

站长札记



站长札记:

ASP 动态网页编程精讲

张 建 王海涛 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

内 容 简 介

作为“站长札记”系列中的一本书，本书系统地介绍了如何使用 ASP 技术开发网站应用程序。全书共 12 讲。第 1 讲至第 8 讲介绍了 ASP 技术的初步认识，ASP 技术的开发环境，ASP 技术涉及到的 HTML、JavaScript、VBScript 和 CSS 技术，ASP 的内量对象，ASP 的常用脚本对象，在 ASP 网页中调用数据库，在 ASP 中使用 XML，以及调试 ASP 程序等内容。第 9 讲至第 11 讲通过三个实例全面地介绍如何使用 ASP 技术进行网络应用程序的开发，以及在 ASP 网页中整合 AJAX 技术。第 12 讲介绍 ASP 程序的性能优化。

本书可以作为大专院校和各类培训机构的教材或者教学参考书，对于自学 ASP 的人员和开发 ASP 的人员，本书也具有很高的参考价值。

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

站长札记：ASP 动态网页编程精讲 / 张建等编著. —北京：电子工业出版社，2006.7

ISBN 7-121-02604-X

I.站... 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 (2006) 第 044334 号

责任编辑：吴 源

印 刷：北京天竺颖华印刷厂

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

北京市海淀区翠微东里甲 2 号 邮编：100036

经 销：各地新华书店

开 本：787×1092 1/16 印张：27.875 字数：690 千字

印 次：2006 年 7 月第 1 次印刷

定 价：40.00 元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。
联系电话：(010) 68279077。质量投诉请发邮件至 zltts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

ASP 技术是近年来编写网络应用程序经常用到的技术之一，使用它可以创建和运行动态的、交互的网络应用程序。使用 ASP 可以组合 HTML 技术、JavaScript 脚本语言、VBScript 脚本语言、CSS 技术和 XML 等创建交互的并基于 Web 文件的应用程序。

本书系统地介绍了构造网络应用程序涉及的方方面面技术，即使读者是一个初学者，通过阅读本书，也可以学会如何独立完成具有简单功能的网络应用程序。

本书共有 12 讲，具体内容如下：

- 对 ASP 技术的初步认识，以及同其他开发技术的异同。
- 如何配置 ASP 技术的开发环境。
- ASP 技术中涉及到的 HTML、JavaScript、VBScript 和 CSS 技术，并将这些技术运用到 ASP 程序中。
 - 使用 ASP 的内置对象，实现网络浏览器上数据的输入输出，网页文件中会话状态的维护，以及 Web 服务器端某些功能的调用。
 - 使用 ASP 的脚本对象，处理对 Web 服务器端驱动器、文件夹（目录）和文件的操作。
 - 使用 ASP 调用数据库，本书将 SQL Server 数据库管理系统作为主要的讲解对象。
 - ASP 技术中涉及到的 XML 技术，重点介绍如何使用 ASP 构造和解析 XML 数据。
 - 如何调试 ASP 程序，包括如何报错和断点跟踪。
 - 通过实现一个网上书店系统，全面地介绍如何使用 ASP 技术进行一个网络应用程序的开发。
- 小型 BBS 系统的实现。
- 以一个网上直播室的实例讲述如何在 ASP 的网页中整合 AJAX 技术。
- 讲述如何提高 ASP 程序的性能和 ASP 程序中可能出现的错误信息。

为了使读者能够深刻地理解所讲述的 ASP 技术，在每一讲中都介绍了很多实例。当介绍完所有的 ASP 技术以后，本书还通过实现三个项目介绍网络应用程序的开发过程，教会读者如何分析、设计和实现网络应用程序。其中讲述的需求分析、数据分析、数据库设计、目录文件设计到最后编码的整个过程，同实际应用中网站系统的实现是完全一致的。

由于程序员在编写程序时难免会遇到麻烦，如果不会调试，将是很苦恼的事情。网络应用程序的性能会直接影响网站的访问量和打开页面的时间，对于一个程序员来说只实现功能是不够的，还需要以高性能的方法来实现。而大多数的 ASP 书籍只讲述如何编写程序，却忽略了如何调试程序，以及如何提高程序性能。为了真正做到实用，本书对这些内容进行了介绍。

本书在编写过程中得到了欧阳宇、杨源、侯振鹏等朋友的支持，他们在技术方面给予了作者很大的帮助。在此，我们要对他们表示由衷的感谢。在全书的编写过程中，我们力求做到深入浅出的整体结构和严谨细致的内容，但由于水平有限，还是难免存在疏漏之处，欢迎读者朋友给予批评和指正。

目 录

第 1 讲 零点起步初识 ASP	2
1.1 初识 ASP	3
1.2 ASP 和 PHP、JSP 的性能对比	3
1.2.1 关于 ASP	3
1.2.2 关于 PHP	4
1.2.3 关于 JSP	4
1.3 正确认识 ASP 的安全性	4
1.4 ASP 网页的基本结构	5
1.5 学习 ASP 的方法	9
1.5.1 学习 ASP 的重要性	9
1.5.2 如何学好 ASP	9
第 2 讲 轻松搭建 ASP 开发环境	12
2.1 ASP 需要什么样的开发环境	13
2.2 在 Windows 98 下搭建 PWS 服务平台	13
2.2.1 安装 PWS	13
2.2.2 启动 PWS	15
2.3 在 Windows 2000/2003/XP 下搭建 IIS 开发平台	15
2.3.1 安装 IIS	16
2.3.2 启动 IIS 并配置网站	17
2.4 使用 Dreamweaver 创建网站	21
2.4.1 Dreamweaver 概述	21
2.4.2 使用 Dreamweaver 建立网站	21
2.5 使用 Visual Studio.NET 创建 ASP 文件	25
第 3 讲 熟悉 ASP 中的程序语言	28
3.1 HTML	29
3.1.1 HTML 的概述	29
3.1.2 HTML 的组成	29
3.1.3 HTML 文件基本结构	30
3.1.4 HTML 文件中的文本和段落	31
3.1.5 在 HTML 文件中显示图片	35
3.1.6 HTML 文件中的链接	36
3.1.7 HTML 文件中的表格	37
3.1.8 HTML 文件中的表单	39
3.2 VBScript	44
3.2.1 VBScript 概述	44
3.2.2 VBScript 基础知识	45

3.2.3	VBScript 的控制语句	48
3.2.4	过程及过程调用	55
3.2.5	函数及函数调用	55
3.2.6	VBScript 中的常用函数	56
3.2.7	VBScript 中建立类和对象	60
3.2.8	VBScript 中的错误处理	61
3.3	JavaScript	62
3.3.1	JavaScript 概述	62
3.3.2	JavaScript 基础知识	64
3.3.3	JavaScript 程序的构成	67
3.3.4	JavaScript 的对象	74
3.4	CSS 样式表单	77
3.4.1	CSS 样式表单概述	77
3.4.2	CSS 样式表单的创建	77
第 4 讲	ASP 自己的金牌工具——内置对象	82
4.1	Response 对象	83
4.1.1	Response 对象的方法	83
4.1.2	Response 对象的属性	87
4.1.3	Response 对象的集合	90
4.2	Request 对象	91
4.2.1	Request 对象的方法	91
4.2.2	Request 对象的属性	91
4.2.3	Request 对象的集合	91
4.3	Application 对象	96
4.3.1	Application 对象的方法	97
4.3.2	Application 对象的集合	97
4.3.3	Application 对象的事件	97
4.4	Session 对象	97
4.4.1	Session 对象的方法	98
4.4.2	Session 对象的属性	98
4.4.3	Session 对象的集合	99
4.4.4	Session 对象的事件	99
4.5	Sever 对象	99
4.5.1	Sever 对象的方法	99
4.5.2	Sever 对象的属性	100
4.6	ObjectContext 对象	100
4.6.1	ObjectContext 的方法	101
4.6.2	ObjectContext 的事件	101
4.6.3	使用 ObjectContext 对象的实例	101
4.7	Global.asa 文件	102

4.7.1	Application 事件	102
4.7.2	Session 事件	103
4.7.3	<OBJECT>声明	103
4.7.4	TypeLibrary 声明	103
4.8	ASP 内置对象的使用实例	104
第 5 讲	ASP 的常用脚本对象	112
5.1	ASP 的常用脚本对象	113
5.2	Dictionary 对象	113
5.2.1	创建 Dictionary 对象	113
5.2.2	Dictionary 对象的方法	113
5.2.3	Dictionary 对象的属性	116
5.3	FileSystemObject 对象	117
5.3.1	创建 FileSystemObject 对象	117
5.3.2	FileSystemObject 对象的方法	117
5.3.3	FileSystemObject 对象的属性	124
5.4	Drive 对象	125
5.4.1	创建 Drives 对象	125
5.4.2	Drive 对象的属性	125
5.4.3	使用 Drive 对象的实例	126
5.5	Drives 对象	128
5.5.1	建立 Drives 对象	128
5.5.2	Drives 对象的属性	128
5.6	File 对象	128
5.6.1	创建 File 对象	128
5.6.2	File 对象的方法	128
5.6.3	File 对象的属性	130
5.6.4	使用 File 对象的属性实例	131
5.7	Files 对象	132
5.7.1	创建 Files 对象	132
5.7.2	Files 对象的属性	132
5.8	Folder 对象	133
5.8.1	创建 Folder 对象	133
5.8.2	Folder 对象的方法	133
5.8.3	Folder 对象的属性	134
5.9	Folders 对象	134
5.9.1	创建 Folders 对象	134
5.9.2	Folders 对象的方法	135
5.9.3	Folders 对象的属性	135
5.10	TextStream 对象	135
5.10.1	创建 TextStream 对象	136

5.10.2	TextStream 对象的方法	136
5.10.3	TextStream 对象的属性	138
第 6 讲	在 ASP 网页中调用数据库	140
6.1	数据库概述	141
6.1.1	关系型数据库	141
6.1.2	数据库设计	143
6.2	Access 2000 数据库应用	146
6.2.1	Access 2000 数据库概述	146
6.2.2	Access 2000 数据库的基本操作	146
6.3	SQL Server 2000 数据库应用	149
6.3.1	SQL Server 2000 数据库概述	149
6.3.2	SQL Server 2000 数据库的基本操作	149
6.4	SQL 语言	164
6.4.1	SQL 简介	164
6.4.2	SQL 的基本语句	165
6.5	使用 ADO 操作数据库	169
6.5.1	连接数据库	169
6.5.2	对数据库执行 SQL 语句	171
6.5.3	处理数据库表记录集	172
第 7 讲	在 ASP 中使用 XML	176
7.1	XML 简介	177
7.1.1	XML 与 HTML 的区别	177
7.1.2	一个简单的 XML 文件	177
7.2	XML 文件的结构	178
7.2.1	文件的序言	178
7.2.2	文件的主体	178
7.2.3	文件内容的补充	180
7.3	XML DOM	181
7.3.1	XMLDOMDocument 对象	181
7.3.2	XMLDOMNode 对象	184
7.3.3	XMLDOMNodeList 对象	189
7.3.4	XMLDOMAttribute 对象	189
7.3.5	XMLDOMCharacterData 对象	190
7.3.6	XMLDOMComment 对象	190
7.3.7	XMLDOMText 对象	191
7.3.8	XMLDOMCDATASection 对象	191
7.3.9	XMLDOMELEMENT 对象	191
7.3.10	XMLDOMEntity 对象	192
7.3.11	XMLDOMEntityReference 对象	192
7.4	在 ASP 中管理 XML 文件	193

7.4.1	创建 XMLDOMDocument 对象	193
7.4.2	访问 XML 文件	193
7.4.3	创建一个 XML 文件并将其保存	193
7.4.4	将 XML 数据发送到客户端	195
第 8 讲	调试 ASP 程序	196
8.1	程序中的错误	197
8.1.1	语法类错误	197
8.1.2	运行时错误	197
8.1.3	逻辑类错误	198
8.2	调试服务器端程序	199
8.2.1	如何发现 ASP 程序中的语法类错误	199
8.2.2	调试运行时错误	202
8.2.3	设置断点并跟踪程序	202
8.2.4	如何解决程序中出现的逻辑错误	206
8.3	调试客户端程序	207
8.3.1	发现客户端代码的语法错误和运行时错误	207
8.3.2	在客户端代码中设置断点并跟踪程序	208
8.4	没有开发工具时如何调试程序	210
8.4.1	调试服务器端程序	210
8.4.2	调试客户端程序	210
第 9 讲	实现网上书店系统	212
9.1	系统分析	213
9.1.1	角色分析	213
9.1.2	基于角色的功能分析	213
9.2	数据分析	214
9.3	数据库设计	215
9.3.1	数据库的逻辑设计	215
9.3.2	数据库表的设计	216
9.4	网站的目录及文件设计	218
9.5	页面程序的具体实现	219
9.5.1	公用 ASP 文件的实现	219
9.5.2	公用 JS 文件的实现	223
9.5.3	会员注册和登录的实现	225
9.5.4	后台管理和登录后台管理	239
9.5.5	图书类别信息管理的实现	244
9.5.6	图书信息管理的实现	252
9.5.7	用户端查询浏览图书的实现	264
9.5.8	网站的首页	274
9.5.9	关于图书订单的实现	277
9.6	总结	297

第 10 讲 一个 BBS 系统的实现	298
10.1 系统分析	299
10.1.1 角色分析	299
10.1.2 基于角色的功能分析	299
10.2 数据分析	300
10.3 数据库设计	300
10.3.1 数据库的逻辑设计	300
10.3.2 数据库表的设计	301
10.4 网站的目录及文件设计	303
10.5 页面程序的具体实现	304
10.5.1 公用 ASP 文件的实现	304
10.5.2 公用 JS 文件的实现	305
10.5.3 会员注册和登录的实现	305
10.5.4 后台管理和登录后台管理	305
10.5.5 论坛类别管理的实现	310
10.5.6 论坛信息管理的实现	318
10.5.7 论坛主页和搜索帖子	327
10.5.8 用户浏览帖子, 会员发新帖, 回帖和版主删除帖子	333
10.6 总结	344
第 11 讲 基于 ASP 的 AJAX 技术实现网上直播室	346
11.1 AJAX 技术简介	347
11.1.1 AJAX 的特点	347
11.1.2 AJAX 涉及的技术	347
11.1.3 XMLHTTP 对象	347
11.2 需求分析	349
11.2.1 角色分析	349
11.2.1 基于角色的功能分析	349
11.3 数据分析	350
11.4 数据库设计	350
11.4.1 数据库的逻辑设计	350
11.4.2 数据库的表设计	350
11.5 网站的目录及文件设计	351
11.6 页面程序的具体实现	352
11.6.1 公用 ASP 文件的实现	352
11.6.2 公用 JS 文件的实现	354
11.6.3 管理员登录的实现	355
11.6.4 关于比赛信息管理的实现	357
11.7 总结	380
第 12 讲 ASP 程序的性能优化	382
12.1 ASP 程序的代码优化	383

12.1.1	尽量减少对 Session 对象和 Application 对象的使用	383
12.1.2	及时关闭不使用的对象	383
12.1.3	合理使用 Include 文件	383
12.1.4	尽量使用系统函数代替自己编写的函数	383
12.1.5	减少动态数组的使用	384
12.1.6	养成提前声明变量的习惯	384
12.1.7	减少 Response.write 的使用	384
12.1.8	使用一个 ASP 文件完成一个操作	385
12.1.9	尽量使用 Server.Transfer 代替 Response.Redirect	385
12.1.10	对 URL 地址进行编码	385
12.1.11	在循环中避免进行字符串连接	385
12.1.12	避免使用 Request ("name") 的方式获得参数值	386
12.2	数据库优化	386
12.2.1	不要随便使用“SELECT * from”语句	386
12.2.2	尽可能使用数据库中的存储过程	386
12.2.3	注意游标的使用方法	386
12.2.4	不要打开无用的独立记录集	387
12.2.5	及时关闭打开的记录集对象以及连接对象	387
12.2.6	比较取得记录集中数据的方法	387
12.2.7	使用字符串建立 SQL 查询	388
12.2.8	where 语句中表之间的关联条件放在后面	388
12.2.9	为了速度放弃数据库结构的规范化	388
12.2.10	为每个表都建立索引	388
12.2.11	分页显示用存储过程来实现	388
12.2.12	数据库中尽量不要存储图片、文件等数据	388
12.3	网页优化	389
12.3.1	减少网页中的图片及其他多媒体信息	389
12.3.2	将有关下载的文件放在一个 FTP 服务器中	389
12.3.3	把动态页面替换成静态页面	389
附录 A	ASP 函数参考	390
附录 B	数据库操作错误信息	422
附录 C	VBScript 错误信息	427
附录 D	JavaScript 错误信息	430
参考文献		433

站长札记：ASP 动态网页编程精讲

张 建 王海涛 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

第 1 讲

零点起步初识 ASP

ASP 技术从一出现到现在一直受到很多网页开发人员的关注, 虽然 ASP 出现后也曾出现了很多 Web 开发技术, 但是由于它简单易用, 非常适用于一般的中小型网站, 所以很多开发人员都使用它来实现 Web 程序。本讲将简单介绍 ASP 技术的一些概念, 以及同其他 Web 开发技术的异同。同时本讲中还会介绍一个 ASP 程序实例, 来简单介绍 ASP 程序的结构。

本讲要点:

- 初识 ASP
- ASP 和 PHP、JSP 的性能对比
- 正确认识 ASP 的安全性
- ASP 网页的基本结构
- 学习 ASP 的方法



1.1 初识 ASP

网页就是我们平时在浏览器中看到的页面。目前的网页有两种，一种是静态网页，另一种是动态网页。什么是静态网页和动态网页呢？如果浏览同一个网页文件，每次打开后看到的页面内容都一样，那它就是静态的，如果每次打开后它的内容有可能是不一样的，那么它就是动态的。以 htm 为扩展名的网页是静态的，而以 asp、aspx、jsp、php 为扩展名的网页就是动态的。

ASP 的全称是 Active Server Pages，它是微软公司推出的一个专门开发动态网页的产品。运行 ASP 程序是通过一个名为 asp.dll 的文件来进行的，这个文件可以对 ASP 文件中的服务器端代码进行解析，生成最终看到的页面。由此可见 ASP 是解释执行的，而用来解释它的文件就是 asp.dll。

一个好的 ASP 程序可能由很多技术组成，但是归根结底无非也就是客户端程序技术和服务器端程序技术。一般情况下，客户端程序技术涉及到 HTML、JavaScript、CSS，服务器端程序技术涉及到 VBScript 和一些数据库技术。我们只要在编程时慢慢积累，这些技术自然都会学会。以上说的这些技术在后面的篇幅中都会介绍到，但主要侧重于服务器端程序技术。

写好了网页程序后，如何才能发布呢？发布一个网站需要一个 Web 服务器作为平台。为了满足这个需要，微软公司推出了 IIS(Internet Information Server)和 PWS(Personal Web Server)这两种 Web 服务器来提供发布 Web 网站的功能。PWS 是面向个人的 Web 服务器，一般情况下，使用 Windows 98 操作系统的用户大多会选择它，它是一个功能简单的小型 Web 服务器。在本书中，推荐您使用 IIS 服务器，它功能强大、操作简单、性能稳定。如果您使用的是 Windows 2000 Server 或者 Windows 2000 Advance Server 或者 Windows Server 2003，那么这个服务器就已经集成在操作系统中了。关于如何使用这个服务器在后面将会详细地介绍。



1.2 ASP 和 PHP、JSP 的性能对比

1.2.1 关于 ASP

ASP 是微软公司用来开发动态网页的产品，它主要采用脚本语言 VBScript（或 JavaScript）作为自己的开发语言。它可以通过在 HTML 代码中嵌入服务器端脚本的方式开发出动态网页程序。

它的特点如下：

(1) 简单易学、无需编译，使用任何文本编辑工具都可以编写，如记事本（但是希望用户使用专业的编辑工具如 .Net 开发工具、EditPlus、Dreamweaver 等，因为这样可以提高开发速度）。

(2) 脚本语言 VBScript、JavaScript 可以在服务器端执行。客户端的脚本可以通过服务器端的脚本来生成。

(3) ASP 能与任何 ActiveX Scripting 语言兼容。除了可使用 VBScript 或 JavaScript 语言来设计外，还可通过嵌入的方式，使用由第三方所提供的其他脚本语言，比如 REXX、Perl、Tcl 等。脚本引擎是处理脚本程序的 COM（Component Object Model）对象。

1.2.2 关于 PHP

PHP 是一种完全免费的、跨平台的、服务器端的嵌入式脚本语言。它大量地借用了 C、Java 和 Perl 语言的语法特点。PHP 使 Web 开发者能够快速地写出动态产生页面。

它的特点如下:

(1) PHP 是完全免费的,可以在 PHP 官方站点 (<http://www.php.net>) 自由下载。还可以获得源代码并且在其中加入自己的东西。

(2) 目前 PHP 可以与许多数据库相连接。PHP 与 C 语言是现在绝佳的组合。虽然 PHP 支持目前流行的大多数数据库,但是 PHP 提供的数据库接口并不统一,比如对 Oracle、MySQL、Sybase 的接口彼此都不一样。MySQL 数据库也是完全免费的。

1.2.3 关于 JSP

JSP 的全称是 Java Server Page,它是 Sun 公司推出的网站开发产品。JSP 技术很容易整合到多种应用体系结构中。作为采用 Java 技术家族的一部分,以及 J2EE 的一个成员,JSP 技术能够完成高度复杂的基于 Web 的应用。

它的特点如下:

(1) 一次编写,各处执行。

因为 JSP 是 J2EE 平台的一部分,所以它也拥有 Java 的这一特点。

(2) JSP 将内容和显示进行分离。

JSP 页面产生的动态内容可以封装在标识和 JavaBeans 群组件中,在服务器端,JSP 引擎可以解释 JSP 标识,产生所请求的内容将结果以 HTML 页面的形式发送回浏览器。在封装 JSP 标识和 JavaBeans 群组件的同时,页面设计者可以使用 HTML 和标识或者 JavaBeans 来编辑页面,在编辑过程中不影响内容的产生。这样的好处是隐藏了网页逻辑运算部分的代码,同时可以使网页的页面设计和逻辑程序设计同时进行。

(3) 基于组件的开发模式提高了程序的开发速度。

绝大多数 JSP 页面依赖于可重用且跨平台的组件(如 JavaBeans 或者 Enterprise JavaBeans)来解决程序中复杂的处理。开发人员可以共享和交换彼此的组件,这样可以加快程序开发过程。

介绍完 ASP、PHP、JSP 后,下面对这三种 Web 开发产品做一个比较。

- 在运算速度上,JSP 比另外两个快很多,因为它不是解释执行的。其次是 PHP,最慢的是 ASP。

- 在学习的难度上,ASP 是最简单的,其次是 PHP,最复杂的是 JSP。因为 JSP 属于 J2EE 的一部分,其功能非常强大,学习的东西自然会很多。

- 从功能上看 JSP 是最强大的,适合大型复杂的 Web 应用,其他两个适合中小型 Web 的应用。

- 从使用人群上来看,ASP 的使用者是最多的,主要是由于它简单易学。其他两个使用人数上相近。

1.3 正确认识 ASP 的安全性

现在来讨论一下 ASP 程序的安全问题,首先谈一谈 ASP 的优点。

(1) 浏览网页的用户看不到 ASP 的源程序, ASP 的源程序保存在 Web 服务器中。从 ASP 的工作原理上看, ASP 在服务器端执行并解释成标准的 HTML 语句, 再传送给客户端浏览器。所以一个 ASP 文件中的服务器端代码, 对于浏览网页的用户来说是不可见的, 这也就提高了 ASP 程序的安全性。

(2) IIS 支持虚拟目录, 虚拟目录可以隐藏站点目录结构的重要信息。如果在 Web 页中使用物理路径, 将暴露有关站点目录的重要信息, 这容易导致系统受到攻击。另外, 当 Web 页面被放置于虚拟目录时, 就可以对目录设置不同的属性, 如: Read、Excute、Script。Read 访问表示将目录内容从 IIS 传递到浏览器。Excute 访问则可以在该目录内执行可执行的文件。建议读者在设置 Web 站点时, 将 HTML 文件同 ASP 文件分开放置在不同的目录下, 然后将 HTML 子目录设置为“读”, 将 ASP 子目录设置为“执行”, 这不仅方便了对 Web 的管理, 而且提高了 ASP 程序的安全性, 防止了程序内容被客户所访问。

虽然微软提供了上面的安全性, 但是 ASP 仍然存在一些漏洞, 黑客可以通过这些漏洞看到程序代码或者进入你的系统。安全因素是制约因特网应用技术发展的一个重要因素。

1.4 ASP 网页的基本结构

一个 ASP 页面基本上由三部分组成: HTML、JavaScript 客户端脚本、和 VBScript 服务器端脚本。

一般编写一个 ASP 网页时, 首先利用页面开发工具(如 Dreamweaver)生成页面, 这时生成的是 HTML 内容。然后编写有关 JavaScript 客户端脚本, 这些脚本很可能是和客户端的一些标签的事件挂钩的(如鼠标在表格上移动时表格的颜色发生变化)。最后编写的是 VBScript 服务器端脚本, 主要是服务器上的逻辑运算(如从数据库中提取数据并传送到客户端)。当然编写的顺序不是固定的, 这要根据自己的习惯和具体的情况来定。

下面将用一个小例子介绍一下 ASP 网页的构成, 这个小例子包括了 HTML, JavaScript 客户端代码和 VBScript 服务器端代码。

文件 E1-1.asp 的代码如下:

```
<%@ Language=VBScript %>
<html>
<head>
</head>
<title>实例</title>
<body>
    <table align="center" border="0">
        <tr>
            <td height="20"></td>
        </tr>
        <tr>
            <td align="right">服务器现在的时间是: </td>
```

```

        <td><%=Now%></td>

    </tr>

    <tr>

        <td height="20"></td>

    </tr>

    <tr>

        <td align="right">其中: </td>

        <td>

            <select id="sport">

                <option value="<%=Year (Now) %>">年</option>
                <option value="<%=Month (Now) %>">月</option>
                <option value="<%=Day (Now) %>">日</option>
                <option value="<%=Hour (Now) %>">时</option>
                <option value="<%=Minute (Now) %>">分</option>
                <option value="<%=Second (Now) %>">秒</option>

            </select>

        </td>

    </tr>

    <tr>

        <td height="20"></td>

    </tr>

    <tr>

        <td align="center" colspan="2">

            <input type="button" id="btn_Show" onclick="Show ( )" value="显示数据">

        </td>

    </tr>

</table>

</body>

</html>

<script language="javascript">
function Show ( )
{
    alert (sport.options[sport.selectedIndex].text + ": " + sport.value);
}
</script>

```


图 1-1 是程序运行的结果，当选择“年”、“月”、“日”、“时”、“分”、“秒”，然后再单击【显示数据】按钮时，将会弹出与对话框当前时间相对应的值。

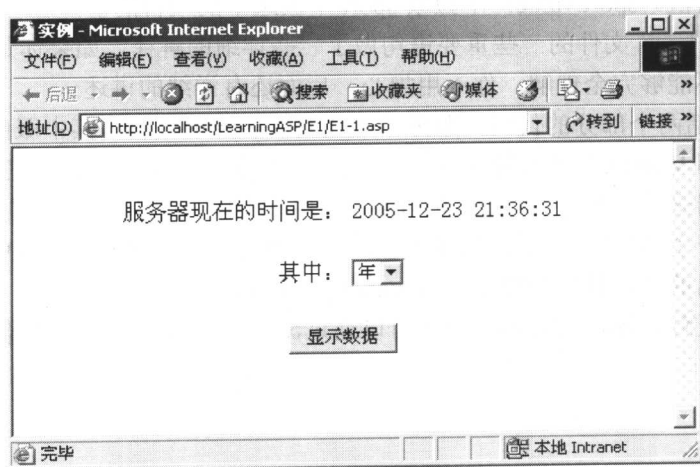


图 1-1 运行的结果

程序讲解：

- (1) `<%@ Language=VBScript %>`: 指出 ASP 文件的服务器端脚本是 VBScript。
- (2) 符号 `<%` 和 `%>` 中间的内容就是服务器端的脚本，这是服务器区分什么是服务器脚本的约定。
- (3) `<% = expression %>`: 将“expression”的值输出（即将 expression 写到 ASP 输出流中）。
- (4) `<%=Now%>`: 输出当前日期和时间。其中 Now: 返回当前的日期和时间。
- (5) `<%=Year (Now) %>`: 输出当前日期和时间中的年值。其中 Year (Date): 返回年值的整数 (Date 是日期参数)。
- (6) `<%= Month (Now) %>`: 输出当前日期和时间中的月值。其中 Month (Date): 返回月份的整数 (Date 是日期参数)。
- (7) `<%= Day (Now) %>`: 输出当前日期和时间中的日期值。其中 Day (Date): 返回日期的整数 (Date 是日期参数)。
- (8) `<%= Hour (Now) %>`: 输出当前日期和时间中的小时值。其中 Hour (Date): 返回小时的整数 (Date 是日期参数)。
- (9) `<%= Minute (Now) %>`: 输出当前日期和时间中的分钟值。其中 Minute (Date): 返回分钟的整数 (Date 是日期参数)。
- (10) `<%= Second (Now) %>`: 输出当前日期和时间中的秒值。其中 Second (Date): 返回秒的整数 (Date 是日期参数)。
- (11) `<input type="button" id="btn_Show" onclick="Show ()" value="显示数据">`: 显示按钮，当单击此按钮时将运行 javascript 函数 Show ()。
- (12) `<script language="javascript">`与`</script>`之间的内容为客户端脚本，脚本的语言是 javascript。
- (13) `function Show () {}`: 定义函数的名字为 Show，{}中间的内容为函数运行的语句。