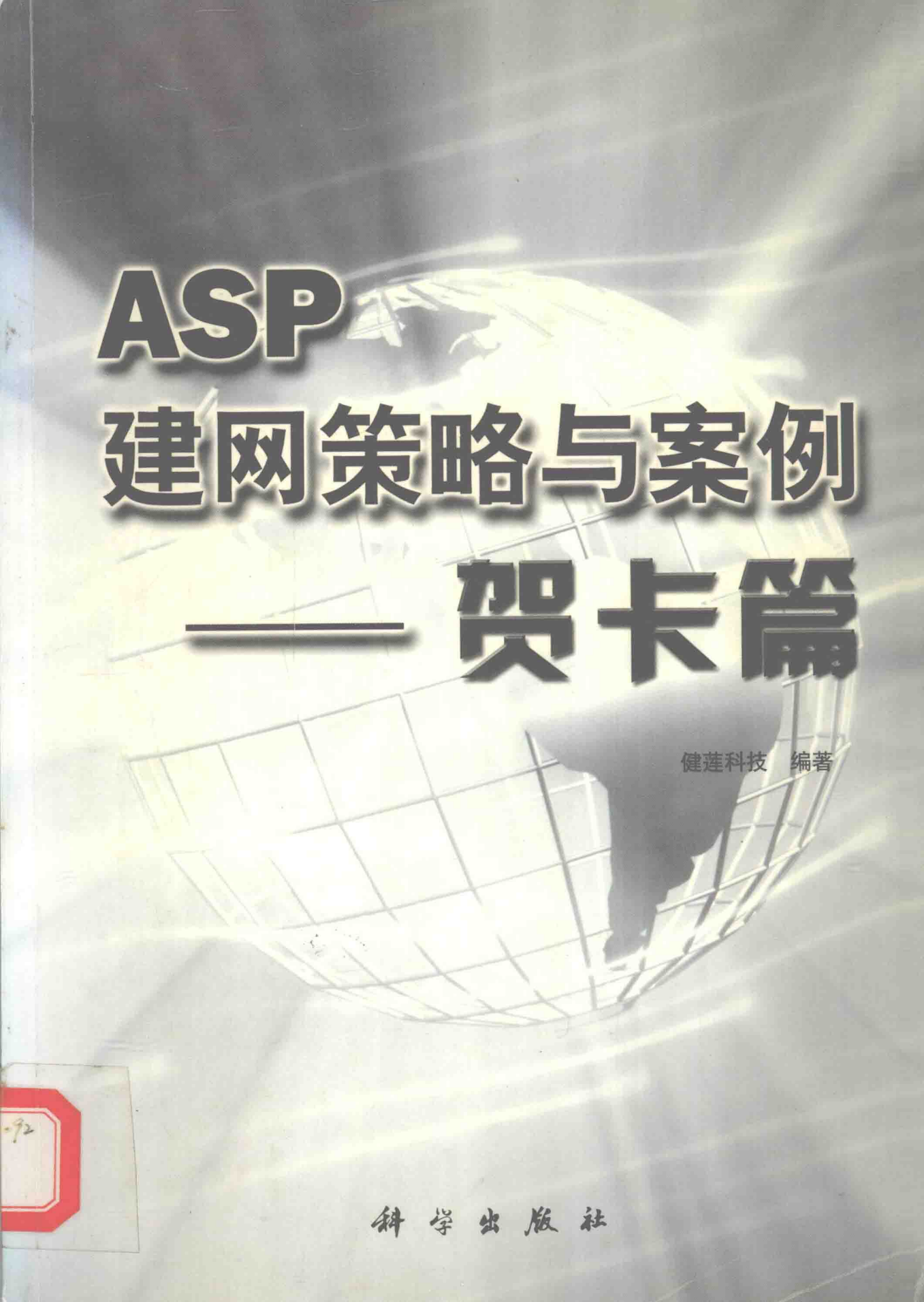




ASP 建网策略与案例 —— 贺卡篇

健莲科技 编著

科学出版社



ASP 建网策略与案例

——贺卡篇

健莲科技 编著

科学出版社

2002

内 容 简 介

ASP 是微软公司在 Web 领域上的一次突破,它打破了以往只能由专业人士开发的束缚,使广大爱好者也能独立地构建自己的动态网站。本书根据作者的实际开发经验,通过实例,由浅入深,循序渐进地介绍 ASP 的各个知识点,通俗易懂;同时,通过对完善专业性网站的人才功能模块、影音功能模块虚拟社区功能模块以及贺卡功能模块的代码解析,使每位读者根据不同模块、不同功能的实现,触类旁通,举一反三,进而提高设计水平,建设优秀的网站。

介绍 ASP 的书很多,本书可以说是出类拔萃。与其他书籍的最大不同之处在于本书所依托的代码基础是一套功能强大代码完善的专业系统,其单方向的专业研究深度同那些泛泛讲解代码的书籍不同。通过阅读本书,相信读者的思维将会得到开阔,制作出专业的 ASP 网站。

本书是为 Web 程序开发人员编写的。

图书在版编目(CIP)数据

ASP 建网策略与案例:贺卡篇 / 健莲科技编著. —北京:科学出版社, 2002

ISBN 7-03-009944-3

I. A... II. 健... III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2001) 第 091114 号

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2002 年 2 月第 一 版 开本: 787×1092 1/16

2002 年 2 月第一次印刷 印张: 17 1/4

印数: 1—4 000

字数: 409 000

全套定价: 116.00 元 (共四册)

(如有印装质量问题,我社负责调换<环伟>)

前 言

互联网正在融入我们的生活，影响和改变着我们的生活。网络提供给我们的不只是一个获取信息的来源，而且还是一个可以相互交流的空间，网络社区正是一种供人们进行交流的网上虚拟空间。同时，Internet 的迅速发展，已超出了我们的想象，而作为 Internet 主要应用的 WWW 技术的发展更是日新月异。如何创建基于 WWW 的网站，有很多可供选择的技术与解决方案，而运用 ASP 来实现，无疑是最方便快捷的。

基于 Web 的网站制作技术，在客户端执行的有 HTML，XML 和脚本语言 VBScript、JavaScript 等技术，在服务器端执行的有 CGI，ASP，JSP 等可供选择。Microsoft Active Server Pages (ASP) 是脚本语言、ActiveX 组件及 HTML 语言等的综合，它提供了一种服务器端脚本编写环境和服务器执行指令的环境，使用 ASP 可以创建和运行动态、交互的 Web 服务器应用程序，接收来自客户端的请求，并对后端数据库进行存取操作，然后将处理结果返回到客户端浏览器。

运用 ASP 进行设计与开发网站，具有方便快捷的优点，同时方便系统的管理和维护。ASP 在国内的应用已相当普遍，大部分的动态网站运用了 ASP 技术。然而在学习 ASP 及其相关的 HTML、脚本语言之后，如何运用它来真正实现自己的动态网站，对很多人来说仍然是比较困难的事情。本书正是致力于给读者一个网站开发与设计的系统解决方案，使读者不仅掌握 ASP 设计的技术要点，而且可以真正完成网站的系统开发。

本书采用了一个贺卡模块的源代码作为导引，在介绍基本的运行环境与 ASP 语言知识的基础上，将贺卡系统模块化，构建了贺卡调用、显示贺卡、搜索贺卡、邮件回复、贺卡发送等子系统，贺卡各模块依照系统设计（包括功能设计、文件结构设计与数据库规划）、源代码解析、范例演示及总结升华的结构进行介绍与分析。

本书的源代码可在 www.matrixsoft.net/ebook/ 上下载。读者在阅读本书之前，需要具备 HTML，JavaScript/VBScript 和数据库操作及网页编辑软件使用等基本知识，在书中将不对这些技术进行详细介绍，读者可以参阅相关书籍。在使用本书的过程中，读者在关注源代码及其解析的同时，应注重系统的开发与设计方法。

本套书主要由张明主编，陈宏传、李岩、红欣、李燕渤、辛珠、宁媛、张羿、杨捷、杨静、刘淑兰、姚辉、刘韬、杨松鹤、刘皓、姬宪军、姚玉英、吴秀霞、辜春明、苏少辉参与了本书的编写与整理工作。由于作者水平有限，时间仓促，错误在所难免，敬请读者批评指正。您可以通过以下方式和我们取得联系：g_auswer@sina.com。

作 者

目 录

第 1 章 构建电子贺卡网站开发与运行环境	1
1.1 选择 ASP	2
1.2 ASP 简介	2
1.2.1 关于 ASP	2
1.2.2 ASP 的新功能	3
1.2.3 创建 ASP 页	4
1.2.4 使用脚本语言	7
1.3 Web 服务器基础	9
1.3.1 硬件要求	10
1.3.2 软件要求	10
1.3.3 选择服务平台	11
1.4 安装与配置 PWS	11
1.4.1 安装 PWS	12
1.4.2 设定 PWS 4.0	16
1.5 IIS 的安装与配置	17
1.5.1 安装 IIS 5.0	17
1.5.2 设置 IIS 5.0	20
第 2 章 网络数据库基础	24
2.1 数据库概述	25
2.1.1 数据库概念	25
2.1.2 数据管理系统	25
2.1.3 Web 数据库简介	26
2.2 流行数据库应用系统简介	26
2.2.1 常见的桌面数据库产品	27
2.2.2 常见的大型数据库产品	27
2.3 安装和使用 Access	28
2.3.1 选择 Access	28
2.3.2 安装 Access 2000	29
2.3.3 Access 2000 的简单基本操作	30
2.4 连接网络数据库	31
2.4.1 基本知识	31
2.4.2 配置数据库驱动程序 DSN	32
2.5 用 Access 制作数据库网页	36

第3章 网络编程基础	42
3.1 ASP 内置对象	43
3.1.1 ASP 内置对象综述	43
3.1.2 Request 对象	43
3.1.3 Response 对象	48
3.1.4 Application 对象	51
3.1.5 session 对象	53
3.1.6ObjectContext 对象	55
3.2 VB Script 基础	55
3.2.1 VBScript 数据类型	56
3.2.2 VBScript 变量	56
3.2.3 VBScript 运算符	58
3.2.4 使用条件语句	58
3.2.5 使用循环语句	60
3.2.6 VBScript 过程	63
3.2.7 VBScript 常用函数	65
第4章 构建简单贺卡调用系统	74
4.1 设计前的思考	75
4.2 预备知识——利用 ADO 访问数据库	75
4.2.1 Connection 对象的应用	75
4.2.2 Recordset 对象的应用	77
4.2.3 Command 对象的应用	79
4.3 创建数据库表单	80
4.4 设计简单调用表单	81
4.4.1 操作数据库	81
4.4.2 设计前端输出页面	81
4.4.3 设计调用程序	82
4.5 设计分页显示功能	88
4.5.1 简单流程图	88
4.5.2 准备数据库	89
4.5.3 编写分页代码	89
4.6 设计自动识别贺卡格式功能	96
第5章 建立完整贺卡显示系统	100
5.1 设计前思考	101
5.2 request 对象应用详述	101
5.3 简单分类调用程序设计	106
5.3.1 建立新的数据库表单	106
5.3.2 文件架构	107
5.3.3 页面设计及代码解析	107

5.3.4	效果演示	116
5.4	贺卡高级分类管理功能编写	117
5.4.1	设计前的思考及数据库规划	117
5.4.2	文件架构及前端页面设计	118
5.4.3	效果演示	131
第 6 章	建立站内贺卡搜索系统	133
6.1	设计前思考	134
6.2	文件架构及流程	134
6.3	输入表单设计	134
6.3.1	首页设计	135
6.3.2	search.asp 设计	138
6.4	编写后台功能代码	139
6.4.1	SELECT 高级操作技巧	139
6.4.2	编写功能代码	140
6.5	范例演示	152
第 7 章	贺卡排行与推荐功能	154
7.1	设计前思考	155
7.2	规划和更新数据库	155
7.3	SELECT 语句	156
7.3.1	SELECT 选择列表	157
7.3.2	SELECT 语句中的通配符	158
7.3.3	FROM 子句	159
7.3.4	使用 WHERE 子句设置查询条件	160
7.3.5	用 ORDER BY 子句对结果排序	161
7.4	排行及推荐页面设计	161
7.5	贺卡排行程序解析	162
7.6	贺卡推荐程序解析	163
7.7	范例演示	178
第 8 章	建设网站在线	179
8.1	设计前思考	180
8.2	文件架构	180
8.3	数据库设计	181
8.4	前端页面设计	182
8.5	后台程序设计	183
8.5.1	站长登录验证	183
8.5.2	选择和更新栏目	184
8.5.3	游标的使用	185
8.5.4	站长栏目管理	189

8.5.5 贺卡信息管理	195
8.6 范例演示	204
第9章 建立邮件回复系统	208
9.1 设计前思考	209
9.2 Jmail 使用知识	209
9.2.1 何为 ASP 组件	209
9.2.2 认识和使用 Jmail	209
9.3 随机密码设计	216
9.4 文件架构	216
9.5 设计输入表单	216
9.6 设计后端 ASP 程序	218
9.7 范例演示	223
第10章 建立完整贺卡发送系统	225
10.1 设计前思考	226
10.2 文件架构	226
10.3 数据库设计	227
10.4 定制贺卡页面设计	228
10.5 贺卡定制及预览程序解析	229
10.6 贺卡发送功能设计	245
10.7 范例演示	250
第11章 建立贺卡收取功能	253
11.1 设计前思考	254
11.2 文件架构	254
11.3 页面设计	254
11.4 程序解析	255
11.5 范例演示	260
第12章 构建完整贺卡系统	263
12.1 完整网站设计思考	264
12.2 完整网站文件架构	264
12.3 网站首页设计	265
12.4 网站设计规范性	265
12.4.1 系统设计规范	265
12.4.2 网页建立规范	266
12.4.3 目录设置规范	267
12.5 保障系统安全	267

第1章 构建电子贺卡网站

开发与运行环境

本章技术要点

- ◆ ASP 概念
- ◆ Web 服务器
- ◆ IIS/PWS 安装及设置

随着网络生活化的不断加深，电子贺卡逐渐进入了我们的生活，从通过 E-mail 互相传送问候，到专业电子贺卡软件问世，再到如今鳞次栉比的电子贺卡网站，电子贺卡似乎已经成了网络生活中不可缺少的一部分。

制作一个专业的电子贺卡网站，必须选择一种开发语言及其开发环境，网络 Web 服务器是网络应用程序的运行平台，它与客户端浏览器交互并执行对后台数据库的操作，然后进行处理，并将结果返回给客户端浏览器。在这个过程中，ASP 程序运行于 Web 服务器并利用 ADO 存取数据库。由此可见，要开发网络服务程序，构建其运行环境是首当其冲的问题。

本章将简要介绍网络程序概念、网络开发使用语言和微软视窗系列操作系统下 Web 服务器的建立与配置方法等内容。其中重点是了解网络程序开发的概念、Web 服务器安装及配置方法。

1.1 选择 ASP

随着互联网的迅速发展,它逐步深入甚至开始左右人们的生活,各种围绕着互联网展开的应用层出不穷,电子贺卡就是其中之一。与传统的贺卡相比,电子贺卡在高效、快速、经济、节约能源等方面都有不可比拟的优势。下面我们就从基础做起,一步一步建立一个具有专业水准的电子贺卡网站。

首先,选择一种开发语言。为了最大限度地开发和利用网络,人们制定了各种网络应用协议和数据格式,其中应用最广泛的要数 HTTP 协议和 HTML(文本标记语言)了。最初,人们的网络活动并不频繁,所以一般的静态网页便可满足人们的要求了。但是,随着人们在互联网上展开的商业、娱乐等活动的增多,静态的网页已经远远不能满足人们的交互需求。因此,人们推出了 CGI(公共网关代理接口),这种应用方式使得人们同网页的交互成为可能。但是开发 CGI 程序对于程序员的要求非常高,这就大大限制了它的推广和应用,直到 ASP 的出现才改变了这种状况。

ASP 以其操作简单、灵活的特性赢得了开发人员的好评,世界各地主要的网络开发商也纷纷表示将会采用它来进行各种网络应用的开发。目前,ASP 已经成为网络应用的首选开发语言之一。由于 ASP 的开发和运行环境要求不高,个人计算机完全可以用来开发 ASP 应用程序。

1.2 ASP 简介

1.2.1 关于 ASP

ASP 提供了服务器端脚本编写环境,使用它可以创建和运行动态、交互的 Web 服务器应用程序,还可以组合 HTML 页、脚本命令和 ActiveX 组件以创建交互的 Web 页和基于 Web 的功能强大的应用程序。

面向对象

ASP 应用程序很容易开发和修改,它基本面向以下人员:

- HTML 编写人员

如果您是位 HTML 编写人员,将发现 ASP 脚本提供了十分简便的创建交互页的方法。如果您想从 HTML 表格中收集数据,用顾客的姓名个人化 HTML 文件或根据浏览器的不同使用不同的特性,您会发现 ASP 提供了一个出色的解决方案。以前,要想从 HTML 表格中收集数据就不得不学习一门编程语言来创建一个 CGI 应用程序。现在,您只要将一些简单的指令嵌入到您的 HTML 文件中,就可以从表格中收集数据并进行分析。您不必再学习完整的编程语言或单独编译程序来创建交互页。随着不断掌握 ASP 和脚本语言的技巧,您可以创建更复杂的脚本。对于 ASP 可以使用 ActiveX 组件来便捷地执行复杂的任务,比如链接数据库以存储和检索信息。

- 有经验的脚本编写人员

如果您已经掌握了一门脚本语言,如 VBScript, JavaScript 或 Perl,并且已经了解 ASP 的使用方法。只要安装符合 ActiveX 脚本标准的相应脚本引擎,就可以在 ASP 页

中使用任何一种脚本语言。ASP 带有 Microsoft Visual Basic Scripting Edition (VBScript) 和 Microsoft JScript 的脚本引擎, 这样您就可以立即开始编辑脚本。至于 PERL, REXX 和 Python 的 ActiveX 脚本引擎可以从第三方开发人员处获得。

- Web 开发人员

如果您已经掌握了一门编程语言, 如 Visual Basic, 您将发现使用 ASP 是快速创建 Web 应用程序的一个非常灵活的方法。通过向 HTML 中添加脚本命令, 您能够创建应用程序的 HTML 接口。通过创建自己的 ActiveX 组件, 可以将应用程序中的商业逻辑封装到能够从脚本、其他组件或从其他程序中调用的模块。

使用 ASP 进行计算的 Web 页可转化为有形的利益, 它使 Web 供应商能够提供交互的商业应用, 而不仅仅是发布信息。例如, 旅行社不仅能够公布航空时刻表, 还可利用 ASP 脚本可以让客户查看现有的航班、比较费用并预定座位。

包含在 Windows NT Option Pack 中的 Microsoft Transaction Server(MTS)可以降低在服务器上构造程序的复杂性和费用。MTS 可以解决开发保密性强、可分级以及可靠性强的 Web 应用程序时所遇到的复杂问题。

ASP 模型

浏览器从 Web 服务器上请求 ASP 文件时, ASP 脚本开始运行; 然后 Web 服务器调用 ASP, ASP 全面读取请求的文件, 执行所有脚本命令, 并将 Web 页传送给浏览器。

由于脚本是在服务器上运行而不是在客户端, 传送到浏览器上的 Web 页是在 Web 服务器上生成的。所以不必担心浏览器能否处理脚本, Web 服务器已经完成了所有脚本的处理, 并将标准的 HTML 传输到浏览器。由于只有脚本的结果返回到浏览器, 所以服务器端脚本不易复制。用户看不到创建他们正在浏览的页的脚本命令。

1.2.2 ASP 的新功能

ASP 在以下功能方面有所增强, 更易于开发应用程序。

- Microsoft 脚本调试程序。您可以使用脚本调试程序调试 .ASP 文件。
- 事务性脚本。ASP 页和页中使用的对象可以运行于事务环境之下。在页上的全部工作只有两种结果: 提交或终止。事务可以保护应用程序免于因失败而导致数据丢失。
- 管理工具。许多应用程序设置可在 Internet 服务管理器上进行, 这是因为有了 Internet Information Server(IIS)和 Personal Web Server(PWS, 在 Windows NT Workstation 上)这样的管理工具。
- 隔离 ASP 应用程序。您可以隔离 ASP 应用程序以便它们能在 Web 服务器上独立的内存空间内运行。万一某个应用程序失败了, 可以保护其他应用程序和 Web 服务器不受影响。它还可以在不停止 Web 服务器的情况下, 停止一个应用程序或从内存中卸载其组件。
- 文件上传。浏览器可以通过 Posting Acceptor 应用程序将文件上传到 Web 服务器。可以编写 ASP 脚本来发送带有从 Posting Acceptor 传来的信息的 E-mail, 如每一个上传文件的位置和名称。请参阅 Posting Acceptor 文档(在 Microsoft Site Server

Express 的目录表中)。

- 新的 ASP 组件。ASP 提供了新的基础组件。
- 应用程序根目录的改变。ASP 应用程序的根目录在您的 Web 站点上可以是物理目录。以前根目录必须映射为虚拟目录现在可以用 Internet 服务管理器指示 ASP 应用程序的根目录。必须将 .ASP 文件置于应用程序树中, Web 服务器才能检测文件的改变并且自动重装它们。
- 支持 Internet Explorer 4.0 通道和 Web 预测。当 ASP 要发送包含通道定义命令的文件时, 它会自动告诉 Internet Explorer 4.0。使用 ASP 很容易就能生成通道定义 (.cdf) 文件。
- 支持 HTTP 1.1 协议。IIS 和 PWS 支持 HTTP 1.1 协议。在响应 HTTP 1.1 的浏览器(例如 Internet Explorer 4.0)时, 这一支持可使 ASP 利用新协议的高效率。例如, 当向浏览器返回页的时候, ASP 使用大块转换编码以提高网络效率。

1.2.3 创建 ASP 页

ASP 文件是以 .ASP 为扩展名的文本文件, 这个文本文件可以包括文本、HTML 标记和 ASP 脚本命令的任意组合。

创建 .ASP 文件非常容易。如果要在 HTML 文件中添加脚本, 只需将该文件的扩展名 .htm 或 .html 替换为 .ASP 就可以了。要使 .ASP 文件可用于 Web 用户, 请将这个新文件保存在 Web 站点上的目录中(请确定启用了该目录的脚本和执行权限)。使用浏览器查看该文件时, 可以看到 ASP 处理并返回了 HTML 页。现在可以在 .ASP 文件中添加脚本命令了。

【注意】 因为 .ASP 文件需要额外的处理, 请不要将所有的 HTML 页都转换为 .ASP 页, 应该只将包含脚本命令的文件转换为 .ASP 文件。 .ASP 和 .htm 文件可以放置在相同的目录中。

使用任何的文本编辑器都可以创建 .ASP 文件。使用那些带有 ASP 增强支持的编辑器将更能提高效率, 如 Microsoft Visual InterDev 4.0。如果您从未使用过 HTML, 请考虑先使用 Microsoft FrontPage。使用 FrontPage 创建文档和格式化文本就像使用文字处理工具一样简单。接下来还可以使用 Insert Script 命令在 FrontPage 创建的 HTML 页中加入简单的 ASP 命令。

加入 Script 命令

脚本是一系列的命令和指令。与 HTML 标签不同, Script 命令指示 Web 服务器执行操作, 而 HTML 标签只是简单地格式化文本或读取图形、视频及音频文件。Script 命令可以在变量中存储用户名, 在返回到浏览器的页中显示用户名或将用户名存储在数据库中。

脚本命令通过定界符与文本进行区别。定界符是一个字符或一串字符, 它标志着单元开始或结束。HTML 使用定界符——小于号(<)和大于号(>)括入 HTML 标签。

ASP 使用定界符 <% 和 %> 括入脚本命令。您可以在定界符中括入任何命令, 只要这些命令对正在使用的脚本语言有效。下例展示了包含脚本命令的简单的 HTML 页:

```
<HTML>
<BODY>
This page was last refreshed on <%= Now %>.
</BODY>
</HTML>
```

VBScript 函数 Now 返回当前的日期和时间。Web 服务器处理该页时, 会使用当前的日期和时间替换 `<%= Now %>` 并且将该页返回到浏览器:

```
This page was last refreshed on 8/1/97 2:20:00 PM.
```

定界符括入的命令称为主脚本命令。这些命令用主脚本语言处理。在脚本定界符中使用的任何命令都必须对主脚本语言有效。默认情况下, 主脚本语言是 VBScript。要设置另一种语言, 就必须重新在页首声明。

如果您已经熟悉了客户端脚本编写(编写在浏览器上运行的脚本), 请注意不要将 HTML `<SCRIPT>` 元素括入表达式。在编写服务器端脚本时, 只用 `<SCRIPT>` 元素在主脚本语言外的其他语言中定义过程。

合成 HTML 和脚本命令

在 ASP 定界符中可以包含任何在主脚本语言中合法的语句、表达式、过程或操作符。在 VBScript 和其他脚本语言中, 语句是语法完整的单元, 该单元表达某项活动、声明或定义。下面的条件语句 `If...Then...Else` 是一条常见的 VBScript 语句。

```
<%
If Time >= #12:00:00 AM# And Time < #12:00:00 PM# Then
    Greeting = "Good Morning!"
Else
    Greeting = "Hello!"
End If
%>
<%= Greeting %>
```

这条语句将值 "Good Morning!" 或值 "Hello!" 存储在变量 `Greeting` 中。`<%= Greeting %>` 命令将当前的变量值发送到浏览器中。

这样, 在(Web 服务器所在时区的)正午 12 点前浏览该脚本的用户将看到:

```
Good Morning!
```

在正午 12 点或以后浏览此脚本的用户将看到:

```
Hello!
```

在语句的不同部分之间可以加入 HTML 文本。例如, 下面的脚本在 `If...Then...Else` 语句中加入了 HTML, 产生了与上面脚本同样的结果:

```
<% If Time >= #12:00:00 AM# And Time < #12:00:00 PM# Then %>
Good Morning!
<% Else %>
Hello!
<% End If %>
```

如果条件为真,也就是说,如果时间为午夜、午夜以后或者中午以前,那么 Web 服务器将跟随条件(“Good Morning”)的 HTML 发送到浏览器;否则,将跟随 Else 的 HTML(“Hello”)发送到浏览器。这种混合 HTML 和脚本命令的方式可以方便地将 If...Then...Else 语句变换为几行 HTML 文本。如果要在 Web 页上显示多处问候,前一个例子更为有用。只需要设置一次变量,就可以重复显示。

最好从脚本命令中将 HTML 文本返回到浏览器,而不用脚本命令散布 HTML 文本。要将文本返回到浏览器,请使用 ASP 的内建对象 Response。下面的例子可以产生与前一个脚本同样的结果:

```
<%  
If Time >= #12:00:00 AM# And Time < #12:00:00 PM# Then  
    Response.Write "Good Morning!"  
Else  
    Response.Write "Hello!"  
End If  
%>
```

Response.Write 将跟随的文本发送到浏览器。要动态构造返回浏览器的文本,请在语句中使用 Response.Write。例如,可能要构造包含一些变量值的字符串。现在,先了解将脚本命令插入到 HTML 页中的几种方法。

在 ASP 定界符中可以包含用默认主脚本语言编写的过程。

如果正在使用 JScript 命令,可以插入花括号指示语句块直接进入 ASP 命令,而不用考虑它是否夹杂着 HTML 标签和文本。例如:

```
<% if (screenresolution == "low") { %>  
This is the text version of a page.  
<% } else { %>  
This is the multimedia version of a page.  
<% } %>
```

使用 ASP 指令

ASP 提供了您使用的脚本语言以外的指令。这是一些输出指令和处理指令。

ASP 的输出指令 `<%= expression %>` 显示表达式的值。这个输出指令等同于使用 Response.Write 显示信息。例如,输出表达式 `<%= sport %>` 将文字 climbing(变量当前的值)传送到浏览器。

ASP 处理指令 `<%@ keyword %>` 为 ASP 提供处理 .ASP 文件所需的信息。例如,下面的指令将 VBScript 设为该页的主脚本语言:

```
<%@ LANGUAGE=VBScript %>
```

处理指令必须出现在 .ASP 文件的第一行。不要将处理指令放在包含 #include 语句的文件中。必须在 at 标记(@)和关键字之间加入一个空格。处理指令有下列关键字:

- LANGUAGE 关键字设置页的脚本语言。
- CODEPAGE 关键字设置页的代码页(字符编码)。
- LCID 关键字设置页的现场标识符。

- TRANSACTION 关键字指定将在事务处理环境下运行的页。
 - ENABLESESSIONSTATE 关键字指定 ASP 页是否使用会话状态。
- 可以在单个指令中包含多个关键字；关键字和值对之间必须由空格分开。不要在等号(=)左右加入空格。下面的例子中设置了脚本语言和代码页：

```
<%@ LANGUAGE=JScript CODEPAGE=932 %>
```

在脚本中的空白字符

如果您的主脚本语言是 VBScript 或者 JScript, ASP 从指令中删除空白字符。使用其他脚本语言时, ASP 保留空白字符, 这样那些需要使用位置和缩进的语言, 如 Python 就可以被正确解释了。空白字符包括空格、制表符、回车和换行符。

使用 VBScript 和 JScript 时, 在打开定界符之后和关闭定界符之前使用空白字符更容易阅读命令。下面的命令都是合法的:

```
<% Color = "Green" %>  
<%Color="Green"%>  
<%  
Color = "Green"  
%>
```

ASP 将一个语句的关闭定界符与下一个语句的打开定界符之间的空白字符删除。要保留两个语句间的空白字符, 比如在句子中显示变量值的时候, 请使用 HTML 不间断空符(sp)。例如:

```
<%  
'Define two variables with string values.  
Hello = "Hello"  
World = "World"  
%>  
<P>My response is to say "<%= Hello %>&nbsp;sp;<%= World %>." </P>
```

1.2.4 使用脚本语言

脚本语言是介于 HTML 和 Java, C++ 和 Visual Basic 之类的编程语言之间的语言。HTML 通常用于格式化文本和链接网页。编程语言通常用于向计算机发送一系列复杂指令。脚本语言也可用来向计算机发送指令, 但它们的语法和规则没有可编译的编程语言那样严格和复杂。脚本语言主要用于格式化文本和使用以编程语言编写的已编译好的组件。

ASP 使 Web 开发商用多种脚本语言编写完整的过程成为可能。事实上, 在单个 .ASP 文件内部就可以使用多种脚本语言。另外, 由于脚本是在服务器端读取和处理的, 所以请求 .ASP 文件的客户端浏览器并不需要支持脚本。

您可以使用任何一种其相应的脚本引擎已安装在 Web 服务器上的脚本语言。脚本引擎是处理用某种语言书写的命令的程序。ASP 带有两个脚本引擎: Microsoft VBScript 和 Microsoft JScript。也可以安装和使用其他的脚本语言引擎, 例如 REXX 和 Perl。

如果您已经是 Visual Basic 程序员, 就可以马上使用 VBScript, 它是 Visual Basic 的

子集。如果您是 Java, C 或 C++ 程序员, 将会发现 JScript 语法对您来说是熟悉的, 虽然 JScript 与 Java 或 C 并没有联系。

如果您熟悉其他脚本语言, 例如 REXX 或 Perl, 您可获取和安装相应的脚本引擎, 这样就可以使用您熟悉的脚本语言了。ASP 是 ActiveX 脚本的宿主。要使用一种语言, 必须安装脚本引擎, 该引擎必须遵循 ActiveX 脚本标准并作为一个 COM (组件对象模型) 对象驻留在 Web 服务器上。

设置主脚本语言

ASP 主脚本语言是用来处理在分界符 `<%` 和 `%>` 内部的命令的语言。默认时, 主脚本语言是 VBScript。您可以将任何一种具有脚本引擎的脚本语言作为主脚本语言。您可以逐页设置主脚本语言, 也可以在 ASP 应用程序中设置所有页的主脚本语言。

为某页设置语言

要设置单个页的主脚本语言, 可将 `<%@ LANGUAGE %>` 指令添加到 .ASP 文件的开头。该指令的语法是:

```
<%@ LANGUAGE=ScriptingLanguage %>
```

其中 ScriptingLanguage 就是您所设置的页的主脚本语言。如果对某页进行了设置, 那么该页将忽略在应用程序中对所有页的全局设置。

请遵循使用 ASP 指令的指导。

【注意】 要将不支持 Object.Method 语法的语言作为主脚本语言使用, 必须先创建 LanguageEngines 注册表键。

为应用程序设置语言

在应用程序中, 如要为所有页设置主脚本语言, 可在 Internet 服务管理器的“App 选项”选项卡上设置“默认的 ASP 语言”属性。

在服务器上使用 VBScript 和 JScript

在服务端通过 ASP 使用 VBScript 时, 两个 VBScript 特征将失效。由于 ASP 脚本是在服务器端执行的, 表示用户接口的元素的 VBScript 语句 InputBox 和 MsgBox 将不被支持。另外, 在服务器端的脚本中, 请不要使用 VBScript 函数 CreateObject 和 GetObject。而要使用 Server.CreateObject, 这样 ASP 就可以跟踪对象实例了。用 CreateObject 或 GetObject 创建的对象不能访问 ASP 内建对象, 也不能参与事务处理。该规则的一个例外是如果使用的是 Admin 对象和 Java monikers。

包含注释

由于 ASP 脚本是在服务器端处理的, 所以即使客户端的浏览器不支持脚本语言, 也不必通过包含进 HTML 注释标记来隐藏脚本, 但客户端的脚本则通常需要这样处理。在内容送到浏览器之前, 所有 ASP 命令都已被处理好了。您可用 HTML 注释将注释加进 HTML 页。注释将返回给浏览器, 若用户浏览 HTML 源文件, 就可以看见。

VBScript 注释

VBScript 支持撇号型注释。与 HTML 注释不同, 当处理脚本时, 它们将被删除, 而不是被送到浏览器。

```
<%
```



```
'This line and the following two are comments.  
'The PrintTable function prints all  
'the elements in an array.  
Call PrintTable(myarray())  
%>
```

输出表达式不能包括注释。例如，下面的第一行将正常运行，而第二行则不能，因为它是以 `<%=` 开头的。

```
<% i = i + 1 'this increments i. This script will work. %>  
<%= name 'this prints the variable name. This script will fail. %>
```

JScript 注释

JScript 支持 `//` 注释符。每一个注释行中都必须使用该注释符。

```
<% Call PrintDate %>  
  
<SCRIPT LANGUAGE=JScript RUNAT=Server>  
  
// This is a definition for the procedure PrintDate.  
function PrintDate()  
{  
    var x  
    x = new Date()  
    // This line sends the current date to the browser,  
    // translated to a string.  
    Response.Write(x.toString())  
}  
  
</SCRIPT>
```

大小写敏感性

VBScript 不区分大小写。例如，您可以用 `Request` 或 `request` 来引用 ASP `Request` 对象。不区分大小写的后果是您不能用大小写来区分变量名。例如，您不能创建两个名为 `Color` 和 `color` 的单独变量。

JScript 区分大小写。要在脚本中使用 JScript 关键字，就必须按参考页中所示的大小写来书写。例如，用 `date` 来代替 `Date` 将导致错误。在 JScript 中，对象名必须大写；方法名和属性名可大写也可小写。本文所示的 ASP 内建对象的情况将在 JScript 命令中正常工作。

1.3 Web 服务器基础

从上面的 ASP 简介中可以看到，Web 服务器起着核心的作用。服务器主要通过运行网络操作系统、网络应用程序并通过网络接口来实现其功能，而实现 Web 的交互，需要在操作系统之上运行 Web 站点服务程序，Apache, WebSite, IIS 等就是这样的服务程序。不同的操作系统要求与相应的 Web 服务器端程序的配合使用，由于已经确定是在 ASP 基础上开发这个网站，所以要求配置能运行 ASP 程序的环境。