 甘 刚 主 审
出版 发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net(万水) sales@waterpub.com.cn
经 售	电话: (010) 63202266(总机)、68331835(营销中心)、82562819(万水) 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787mm×1092mm 16 开本 17 印张 409 千字
版 次	2007 年 6 月第 1 版 2007 年 6 月第 1 次印刷
印 数	0001—4000 册
定 价	26.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

21 世纪高等院校计算机科学规划教材

编委会

主 任 袁开榜

副 主 任 孙春亮 杨庆川

编委名单

张仕斌 梅 挺 蔡乐才 吴文权

汪启荣 李秀疆 谢建华 甘 刚

冼 进 张 松 徐振明 刘 涛

瞿 中 黄同愿 李明富 刘文清

游洪跃 梁 洁 王立君 杨元泓

序

随着计算机科学与技术的发展, 计算机应用已经渗透到人们生活、工作和学习的方方面面, 从而日益改变着人类传统的工作与生活方式。这就要求当代大学生在校学习期间就应储备更多的计算机专业知识。

为了大力推广计算机应用技术, 更好地适应当前高等院校计算机教育模式的转变, 我们组织一批学术水平较高、教学经验丰富、实践能力较强的学术带头人、科研人员和从事相关课程教学的主要骨干教师, 对当前国内外高等院校计算机教育的教学现状与发展趋势、新形势下如何加强高等院校的教材建设等问题进行了深入的研究和探讨, 并成立了“21世纪高等院校计算机科学规划教材”编委会, 在明确了高校的人才培养模式、培养目标和课程体系的框架下, 组织编写了本套“21世纪高等院校计算机科学规划教材”。

本套教材具有以下显著特色:

(1) 充分体现了计算机教育第一线的需求。在编写之初, 编委会经过大量的前期调研和策划, 广泛地了解各高等院校的教学现状、市场需求, 研讨了课程设置、课程体系, 拟定了相关的知识单元和知识点, 充分听取了教学第一线教师对计算机教育的意见, 使本套教材充分反映了老师们的需求。

(2) 各高校计算机院(系)院长(主任)对本套教材的建设十分重视, 热情鼓励教师积极参与编写, 充分展现了各个高校在计算机教育教学改革中取得的最新教研成果。

(3) 本套教材在内容安排上既注重内容的全面性, 也充分考虑了不同学科、不同专业对计算机知识的不同需求的特殊性。

(4) 本套教材为了充分调动学生分析问题、解决问题的积极性, 以及锻炼学生的实际动手能力, 在全书中大力增加了实践检验所占的比重。

(5) 力求实践性强是本套教材的一大特色, 通过案例教学, 将最急需、最实用的计算机知识传授给学生。

为进一步体现实用性, 本配套教材在编写时配有课程学习辅导、实验指导、综合实训、电子教案等, 以使教材向多元化、多媒体化发展, 满足广大教师的教学需要。

总之, 本套教材凝聚了众多长期工作在教学、科研第一线的教师及科研人员的教学科研成果、教学经验和智慧, 在写法上体现了理论与实践相结合, 相关的知识点讲解清晰、透彻, 注重教学实践, 力求科学实用, 符合教学习惯。语言通俗易懂, 内容丰富翔实, 既有对基本理论及使用方法的透彻讲解, 又注重实例与技巧的融会贯通。这套教材是新形势下计算机教育改革的一种新的尝试, “新”就会有許多值得修改的地方。我们期待广大读者对本套规划教材提出宝贵意见, 以便进一步修订, 使本套规划教材不断完善和提高。

21世纪高等院校计算机科学规划教材编委会

主任 袁开榜

2006年6月

前 言

ASP.NET 技术是 Microsoft 公司推出的新一代基于 B/S 的动态 Web 开发工具，是目前电子商务、网站设计以及电子政务等 Web 应用开发的主流技术之一。

ASP.NET 不同于 ASP，它建立在 .NET Framework 之上，提供了更易于编写且结构更加清晰的代码，它主要使用 C# 或 VB.NET 等模块化语言编写程序，提高了代码的再利用性和共享性。ASP.NET 在结构上几乎完全是基于组件和模块化的，Web 应用程序开发人员使用该技术可以实现模块化的、功能更强大的应用程序。

本书特点：

本书从 Web 编程的基本概念入手，介绍了 .NET 的运行环境、语言语法、对象组件和控件使用，并结合 Web 编程中常用的数据库的交互操作，详细介绍了 ASP.NET 与 SQL Server 的连接以及交互操作的过程。最后本书还对 .NET 中的异常处理情况、如何提高网站性能，以及常见的文件操作方法进行了讲解。在讲述过程中，原理讲述与代码分析相结合，步骤与注释相结合。此外，本书在最后两章中详细讲解了两个实际使用 ASP.NET 开发的项目的整个开发过程和综合技术的使用。

概括来讲，本书具有如下特点：

- 从原理入手，内容全面，涉及一般概念、控件、对象的使用、数据库交互，以及设计与开发的整个流程的所有内容。
- 全书内容结构合理、取舍得当，非常适合初学者和具有一定 Web 基础知识和 HTML 知识的读者学习使用，能够使读者对 ASP.NET 项目开发有一个整体概念，并为更好地完成 Web 程序设计打下坚实基础。
- 讲解详细。对控件的使用、对象的属性和方法、数据库连接与显示等开发的一些重要知识进行了详细介绍，不但能使读者掌握原理性知识，还可以照本演练。
- 示例丰富、注释详尽。每一章中，针对不同的讲解知识都有示例进行说明，这些示例大多来自实际的项目，实用且典型，对涉及的关键代码有详细的注释。

组织结构：

全书按照 ASP.NET 设计开发的一般过程来编排内容，使得读者能够通过不断地深入学习学会使用 ASP.NET 技术开发 Web 应用程序。全书共 11 章，具体内容安排如下：

第 1 章和第 2 章，介绍 ASP.NET 的编程基础知识、运行开发环境以及常用语言 C# 的语法结构；第 3 章至第 9 章，重点介绍了 ASP.NET 的页面设计技术（控件的使用、组件对象的属性和方法）、数据库的访问技术、系统的配置、异常处理机制和网站设计中提高性能的页面缓存，以及文件操作等知识与实现技巧；第 10 章和第 11 章，分别以两个实际项目——资产报修系统和在线产品预定系统为例，介绍了数据库技术、软件开发技术以及 ASP.NET

技术在一个实际开发项目中的综合利用。

读者对象:

本书内容全面、逻辑与结构合理,讲解由浅入深、示例典型实用、注释详尽,非常适合作为高等院校计算机类、软件开发类专业开展 Web 应用开发课程教学的教科书,同时也可作为广大 Web 应用程序开发人员的自学用书。

概括来讲,本书适合以下人员:

- 对 Web 开发感兴趣或者刚开始学习 Web 技术的开发人员和读者。
- 对 ASP 和 Web 开发有一定了解且想了解 ASP.NET 技术的开发人员和读者。
- 有一定 ASP.NET 基础且希望通过更多实例熟悉 ASP.NET 技术的人员。

尽管本书面对初次接触 Web 应用开发的人员非常适合,但对于读者来说,掌握一定网络知识和 HTML 语言更容易理解本书所授知识,本书使用的语言为 C#语言,虽然书中也对该语言有一定介绍,但若读者了解一点 C 语言的结构将更有利于尽快地熟悉语言。

编者与致谢:

本书由王祖俪主编,王敏、吴震、刘泉、王力洪等编著。其中,第 1 章、第 4 章、第 6 章、第 7 章、第 8 章由王祖俪编写,第 2 章、第 3 章、第 5 章由王敏编写,第 9 章由吴震编写,第 10 章由王力洪编写,刘泉编写第 11 章并负责全书的规划安排,甘刚负责主审、修改和定稿工作。另外,参与编写工作的还有邹素琼、洗进、赵秋云、赵继军、彭艺、曲辉辉、周章、蒋波、徐留旺、曹振宇、张婷、温凌霜、鲁得翠、蒋泽平、魏乐、韩翔、程小英、谭小丽、卢丽娟、李小琼等,在此一并表示感谢!

本书在编写过程中,张景峰老师审阅了全稿并对书稿提出了宝贵意见和建议,还得到了成都信息工程学院网络工程系诸多老师的帮助,并参阅了许多同行作者的著作,在此向所有为本书做出贡献的同志致以衷心的感谢。

配套服务:

为充分展现本书编写特点,帮助读者深刻理解本书编写意图与内涵,进一步提高对本书教学的使用效果,我们建立本书使用指导联络方式,这将是读者与编者之间交流沟通的直通车,欢迎读者将图书使用过程中的各种问题与探讨、建议反馈给我们,本书编者会竭诚给你答复。我们的 E-mail: china_54@tom.com。

目 录

序	
前言	
第 1 章 ASP.NET 概述	1
1.1 Web 基础知识	1
1.1.1 HTTP 协议	1
1.1.2 Web 服务器	1
1.1.3 静态网页	2
1.1.4 动态页面	2
1.2 ASP.NET 工作原理	3
1.2.1 从 ASP 到 ASP.NET	3
1.2.2 NET 框架	4
1.2.3 ASP.NET 设计目标	5
1.3 ASP.NET 运行环境	6
1.3.1 IIS 安装与配置	6
1.3.2 ASP.NET 安装配置运行环境	10
1.3.3 Visual Studio.NET 界面介绍	11
1.3.4 脚本语言	15
1.4 用户登录程序	15
1.5 本章小结	16
1.6 实战检验	17
理论巩固	17
上机实战	17
第 2 章 C# 基础知识	18
2.1 C# 设计理念	18
2.2 面向对象程序	19
2.3 C# 数据类型	20
2.3.1 值类型	20
2.3.2 引用类型	23
2.4 C# 语法	25
2.4.1 声明变量	25
2.4.2 C# 基本语法	26
2.5 函数	28

2.5.1	定义函数	29
2.5.2	调用函数	29
2.6	本章小结	30
2.7	实战检验	30
	理论巩固	30
	上机实战	31
第 3 章	ASP.NET 组件对象	32
3.1	Page 类	32
3.2	Request 对象	34
3.3	Response 对象	38
3.4	HttpServerUtility 对象	40
3.5	HttpApplication 对象	43
3.5.1	Application 对象的典型应用——计数器	43
3.5.2	计数器的锁定	44
3.6	Session 对象	44
3.6.1	Session 的意义与基本使用——页面间传值	45
3.6.2	改进的计数器实例	47
3.6.3	Session 的配置	50
3.7	HttpCookies 对象	50
3.7.1	常用属性	50
3.7.2	设置 Cookie 值	51
3.7.3	读取 Cookie	51
3.7.4	Cookie 的删除	52
3.8	本章小结	52
3.9	实战检验	52
	理论巩固	52
	实战检验	52
第 4 章	高级控件和用户控件	53
4.1	HTML 服务器控件	53
4.1.1	基本 HTML 标注	53
4.1.2	HTML 服务器控件构架	54
4.2	Web 服务器控件	70
4.2.1	Label Web 控件	70
4.2.2	Button 控件	71
4.2.3	Image 控件	74
4.2.4	HyperLink 控件	74
4.2.5	LinkButton Web 控件	75

111	4.2.6	ImageButton Web 控件	76
111	4.2.7	TextBox Web 控件	76
111	4.2.8	Table Web 控件、TableRow Web 控件及 TableCell Web 控件	77
111	4.2.9	Panel Web 控件	80
111	4.2.10	RadioButton Web 控件	81
111	4.2.11	RadioButtonList Web 控件	82
111	4.2.12	Listitem Web 控件	83
111	4.2.13	CheckBox Web 控件	84
111	4.2.14	CheckBoxList Web 控件	85
111	4.2.15	DropDownList Web 控件	86
111	4.2.16	ListBox Web 控件	87
122	4.3	Web 服务器验证控件	88
124	4.3.1	RequireFieldValidator Web 控件	89
124	4.3.2	CompareValidator Web 控件	90
124	4.3.3	RangeValidator Web 控件	91
131	4.3.4	RegularExpressionValidator Web 控件	92
134	4.3.5	CustomValidator Web 控件	93
134	4.4	用户控件	94
130	4.4.1	用户控件简介	95
141	4.4.2	创建用户控件实例	95
144	4.4.3	使用用户控件实例	95
144	4.5	自定义控件	96
146	4.5.1	自定义控件	97
148	4.5.2	创建自定义控件实例	97
148	4.5.3	使用自定义控件	104
149	4.6	本章小结	106
122	4.7	实战检验	106
122		理论巩固	106
101		上机实战	106
107	第 5 章	应用程序配置	107
107	5.1	Global.asax	107
107	5.1.1	Global.asax 简介	107
107	5.1.2	Global.asax 的文件格式	108
107	5.2	Web.config	109
109	5.2.1	配置文件	109
109	5.2.2	Web.config 文件格式	109
111	5.2.3	在 Web.config 中保存设置	111

5.3	本章小结	112
5.4	实战检验	112
	理论巩固	112
	上机实战	112
第 6 章	数据库访问技术	113
6.1	数据库操作概述	113
6.2	结构化查询语句——SQL	115
	6.2.1 SELECT 查询	116
	6.2.2 数据更新	118
6.3	Connection 对象	119
	6.3.1 OleDbConnection 的使用	119
	6.3.2 SqlConnection 的使用	122
	6.3.3 SqlConnection 类的属性和方法	124
6.4	在 ASP.NET 中存取数据	124
	6.4.1 Command 对象	124
	6.4.2 DataReader 对象	131
6.5	DataSet	134
	6.5.1 使用 DataSet	134
	6.5.2 表间关系 DataRelation	139
	6.5.3 使用数据适配器填充 DataSet	141
6.6	数据绑定控件——Repeater 服务器控件	144
	6.6.1 Repeater 服务器控件概述	144
	6.6.2 Repeater 控件实例	146
6.7	数据绑定控件——DataList 服务器控件	148
	6.7.1 DataList 控件概述	148
	6.7.2 DataList 控件实例	149
6.8	数据绑定控件——DataGrid 服务器控件	152
	6.8.1 数据基本显示	152
	6.8.2 排序	161
	6.8.3 分页	162
6.9	本章小结	164
6.10	实战检验	164
	理论巩固	164
	上机实战	164
第 7 章	创建高效应用程序	165
7.1	缓存原理	165
7.2	页面输出缓存	166

881	7.2.1	页面输出缓存概述	166
881	7.2.2	HttpCachePolicy 类	167
881	7.2.3	缓存页的多个版本的实例	168
131	7.3	页片段缓存	169
521	7.4	数据缓存	169
521	7.4.1	数据缓存基础	169
521	7.4.2	数据缓存的操作实例	170
891	7.5	本章小结	171
141	7.6	实战检验	172
141		理论巩固	172
141		上机实战	172
	第 8 章	页面跟踪、调试和异常处理	173
151	8.1	常见错误类型	173
161	8.2	调试	174
161	8.2.1	调试错误信息的显示	174
161	8.2.2	按条件显示错误信息	176
161	8.2.3	Visual Studio.NET 的其他调试技术	177
171	8.3	异常处理	177
171	8.3.1	异常处理语句	177
181	8.3.2	异常处理实例	178
191	8.4	跟踪	179
200	8.4.1	页级跟踪实例	179
200	8.4.2	应用程序级跟踪实例	180
205	8.5	本章小结	182
205	8.6	实战检验	182
213		理论巩固	182
213		上机实战	182
	第 9 章	文件及文件操作	183
225	9.1	数据流	183
225	9.1.1	Stream 类	183
235	9.1.2	StreamWriter 类	184
239	9.1.3	StreamReader 类	185
239	9.1.4	BinaryWriter 类	185
239	9.1.5	BinaryReader 类	186
239	9.1.6	MemoryStream 类	186
245	9.1.7	NetworkStream 类	187
145	9.2	文件流	188

801	9.2.1	Directory 类	188
801	9.2.2	DirectoryInfo 类	189
801	9.2.3	File 类	190
801	9.2.4	FileInfo 类	191
801	9.3	文件操作	192
801	9.3.1	创建文件实例	192
801	9.3.2	文件写操作实例	192
801	9.3.3	文件读操作实例	193
801	9.3.4	文件删除操作实例	194
801	9.3.5	新增目录操作实例	194
801	9.3.6	删除目录操作实例	194
801	9.3.7	访问目录操作实例	195
801	9.4	本章小结	195
801	9.5	实战检验	196
801		理论巩固	196
801		上机实战	196
801	第 10 章	资产报修系统	197
801	10.1	系统概述	197
801	10.1.1	系统功能	197
801	10.1.2	页面结构设计	198
801	10.2	数据库设计	199
801	10.3	各子系统设计与实现	200
801	10.3.1	用户登录的设计与实现 (包括首页面)	200
801	10.3.2	数据库文件的连接	206
801	10.3.3	资产注册	206
801	10.3.4	资产浏览	213
801	10.3.5	用户报修	218
801	10.3.6	报修查看	222
801	10.3.7	报修查询	228
801	10.3.8	报修处理	231
801	10.3.9	系统维护	237
801	第 11 章	在线产品预定系统	239
801	11.1	创建系统前的准备	239
801	11.1.1	系统功能	239
801	11.1.2	数据库需求分析	239
801	11.1.3	文件结构和程序流程	241
801	11.2	创建系统	241

11.2.1	产品发布	241
11.2.2	产品管理	245
11.2.3	产品浏览	249
11.2.4	产品预定	252
11.2.5	预定受理	255
参考文献		256

第 1 章 ASP.NET 概述

知识点:

- ◇ 了解 Web 工作的原理
- ◇ 区别 ASP 与 ASP.NET
- ◇ 掌握 .NET 平台的工作环境和构成
- ◇ 了解 ASP.NET 程序的基本结构

本章导读:

本章将介绍 ASP.NET 程序设计应具备的基础知识,包括 Web 技术的基础知识、ASP 与 ASP.NET 的区别、ASP.NET 的架构与运行环境,以及了解 ASP.NET 的程序结构。

1.1 Web 基础知识

网页状态 8.1.1

1.1.1 HTTP 协议

HTTP 协议即超文本传输协议 (HyperText Transfer Protocol)。这个协议是在 Internet 中进行信息传送的协议,浏览器默认使用这个协议。

HTTP 请求是指从浏览器向 Web 服务器发出的搜索某个 Web 网页的请求。当 Web 服务器收到这个请求之后,就会按照请求的要求,找到相应的网页。如果可以找到这个网页,那么就 把网页的 HTML 代码通过网络传回浏览器;如果没有找到这个网页,就发送一个错误信息给发出 HTTP 请求的浏览器。后面的这些操作称为 HTTP 响应。

1.1.2 Web 服务器

Web 是基于客户机/服务器的一种体系结构,一般用户的计算机称为客户机,用于提供服务的机器称为服务器。在 Web 方式下客户端软件常用浏览器,如图 1-1 所示,客户机向服务器发送请求,要求执行某项任务,而服务器执行此项任务,并向客户机返回响应。Web 客户程序称为浏览器 (Browser),而浏览器程序基本上都是标准化的,因此,Web 体系结构可以称为浏览器/服务器结构 (B/S 结构)。在该结构中,通常很容易将客户机和服务器理解为两端的计算机。但事实上,“客户机”和“服务器”从概念上更多的是指软件,即两台机器上相应的应用程序,或者说是“客户机进程”和“服务器进程”。

基于 Web 的数据库应用采用三层客户/服务器结构,也称为 Browser/Server/Database Server 结构。第一层为浏览器,第二层为 Web 服务器,第三层为数据库服务器。浏览器是用户输入

数据和显示结构的交互界面，用户在浏览器表单中输入数据，然后将表单中的数据提交并发送到 Web 服务器，Web 服务器应用程序接收并处理用户的数据，并从数据库中查询用户数据或把用户数据录入到数据库中。最后 Web 服务器把返回的结果插入 HTML 页面传送到客户端，在浏览器中显示出来。

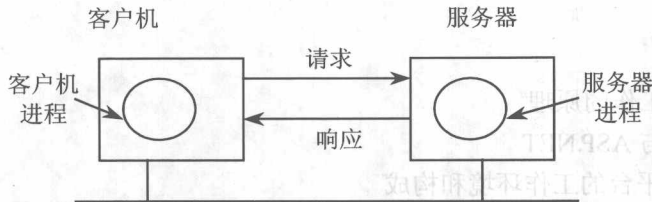


图 1-1 客户机/服务器模型

比较常见的 Web 服务器有 Apache 和 IIS。由于 ASP.NET 只能在 IIS 上运行，所以本书介绍的重点放在 IIS 上。IIS 是微软公司的操作系统 Windows 2000/XP 所提供的，本书后面的章节会详细介绍 IIS。

1.1.3 静态网页

静态网页就是用纯 HTML 代码编写的网页。这些网页的代码是用一些编辑器输入的，或者是用一些网页设计程序生成的，保存为.html 或.htm 文件的形式。由于这些网页中没有任何与用户相关的部分，所以在设计完成之后，这类程序不会被服务器端执行，所以无论是哪个用户访问这个网页、在什么时候访问这个网页、以何种方式进入这个网页，它的样子都不会发生任何变化。采用静态网页会导致很大的局限性，如果希望为用户显示一些个性化的信息，使用静态网页将无法达到这个目的。

1.1.4 动态页面

在动态网页产生之前，所有的网页都是静态的，动态网页可以为不同的用户提供个性化的服务。动态网页的扩展名一般为.asp 或.aspx。在浏览这类网页的时候，必须先服务器端执行程序，再将执行结果下载到本地浏览器上。这类程序会被服务器端执行，因执行的条件不同，所产生的结果也会有所不同。脚本语言是 DHTML 的一个重要组成部分，脚本编程实际上是通过下载到浏览器上的程序来完成所有的有关动态服务的工作，在 DHTML 中主要有两种脚本语言：JavaScript 和 VBScript，其中 JavaScript 是一种类似于 Java 编程语言的脚本语言，是第一个在 Web 上使用的脚本语言。它基于 Java 的语法，简单、易学。VBScript 是微软公司在 VB 编程语言基础上设计的一种脚本语言，由于 VB 比较流行，且和微软的其他产品有着密切的联系，所以 VBScript 是脚本语言的理想选择。常见的服务器端编程技术有：CGI、PHP、ASP、JSP 和 ASP.NET。

1.2 ASP.NET 工作原理

1.2.1 从 ASP 到 ASP.NET

ASP (Active Server Pages, 动态服务器页面) 是一种功能强大而且易于学习的服务器端脚本编程环境。它类似于 HTML、Script 与 CGI 的结合体。当用户通过浏览器向 Web 服务器请求 ASP 页面时, ASP 脚本开始运行, Web 服务器开始调用 ASP 环境读取用户所请求的文件, 并执行其中的所有脚本命令, 最后将处理以后的 Web 页面传递到客户端的浏览器。这意味着 ASP 与客户端浏览器无关, 这样减轻了客户端浏览器的负担, 大大提高了交互的速度, 此外 ASP 还允许创建 ActiveX 组件。

从 1996 年到现在, ASP 发生了重大变化, 从 ASP 1.0 诞生, 到发布 ASP 2.0, 到 IIS 5.0 所附带的 ASP 3.0, 到 2001 年, ASP+ 的出现, 它实际上是 ASP.NET 的前身, 微软公司为了配合其 .NET 计划, 同时为了表明这个新版本不只是 ASP 3.0 的简单升级, 将其命名为 ASP.NET。

ASP.NET 从结构上与以前的版本大相径庭, 它几乎完全是基于组件和模块化的。开发人员可以使用这个环境来实现更加模块化、功能更强大的 Web 应用程序。从更深的层次来研究 ASP.NET, 会发现 ASP.NET 与 ASP 之间的区别主要体现在以下几个方面:

(1) 效率。ASP 是一个脚本编程环境, 只能用 VBScript 或 JavaScript 这样的非模块化语言来编写。当 ASP 程序完成之后, 在每次请求时解释执行。解释是指把程序语句翻译成机器可以执行的二进制代码的过程, 这个过程在程序执行中进行, 所以没有办法对程序进行相关的优化。这就意味着它在使用其他语言编写的大量组件的时候会遇到困难, 并且无法实现对操作系统的底层操作。而在 ASP.NET 中, 所有的程序执行都是经过服务器编译的, 即把程序语句翻译成机器可以执行的二进制代码, 而这一过程是在程序执行之前进行的, 并且它建立在 .NET Framework 之上, 可以使用 Visual Basic、C++ 这样的模块化程序设计语言, 在第一次执行时进行了编译, 之后的执行不需要重新编译就可以直接运行, 所以速度和效率都比 ASP 提高了很多。

(2) 可重用性。在写 ASP 应用程序的时候, ASP 代码和 HTML 混合在一起。只要有需要, 就可以在任意的一个位置插入一段代码来实现特定的功能。这种方法表面上看起来很方便, 但在实际的工作中会产生大量烦琐的页面, 很难读懂, 导致代码维护很困难。虽然可以使用 include 指令尽量让程序模块化, 但是仍然不是一个最终彻底的解决方案。而 ASP.NET 则可以实现代码和内容的完全分离, 使得前面提到的问题迎刃而解。

(3) 代码量。ASP 需要对所有要实现的功能通过编写代码来实现。例如, 所有的 ASP 程序员都遇到过这样的情况: 为了保证一个用户数据提交页面的友好性, 当用户输入错误的时候会显示错误的位置, 并尽量把用户原来的输入在控件中显示出来。这样的一个行为需要程序员编写大量的代码来实现。但是它确实是一个非常简单的功能, 在其他的应用程序中根本不需要编写这么多的代码就可以实现。在 ASP.NET 中, 程序员会发现只要预先说明, ASP.NET 就可以自动实现这样的功能。所以相对来说, 要实现同样的功能, 使用 ASP.NET 比使用 ASP 的代码量要小得多。