

深入浅出

ASP

# 程序设计与开发

张德静 / 编著

- ◆ 由国内资深ASP专家精心编著,融汇作者多年的实际编程技巧与开发经验
- ◆ 每个知识点都辅以实例进行讲解,综合实例更能让您快速掌握ASP的精髓
- ◆ 大量的程序代码注释,有的放矢地讲解利用ASP技术开发动态网站的全过程
- ◆ 本书内容全面、系统,知识讲解深入浅出,堪称ASP程序设计的权威教材



CD-ROM

随书光盘内含书中所讲范例的完整源代码,部分课后习题答案及源代码和教学用 PowerPoint 演示文档



中国青年出版社  
中国青年电子出版社

<http://www.zlbooks.com> <http://www.cqchina.com>

深入浅出

ASP

# 程序设计与开发

张德静 / 编著



中国青年出版社  
中国青年电子出版社

<http://www.21books.com> <http://www.cqchina.com>

本书由中国青年出版总社独家出版。未经出版者书面许可，任何单位和个人不得以任何形式复制或传播本书的部分或全部内容。

### 图书在版编目(CIP)数据

深入浅出 ASP 程序设计与开发 / 张德静编著. —北京：中国青年出版社，2005

ISBN 7-5006-6190-8

I. 深... II. 张... III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2005) 第 020785 号

书 名：深入浅出 ASP 程序设计与开发

编 著：张德静

出版发行：中国青年出版总社

地址：北京市东四十二条 21 号 邮政编码：100708

电话：(010) 84015588 传真：(010) 64053266

印 刷：沈阳铁路局锦州印刷厂

开 本：185 × 260 1/16 印 张：24

版 次：2005 年 5 月北京第 1 版

印 次：2005 年 5 月第 1 次印刷

书 号：ISBN 7-5006-6190-8/TP · 447

定 价：36.00 元 (附赠 1CD)

# 前 言

21 世纪是网络的世纪。门户网站、专业网站、购物网站等各种千姿百态的网站构成我们的互联网世界，通过分析这些网站除了内容不同外，搭建这些网站的要素几乎完全一样，页面、社区、搜索、E-Mail 这些普通的程序构成了虚拟世界的骨架，而 ASP (Active Server Pages) 是目前开发动态网页最流行、最容易掌握的方法之一。

ASP 动态网页是美国微软公司推出的一种用以取代 CGI (通用网关接口) 的技术。ASP 本身不是一种脚本语言，它只提供了一种使镶嵌在 HTML 页面中的脚本程序得以运行的环境。我们可以通过 ASP 结合 HTML 语言、ASP 指令和 ActiveX 元件以及数据库等方面知识，使自己的 Web 服务器创建并运行动态的交互式 Web 站点。目前，ASP 技术已经相当普及，在 Internet 上几乎随处可见它的踪影。应用 ASP 技术，无须复杂的编程就可以开发出专业的动态 Web 站点。可以说，ASP 的出现给以静态内容为主的 Web 带来了全新的动态效果，同时也开创了非专业人员开发高水平网站的历史。

本书以丰富的应用实例为主线，运用大量的图片和丰富的实例，由浅入深，真实地再现了作者指引读者学习 ASP 技术的全过程。本书一共分为 15 个章节，第 1 章和第 2 章分别介绍了 ASP 的概念和 HTML 的概念，第 3 章讲述 VBScript 的基本语法和概念，第 4 章是对 ASP 基础编程知识的讲解，第 5 章~第 7 章阐述了 ASP 的 5 大内置对象 (Request、Response、Server、Application 和 Session)、Cookies 集合以及 Global.asa 文件，第 8 章介绍了 ASP 的内置组件，第 9 章~第 11 章讲解数据库基础知识以及对数据库的操作，第 12 章~第 15 章是利用 ASP 技术开发网站的实例，如网上考试系统，图书管理系统，网上购物系统以及企业内部办公管理系统。

另外，本书的前 11 章每章后面都有精心设计的课后练习题，这些练习题对读者巩固各章的知识，增加实际操作经验起着非常重要的作用。书中范例的源程序与课后习题答案均可在随书光盘中找到，并且，在光盘中我们还特意准备了与本书配套的 PowerPoint 文档，方便教学中的演示和讲解。本书适合作为初、中级读者学习 ASP 的入门图书，而丰富的实例与各种技巧的融合也适合对 ASP 有一定了解的读者作为参考书目，方便随时查阅。

本书能够在如此短的时间内顺利出版，要特别感谢为本书付出努力的所有工作者，他们在本书的编写、校审、定稿及录入过程中作了大量的工作，在此表示深深的感谢。由于作者水平有限，书中出现不足和疏漏在所难免，期盼广大读者在从本书受益的同时也能对本书中存在的不妥之处给予批评指正。

作者

2005 年 3 月

# 目 录

## 第 1 章 初识 ASP

1.1 网络程序设计语言的产生背景	1
1.2 WWW 的工作原理	2
1.3 目前主要的网络程序设计语言	3
1.3.1 ASP	3
1.3.2 PHP	3
1.3.3 JSP	3
1.4 ASP 入门	4
1.4.1 什么是 ASP	4
1.4.2 ASP 能做什么	5
1.4.3 ASP 是怎样工作的	5
1.4.4 ASP 与 HTML	6
1.4.5 ASP 构建网站的特点	6
1.4.6 ASP 的组成	7
1.4.7 ASP 的开发工具	10
1.4.8 ASP 程序的调试技巧	10
1.5 IIS 的安装配置	11
1.5.1 IIS 的安装	11
1.5.2 IIS 中虚拟目录的建立	13
1.5.3 认识 Web 属性	16
1.6 创建第一个简单的 ASP 页面	16
练习一	17

## 第 2 章 HTML 基础

2.1 什么是 HTML	19
2.1.1 HTML 文件是什么	20
2.1.2 一个简单的例子	20
2.1.3 关于 HTML 编辑器	21
2.2 使用 HTML 创建基本网页	21
2.2.1 创建基本框架	21
2.2.2 创建文本	22
2.2.3 创建图像	24

2.2.4 创建超链接	25
2.2.5 创建列表	27
2.2.6 创建表格	28
2.2.7 创建表单	31
2.3 块级元素的使用	33
2.3.1 DIV 的使用	33
2.3.2 SPAN 的使用	34
2.3.3 ADDRESS 的使用	35
2.3.4 DEL 的使用	35
2.3.5 PRE 的使用	36
2.4 常用组件介绍	37
2.4.1 Applet 的使用	37
2.4.2 Marquee 的使用	38
2.4.3 Embed 的使用	38
2.4.4 Object 的使用	39
2.4.5 MAP 的使用	40
2.5 网上商店用户登录页面设计	41
练习二	42

## 第 3 章 ASP 脚本语言——VBScript

3.1 VBScript 基础及语法	45
3.1.1 VBScript 运算符	45
3.1.2 流程控制	46
3.1.3 如何在 ASP 中使用 VBScript	50
3.2 VBScript 变量及数据类型	51
3.2.1 变量的声明	52
3.2.2 变量类型	52
3.2.3 变量数据范围	53
3.2.4 VBScript 中的常量	53
3.3 VBScript 中的数组	55
3.3.1 数组的声明	55
3.3.2 动态数组	55

3.3.3 常用数组函数	56	4.7.7 #lastmod	96
3.4 VBScript 过程	57	4.7.8 .inc 文件	96
3.4.1 Sub 过程	57	练习四	96
3.4.2 Function 过程	58	<b>第 5 章 使用 Response 对象和 Request 对象</b>	
3.5 VBScript 常用函数	59	5.1 Response 对象	99
3.5.1 数学函数	59	5.1.1 向浏览器输出数据	99
3.5.2 类型转换函数	59	5.1.2 停止输出数据	100
3.5.3 日期和时间函数	61	5.1.3 Response 的数据集合	101
3.5.4 字符串运算函数	62	5.1.4 网页重定向	102
练习三	63	5.1.5 Response 的属性	103
<b>第 4 章 ASP 基础</b>		5.1.6 Response 的方法	104
4.1 ASP 文件的结构	65	5.2 使用 Request 对象	106
4.2 声明脚本语言	66	5.2.1 从浏览器获取数据	106
4.2.1 如何使用 ASP 脚本	67	5.2.2 利用 QueryString 集合获取数据	108
4.2.2 <Script>标记和<%...%> 标记的区别	69	5.2.3 获取服务端环境变量	110
4.3 ASP 中结合对象与组件	72	5.3 一个调查表实例	111
4.4 ASP 基本语法	74	练习五	113
4.5 HTML 文件中应用 ASP 变量、常量 和数组	74	<b>第 6 章 Application 和 Session 对象的应用</b>	
4.6 在 ASP 中使用基本脚本语言	76	6.1 Application 对象的应用	115
4.6.1 条件运算	76	6.1.1 Application 对象简介	115
4.6.2 循环语句	79	6.1.2 自定义属性的应用	116
4.6.3 格式化日期和时间	82	6.1.3 使用 Application 事件和数据集合	117
4.6.4 格式化数字	85	6.1.4 一个聊天室实例	119
4.6.5 使用数学函数	86	6.1.5 网页计数器实例	122
4.6.6 使用字符串函数	88	6.2 Session 对象的应用及工作原理	124
4.6.7 Split 函数的使用	88	6.2.1 Session 对象的数据集合	125
4.6.8 使用 With 语句	90	6.2.2 Session 对象的属性	126
4.6.9 调用函数和过程	91	6.2.3 Session 对象的方法	127
4.7 使用 Server Side Include (SSI)	92	6.2.4 Session 对象的事件	127
4.7.1 SSI 的功能	92	6.2.5 Session 对象的会话	128
4.7.2 #Include 指令	94	6.2.6 密码验证实例	129
4.7.3 使用动态的 SSI	95	练习六	130
4.7.4 应用程序的输出	95	<b>第 7 章 Server 对象、Cookies 集合 和 Global.asa 文件</b>	
4.7.5 包含文件的大小	95	7.1 Server 对象的应用	133
4.7.6 #config	95		

7.1.1 Server 对象概述.....	133	<b>第 10 章 使用 SQL 语句操作数据库</b>	
7.1.2 Server 对象属性应用.....	134	10.1 SQL 语言简介.....	189
7.1.3 向浏览器输出 HTML 代码.....	134	10.2 Select 语句的基本形式.....	189
7.1.4 取得文件的路径.....	135	10.3 Select 结构各子句.....	190
7.1.5 在浏览器中输出 URL 代码.....	136	10.3.1 From 子句.....	190
7.1.6 CreatObject 的使用.....	137	10.3.2 Where 子句.....	191
7.2 Cookies 集合.....	138	10.3.3 Group By 子句.....	194
7.2.1 Cookies 工作原理.....	139	10.3.4 Order By 子句.....	196
7.2.2 将 Cookies 写入浏览器.....	140	10.3.5 Compute 子句.....	197
7.2.3 从浏览器读取 Cookies.....	140	10.4 使用函数.....	198
7.3 使用 Global.asa 文件.....	141	10.5 利用 SQL 语句操作数据.....	202
7.3.1 Global.asa 文件概述.....	141	10.5.1 插入数据.....	202
7.3.2 动态在线人数统计实例.....	143	10.5.2 修改数据.....	203
练习七.....	144	10.5.3 删除数据.....	204
<b>第 8 章 ASP 的内置组件</b>		10.6 视图和索引.....	204
8.1 用 Browser 组件来获取客户浏览器 的信息.....	145	10.6.1 创建视图.....	204
8.2 FileSystem 组件来处理文件.....	146	10.6.2 创建索引.....	205
8.3 对文件夹的处理.....	153	10.7 Transact SQL 程序设计.....	208
8.4 处理驱动器并获取驱动器信息.....	154	10.7.1 基本语法及变量.....	209
8.5 利用 Content Linker 组件进行 网络导航.....	156	10.7.2 流程控制语句.....	210
8.6 Ad Rotator 组件来制作动态广告.....	158	练习十.....	213
练习八.....	160	<b>第 11 章 使用 ASP 数据库对象 操作数据库</b>	
<b>第 9 章 数据库基础知识</b>		11.1 使用 ADO 连接数据源.....	215
9.1 数据库系统的概念.....	163	11.2 使用 Connection 对象连接数据库.....	215
9.2 建立 Access 数据库.....	166	11.2.1 使用 Connection 对象打开和 关闭数据库.....	215
9.3 SQL Server 数据库简介.....	169	11.2.2 向浏览器以表格形式输出内容.....	217
9.4 SQL Server 2000 的使用.....	171	11.3 使用 RecordSet 对象.....	218
9.4.1 SQL Server 2000 的介绍.....	171	11.3.1 RecordSet 对象的属性和方法.....	218
9.4.2 SQL Server 2000 的安装.....	172	11.3.2 实现数据库的分页显示.....	219
9.4.3 如何建立数据库.....	176	11.4 使用 Command 对象.....	220
9.4.4 如何创建表.....	179	11.4.1 Command 对象方法概述.....	220
9.4.5 建立用户和设定密码.....	182	11.4.2 使用 Command 对象执行 SQL 语句.....	221
9.5 设置数据源.....	185	练习十一.....	222
练习九.....	187		

**第 12 章 网上考试系统**

12.1 系统总体设计	223
12.1.1 设计目标	223
12.1.2 设计思想	223
12.1.3 实现的功能	224
12.2 数据库设计	225
12.2.1 数据库规划	225
12.2.2 数据库表的设计	225
12.3 页面开发与程序设计	229
12.3.1 登录和注册页面	229
12.3.2 考试页面 (test.asp)	235
12.3.3 阅卷页面 (test_over.asp)	242
12.3.4 网上考试管理页面	246
本章小结	256

**第 13 章 图书管理系统**

13.1 总体设计	257
13.1.1 设计目标	257
13.1.2 设计思想	257
13.1.3 实现的功能	258
13.2 数据库设计	258
13.2.1 数据库规划	258
13.2.2 数据库表的设计	259
13.3 页面开发与程序设计	261
13.3.1 登录和注册页面 (index.htm)	261
13.3.2 图书管理主页 (index.asp)	263
13.3.3 图书入库页面 (tushuruku.asp)	266
13.3.4 图书信息修改页面 (tushumanage.asp)	269
13.3.5 图书注销页面 (tushuzhuxiao.asp)	278
13.3.6 图书类别管理页面 (tushuleibie.asp)	280
13.3.7 图书信息查询页面 (tushucx.asp)	284
13.3.8 图书管理员管理 (xtadminmanage.asp)	287

本章小结	291
------	-----

**第 14 章 网上购物系统**

14.1 购物系统总体设计	293
14.1.1 设计目标	293
14.1.2 实现的功能	293
14.2 数据库设计	294
14.2.1 数据库规划	294
14.2.2 数据库表设计	294
14.3 页面与程序设计	298
14.3.1 主页设计	298
14.3.2 用户注册页面	310
14.3.3 货物选购页面 (basket.asp)	313
14.3.4 收银台页面 (cash.asp)	317
14.3.5 购物受理页面 (cashsave.asp)	321
本章小结	328

**第 15 章 企业内部办公管理系统**

15.1 系统总体设计	329
15.1.1 设计目标	329
15.1.2 设计思想	330
15.1.3 实现的功能	330
15.2 数据库设计	332
15.2.1 数据库规划	332
15.2.2 数据库表的设计	332
15.3 页面开发与程序设计	335
15.3.1 登录页面 (default.asp)	335
15.3.2 系统主页面 (gensystem.asp)	339
15.3.3 通告发布 (newnotice.asp)	344
15.3.4 管理通告 (noticecontrol.asp)	346
15.3.5 浏览通告 (noticelook.asp)	351
15.3.6 个人档案编辑 (personinf.asp)	356
15.3.7 员工基本档案 (stafpersoninf.asp)	361
15.3.8 个人工作计划管理 (dayrep.asp)	364
本章小结	366

**附录 习题和答案**

# 第 1 章 初识 ASP

## 学习要点

- WWW 的工作原理
- 常用网络程序设计语言
- ASP 的初步了解
- IIS 的安装

## 1.1 网络程序设计语言的产生背景

Internet 是 20 世纪发展最快、规模最大、涉及面最广的科技成果。互联网起源于 1969 年美国国防部高级研究计划署协助开发的 ARPANET 网。1987 年，在美国国家科学基金会的推动下，将之主要从军事用途转向为科学研究和民事用途，这就形成了今天的 Internet 主干网雏形。Internet 是一个巨大的、全球范围的计算机网络，它是借助于现代通信和计算机技术实现全球信息传递的一种快捷、有效、方便的工具。

长期以来，人们只是通过传统的媒体（如电视、报纸、杂志和广播等）获得信息。由于计算机网络的发展，信息的获取变得非常及时、迅速和便捷。人们可以从网络上得到各种类别的信息，如文献期刊、产业信息、气象信息、论文检索等等。

最初的 WWW 网页都是用超文本标记语言 HTML (HyperText Markup Language) 来实现的。人们可以通过在上面放置各种 HTML 元素以实现文本、图像、超链接、表格等内容。

可是，它也存在一定的缺陷，HTML 制作的网页是静态网页。所谓静态网页，就是网页内容在设计时就定好了，如果想修改或维护网页，只能修改代码。试想，如果在网上放一个通讯录，由于地址、电话等经常变化，就不得不经常打开源代码修改，这样不仅麻烦，还容易出错。当今的 Web 已被赋予更丰富的内涵。现在，不仅需要 Web 提供所需的信息，还需要提供可个性化搜索的功能，如收发 E-mail、从事电子商务等。为实现以上功能必须使用更新的网络编程技术制作动态网页。所以，动态网页的研究和开发就成为需要迫切解决的问题了。

开始，人们利用 CGI (Common Gateway Interface, 通用网关接口) 来实现网上数据管理，CGI 是在服务器上运行的网关守护进程。根据客户端进行请求时所采用的方法，服务器负责收集有关客户端提供的信息，并根据客户端请求的 CGI 脚本程序文件名执行该脚本程序。CGI 脚本程序启动后，服务器将客户端的信息传递给此程序，该程序对客户信息处理后，将运行结果交给服务器，由服务器负责再向客户端传递，处理完毕后 CGI 程序结束运行。

但是 CGI 不仅开发不便捷, 执行起来效率很低, 代码的重用性也很低。因此, 产生一种使用简单、功能强大的网络程序设计语言就势在必行了。

在这种背景下, ASP、PHP 和 JSP 应运而生了。

## 1.2 WWW 的工作原理

互联网虽然已经有二十多年的历史。但若论广为人知, 却也不过是近几年的事。是什么原因使互联网变得如此流行呢? 其主要原因莫过于 World Wide Web (简称 WWW 万维网) 的出现与应用。

WWW 是 World Wide Web (环球信息网) 的缩写, 也可以简称为 Web, 中文名字为“万维网”。它起源于 1989 年 3 月, 由欧洲量子物理实验室 CERN (the European Laboratory for Particle Physics) 所发展出来的主从结构分布式超媒体系统。通过万维网, 人们只要通过使用简单的方法, 就可以很迅速方便地取得丰富的信息资料。现在, Web 服务器成为 Internet 上最大的计算机群, Web 文档之多、链接的网络之广, 令人难以想像。可以说, Web 为 Internet 的普及迈出了开创性的一步, 是近年来 Internet 上取得的最激动人心的成就。

WWW 中的信息资源主要由一篇篇的 Web 文档, 或称 Web 页的基本元素构成。这些 Web 页采用超级文本 (Hyper Text) 的格式, 即可以含有指向其他 Web 页或其本身内部特定位置的超级链接, 或简称链接。可以将链接理解为指向其他 Web 页的“指针”。链接使得 Web 页交织为网状。这样, 如果 Internet 上的 Web 页和链接非常多的话, 就构成了一个巨大的信息网。由于用户在通过 Web 浏览器访问信息资源的过程中, 无需再关心一些技术性的细节, 而且界面非常友好, 因而 Web 在 Internet 上一推出就备受欢迎, 并迅速得到了爆炸性的发展。

当用户从 WWW 服务器上取到一个文件后, 用户需要在自己的屏幕上将它正确无误地显示出来。由于将文件放入 WWW 服务器的人并不知道将来阅读这个文件的人到底会使用哪一种类型的计算机或终端, 要保证每个人在屏幕上都能读到正确显示的文件, 必须以某种各类型的计算机或终端都能“看懂”的方式来描述文件, 于是就产生了 HTML——超文本语言。

HTML (Hype Text Markup Language) 的正式名称是超文本标记语言。HTML 对 Web 页的内容、格式及 Web 页中的超级链接进行描述, 而 Web 浏览器的作用就在于读取 Web 网点上的 HTML 文档, 再根据此类文档中的描述组织来显示相应的 Web 页面。

WWW 采用的是客户/服务器结构, 服务器生成并传递文档, WWW 浏览器接收文档, 并在客户机对文档进行解释表达。首先, 用户通过指定的一个 URL, 发出通常是 HTTP 协议中的 GET 请求。该请求通过浏览器程序发出, 并被相应的 WWW 服务器上的 HTTP 守护程序 (httpd) 所接收, 客户机即与此服务器开始通信, 它们之间的请求与响应方式遵循 HTTP 协议。在大多数情况下, 客户机的请求希望获取某一 HTML 格式的文件, 于是 WWW 服务器搜索其文档空间, 若找到文件, 则将此 HTML 文件传给客户机。最后, 客户机的浏览程序将 HTML 文件解释后显示在用户的屏幕上, 然后断开与服务器的连接。

## 1.3 目前主要的网络程序设计语言

早期的 CGI 技术发展成熟而且功能强大,但由于编程困难、效率低下、修改复杂等原因,逐渐被新技术所取代,下面介绍几种目前备受关注的动态网页技术标准。

### 1.3.1 ASP

ASP (Active Server Pages) 动态网页,是微软公司推出的一种用以取代 CGI (Common Gateway Interface) 通用网关接口的技术。ASP 本身并不是一种脚本语言,它只是提供了一种使镶嵌在 HTML 页面中的脚本程序得以运行的环境。我们可以通过 ASP 结合 HTML 语言、ASP 指令和 ActiveX 元件以及数据库等方面的知识,使用自己的 Web 服务器创建并运行动态的交互式 Web 站点。用户端只要使用可执行 HTML 码的浏览器,即可浏览 Active Server Pages 所设计的网页内容。Active Server Pages 所使用的脚本语言 (VBScript, JScript) 均在 Web 服务器端执行,用户端的浏览器无需能够执行这些脚本语言。

目前,ASP (Active Server Pages) 技术已经相当普及,在 Internet 上几乎处处可以看到它的身影。应用 ASP 技术,无须复杂的编程,就可以开发出专业的动态 Web 站点。ASP 的出现,给以静态内容为主的 Web 带来了全新的动态效果。使您的站点具有更加灵活和方便的交互性,在 Internet 中实现信息的传递和检索越来越容易。

### 1.3.2 PHP

PHP (Hypertext Preprocessor) 超文本预处理,是一种 HTML 内嵌式的语言。而 PHP 独特的语法混合了 C、Java、Perl 以及 PHP 式的新语法。它可以比 CGI 或者 Perl 更快速地执行动态网页。

PHP 最初是在公元 1994 年 Rasmus Lerdorf 开始计划发展。在 1995 年以 Personal Home Page Tools (PHP Tools) 的名称开始对外发表第一个版本。在 1995 年中,第二版的 PHP 问世。在 1997 年中,开始了第三版的开发计划,PHP3 跟 Apache 服务器紧密结合的特性;加上它不断地更新及加入新的功能;并且它几乎支持所有主流与非主流数据库;它的源代码完全公开,在 Open Source 意识抬头的今天,它更是这方面的中流砥柱。PHP 的第四代 Zend 核心引擎已经进入测试阶段。整个脚本程序的核心大幅改动,让程序的执行速度满足更快的要求。而在 Internet 上它也支持了相当多的通讯协议 (protocol),包括了与电子邮件相关的 IMAP 和 POP3,网管系统 SNMP,全球信息网 HTTP、Apache 服务器以及其他网络的相关函数。PHP 不断有新的函数库加入和它不停更新的活力,使得 PHP 无论是在 UNIX 还是在 Win32 的平台上都可以有更多新的功能。它提供丰富的函数,使得在程序设计方面有着更好的支持。

### 1.3.3 JSP

JSP (Java Server Pages) 是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一

种动态网页技术标准。在传统的网页 HTML 文件 (\*.htm, \*.html) 中加入 Java 程序片段 (Scriptlet) 和 JSP 标记 (tag), 就构成了 JSP 网页 (\*.jsp)。

JSP 和 ASP 在技术方面有许多类似之处, 不过两者来源于不同的技术规范组织。ASP 一般只应用于 Windows NT/2000/2003 平台, 而 JSP 则可以在多个服务器平台上运行, 用 JSP 开发的 Web 应用是跨平台的, 既能在 Linux 下运行, 也能在其他操作系统上运行。JSP 将网页逻辑与网页设计和显示分离, 支持可重用的基于组件的设计, 使基于 Web 的应用程序的开发变得迅速简易, 所以被许多人认为是未来非常有发展前途的动态网站技术。

---

**提示:** ASP 与 JSP、PHP 一样, 都可以开发在服务器端执行的 Web 程序。但 ASP 有简单易学、易用, 能快速开发, 同时对服务器系统要求不高等特点。

---

## 1.4 ASP 入门

### 1.4.1 什么是 ASP

Active Server Page 是一种包含了使用 VB Script 或 JSript 脚本程序代码的网页。当浏览器浏览 ASP 网页时, Web 服务器就会根据请求生成相应的 HTML 代码, 然后再返回给浏览器, 这样浏览器端看到的就是动态生成的网页。ASP 是微软公司开发的代替 CGI 脚本程序的一种应用, 它可以与数据库和其他程序进行交互。ASP 的网页文件的格式是 .ASP。

ASP 的含义从字面上说, 包含三方面含义:

#### 1. Active

ASP 使用了 Microsoft 的 ActiveX 技术。ActiveX (COM) 技术是现在 Microsoft 软件的重要基础。它采用封装对象, 程序调用对象的技术, 简化编程, 加强程序间合作。ASP 本身封装了一些基本组件和常用组件, 有很多公司也开发了很多实用组件。只要您可以在服务器上安装这些组件, 通过访问组件, 您就可以快速、简易地建立自己的 Web 应用。

#### 2. Server

ASP 运行在服务器端, 这样就不必担心浏览器是否支持 ASP 所使用的编程语言。ASP 的编程语言可以是 VBScript 和 JScript。VBScript 是 VB 的一个简集, 会 VB 的人可以很方便的快速上手。然而 Netscape 浏览器不支持客户端的 VBScript, 所以最好不要在客户端使用 VBScript。而在服务器端, 则无需考虑浏览器的支持问题。Netscape 浏览器也可以正常显示 ASP 页面。

#### 3. Pages

ASP 返回标准的 HTML 页面, 可以正常地在常用的浏览器中显示。浏览者查看页面源文件时, 看到的是 ASP 生成的 HTML 代码, 而不是 ASP 程序代码。这样就可以防止别人抄袭程序。

由此我们可以看出, ASP 是在 IIS 下开发 Web 应用的一种简单、方便的编程工具。在了解了 VBScript 的基本语法后, 只需要清楚各个组件的用途、属性和方法, 就可以轻松编写出自己的 ASP 程序。

### 1.4.2 ASP 能做什么

ASP 是基于 Web 的一种编程技术, 它可以完成以往 CGI 程序的所有功能, 如计数器、留言簿、公告板、聊天室等等。一旦您熟练地掌握了它, 那么它就可以做任何您能想到的东西。ASP 能从 HTML 表单中收集用户资料, 能与数据库连接(当然包括从数据库中读出和写入数据), 甚至能用来接收和发送 E-mail。ASP 可以轻松地实现对页面内容的动态控制, 根据不同的浏览者, 显示出不同的页面内容。而浏览者一点儿都不会觉察出来, 就像专门为其制作的页面一样。

### 1.4.3 ASP 是怎样工作的

ASP 能在 Web Server (尤其是 IIS) 端集成 Script 语言到 HTML 网页的环境, 然后利用后者 (HTML 网页) 内含的 Script 程序代码取代原有的 CGI 和 ISAPI 程序, 以便执行原有的 CGI 功能。

IIS 的主要功能是支持静态 HTML 主页, 当用户向一个使用 IIS 的服务器提出请求后, 服务器将会从自己的硬盘或者内存中提出一个静态主页发送到用户的浏览器上, IIS 的主要作用就是作为用户浏览器到服务器硬盘的一些文件的一个有效接口。从这个意义上, IIS 和其他的 Web Server 没有什么区别, 任何 Web Server 的主要作用都是支持静态 HTML 主页。

要了解 ASP 是怎样工作的, 首先来看 Web Server 对 HTML 的支持过程和步骤。用户先在浏览器的地址栏中添上要访问的主页地址并按回车触发这个申请, 浏览器将申请发送到诸如 IIS 的 Web Server 上, Web Server 接收这些申请并根据.htm 或.html 的后缀名意识到这是 HTML 文件, Web Server 从当前硬盘或内存中读取正确的 HTML 文件然后将它送回用户浏览器, .html 文件将会被用户的浏览器解释并将结果显示在用户浏览器上。当然, 这种过程可能会更加复杂一些, 例如, Form 中的内容以及查询数据库的表单传递。在大多数情况下, 对于各种典型的 Web Server 来说, 这种过程是按部就班的, Web Server 接收文件申请并从内存或者硬盘中读取正确的文件。

Active Server Pages 改变了这一切, 在 IIS 仍然支持静态 HTML 主页的前提下, 利用 Active Server Pages 可以根据用户要求在 Web Server 上建立新的主页。

为了便于区分它与支持静态主页的区别, 我们来看一下它的工作步骤: 首先一个用户在浏览器的网址栏中添入 Active Server Pages 文件名称, 并回车触发这个 Active Server Pages 的申请, 浏览器将这个 Active Server Pages 的请求发送给 IIS, Web Server 接收这个申请要求并由于其.asp 的后缀意识到这是个 Active Server Pages 请求, Web Server 从硬盘或者内存中接收正确的 ASP 文件, Web Server 将这个文件发送到一个叫做 ASP.DLL 特定文件中, Active Server

Pages 文件将会从头至尾被执行并根据命令要求查询数据库,生成响应的静态主页,主页将被送回浏览器,用户浏览器解释执行并显示在用户浏览器上。其过程如图 1-1 所示。

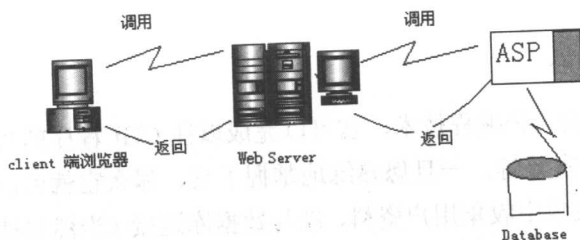


图 1-1 ASP 工作过程

可以看出,以上步骤已经进行了很大的简化,一个 ASP 页面并不一定每一次都重新编译解释,如果再次接受以前的那个请求而且没有任何变化,Active Server Pages 会从 Cache 中提出结果而不是再次运行。

#### 1.4.4 ASP 与 HTML

对于 Web Server 来说,Active Server Pages 与 HTML 有着本质的区别。因为 HTML 的开发是为了在计算机、网络及操作系统之间显示和传送信息。这种标准化的文档仅是一种简单的文本和标记,其中的图像和其他非文本的内容以单独文件形式存放,它们可以在各种网络上自由传送。由于信息的格式是固定的,其组成元素的含义也是在 HTML 中定义好的。对于一个“阅读器”或浏览器应用程序,用任何程序语言在任何平台或操作系统中相对来说是比较容易实现的。HTML 是不经任何处理送回给浏览器,而 Active Server Pages 的每一个命令都首先被用来生成 HTML 文件,因此,Active Server Pages 允许生成动态内容。

另一方面,对于浏览器来说,Active Server Pages 和 HTML 几乎是没有什么区别的,仅仅是后缀为 .asp 和 .htm 的区别,当我们在客户端提出 Active Server Pages 的申请后,我们的浏览器接受的是 HTML 格式的文件,因此它适用于任何浏览器。

#### 1.4.5 ASP 构建网站的特点

ASP 是由微软开发的服务器端直译软件。利用 ASP,只需要具备简单的 HTML 语法常识,再加上 JavaScript 或 VBScript 的一点基础,就可以利用 ASP 来创建出强大的交互式网页。下面介绍用 ASP 程序构建网站所具有的特性。

- (1) 使用 JScript 或 VBScript 等简易的 Script 语言,结合 HTML 代码,就可以快速地完成网站的应用程序。
- (2) 不需编译或链接就可以直接执行,并整合于 HTML 当中。
- (3) 使用一般文本编辑程序,如记事本就可以编辑设计。
- (4) 与浏览器无关。客户端只要使用可执行 HTML 代码的浏览器就可以浏览 ASP 的网

页内容。不存在浏览器兼容的问题，由于 ASP 程序是在服务器端运行的，当客户端浏览器浏览 ASP 网页时，服务器会将该网页文件重新解释一遍，并将生成的标准 HTML 文件发送给客户端浏览器，因为送出的是标准的 HTML 文件，当然不会存在浏览器兼容的问题了。

(5) ASP 与任何 ActiveX Script 语言完全兼容。除了可使用 VBScript 和 JavaScript 语言来设计外，可通过任何 Plug-in 的方式，使用厂商提供的其他脚本语言，如 REXX、Perl、Tcl 等。

(6) ASP 的源程序不会传到使用者的浏览器上，因而可以保护源程序不会外漏。

(7) 采用了面向对象技术。

(8) 可通过 ActiveX 服务器组件来扩充增加功能。ActiveX 服务器元件具有无限可扩充性。可以使用 Visual Basic、Java、Visual C++、COBOL 等编程语言来编写您所需要的 ActiveX Server Component。

### 1.4.6 ASP 的组成

由以前几节可知，ASP 程序可以包括许多的组成组件，除了 HTML 标记，还包括服务器端的 Script 程序代码：即位于<%...%>内的程序代码，客户端的 Script 程序代码：即位于<Script>...</ Script>内的程序代码。到底 ASP 文件可能包括哪些组件呢？答案不确定，但有 4 个元素可以说是最为重要的，它们是：HTML、ASP 的内建对象、ActiveX 服务器组件还有 Scripting Language。

#### 1. ASP 的脚本语言

在服务器端使用脚本语言是 ASP 的特色，那么什么是脚本语言呢？它和 HTML 以及编程语言有什么不同呢？

脚本语言介于 HTML 和 Java、C++或 Visual Basic 等编程语言之间。HTML 通常用于格式化和链接文本，而编程语言通常用于向机器发出一系列复杂的指令，脚本语言介于两者之间但它的函数与编程语言更为相象一些。

在服务器端使用脚本语言，需要在服务器端安装脚本引擎。脚本引擎是用于处理脚本的 COM（组件对象模型）对象。ASP 脚本引擎提供主机环境并把.asp 文件中的脚本交给脚本引擎处理。对于.asp 文件中使用的每种脚本语言，都要将它们相应的脚本引擎安装在 Web 服务器上。ASP 中缺省语言是 VBScript，所以您不用担心要去安装 VBScript 的脚本引擎，当您安装完 Active Server Pages 时，就已存在了。使用 JScript 也不必有这样的担心，但是要使用一些不太常用的脚本语言的话，却需要安装相应的脚本引擎。

ASP 使 Web 开发者可以用各种脚本语言编写程序而不用担心浏览器是否支持。实际上，在一个.asp 文件中就可以使用多种脚本语言，只要您在每段脚本程序开始处用 HTML 标签标明所用的脚本语言。

VBScript 是缺省的主脚本语言。主脚本语言用在定界符<%和%>之内，您可以在定界符

之内使用任何有效的 VBScript 命令，ASP 会按 VBScript 处理这些命令。如果您想在某一页中将一种脚本语言设为主脚本语言，只要在 .asp 文件开始处加上如下命令就可以了。

`<%@LANGUAGE=ScriptingLanguage%>`其中，`ScriptingLanguage` 代表您想设置的主脚本语言。如若您想将某一种脚本语言设为您的 Web 服务器上所有页的主脚本语言，您就需要动用功能强大的 RegisterEditor（注册编辑器，regedit.exe），ASP 注册参数的注册路径如下：

```
1 HKEY_LOCAL_MACHINE\SYSTEM
2 \CurrentControlSet
3 \Services
4 \W3SVC
5 \ASP
6 \Parameters
```

其中，注册表项 `DefaultScriptLanguage` 的缺省值是“VBScript”，将其改为欲设置的脚本语言即可。若想将 JScript 设为主脚本语言，则将注册表项 `DefaultScriptLanguage` 的值改为“JScript”就行了。在同一个 .asp 文件中混合使用不同的脚本语言是 ASP 迷人的特色之一。这样，您就可以发挥各种脚本语言在不同方面的优势，用不同的脚本语言编写不同的脚本程序来完成不同的工作。

脚本程序实际上就是一组执行特定任务的脚本命令。如果您经常要执行某一组特定任务，您就可以将之定义成程序，然后重复调用。程序要按特定的脚本语言的规则写在标签 `<SCRIPT>` 和 `</SCRIPT>` 之间。当然，如果您用主脚本语言写程序的话，写在定界符 `<%` 和 `%>` 之间也可以。如果您的程序只在某一页内调用的话，把它放在这一页内就行了，如果您在多页中都要调用某些程序的话，您可以把这些程序放在一个单独的 .asp 文件中，然后在每个要调用这些程序的 .asp 页中将那个 .asp 文件包含进来。要在 .asp 页中包含其他文件，使用以下的语法：

```
<!--#INCLUDE VIRTUAL|FILE="filename"-->
```

其中，`filename` 是您要包含的文件。关键字 `VIRTUAL` 和 `FILE` 任选其一，`VIRTUAL` 表示要包含的文件在一个虚拟目录内（该目录为 Web 共享目录），`FILE` 表示要包含的文件在一个当前文件相关目录内。例示如下：假设您有两个 Web 共享目录 `Web1` 和 `Web2`，目录 `Web1` 内有文件 `file11.asp` 和 `file12.asp`，目录 `Web2` 下有文件 `file2`，若要在文件 `file11.asp` 中包含 `file2.asp`，则将如下命令加入到文件 `file11.asp` 中：

```
<!--INCLUDE VIRTUAL="Web2/file2.asp"-->
```

若要在 `file11.asp` 中包含 `file12.asp`，则将如下命令加入到文件 `file11.asp` 中：

```
<!--INCLUDE FILE="file12.asp"-->
```

或者使用如下命令亦可：

```
<!--INCLUDE VIRTUAL="Web1/file12.asp"-->
```

程序写好了，怎么调用它呢？如果您用的脚本语言是 VBScript，只要一个简单的关键字 Call 再加上程序名就可以了。如果您要调用的程序有参数，要把参数用括号括起来。如果您省略关键字 Call，括号也要省略。下面例示了在同一个 ASP 页中用两种不同的脚本语言（VBScript 和 JScript）创建和调用程序。

```

1 <HTML>
2 <HEAD>
3 <TITLE>使用两种不同的脚本语言</TITLE>
4 </HEAD>
5 <BODY>
6 <% Call Echo %>           <!--调用 VBScript 程序>
7 <% Call PrintDate %>     <!--调用 JavaScript 程序>
8 </BODY>
9 </HTML>
10 <SCRIPT LANGUAGE=VBScript RUNAT=Server>
11 Sub Echo()
12 ... ..                   <!--用 VBScript 写的程序，内容省略>
13 End Sub
14 </SCRIPT>
15 <SCRIPT LANGUAGE=Jscript RUNAT=Server>
16 function PrintDate()
17 {
18 ... ..                   <!--用 JavaScript 写的程序，内容省略>
19 }
20 </SCRIPT>

```

## 2. ASP 的内建对象

ASP 功能的核心便是内建的对象（Request, Response, Server, Application, Session, ASPError, ObjectContent 对象），他们能够提供并处理一般 Web 应用程序的标准机制，例如：取得客户端信息、响应信息、保存变量、维护状态（State）/存取服务器公用程序、转承数据等，其行为就像是一般的 COM 对象，不过专属 ASP 所有。下面简单介绍各个内建对象的功能，具体的应用请参照第 4 章。

表 1-1 ASP 内建对象

对象名称	功能简述
Response	用来传输信息到客户端浏览器
Request	用来读取客户端浏览器的信息
Server	提供一些 Web Server 端的相关信息
Session	用来存储不同用户的信息
Application	用来存储所有用户共享的信息
ObjectContext	用于提交或撤销由 ASP 脚本初始化的事务