

旗標



旗标系列图书

Active · Server · Pages &

数据库



■ 王国荣 著
朱琳杰
王 伟 改编



ASP(Active Server Pages)

制作网页的强力工具



ADO(ActiveX Data Objects)

存取Web数据库不可缺少的对象



RDS(Remote Data Service)

真正跨越Internet的数据存取服务



案例研究:

访问者计数器、留言簿、
新闻讨论组、电子购物、
搜索引擎、密码验证...



人民邮电出版社

旗标出版股份有限公司

旗 标 系 列 图 书

Active Server Pages & Web 数据库

王国荣 著

朱琳杰 王伟 改编

人 民 邮 电 出 版 社

内 容 提 要

本书全面介绍了 Active Server Pages 网页设计技术,特别是与 Web 数据库的连接与现实。主要包括:ASP 与 Web 数据库的基础;ASP 制作网页技术;ADO、RDO 等数据访问对象在 ASP 中的灵活运用;此外,本书特别介绍了若干极其具有参考价值的案例,如:访问者计数器、留言簿、新闻讨论组、电子购物、搜索引擎、密码验证等。

本书适用于所有想开发高级 Web 功能的 Internet 开发人员使用,也可作为大专院校师生的重要参考书。

旗标系列图书

Active Server Pages & Web 数据库

- ◆ 著 王国荣
 - 改 编 朱琳杰 王 伟
 - 责任编辑 李振广
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
 北京顺义向阳胶印厂印刷
 新华书店总店北京发行所经销
 - ◆ 开本:720×980 1/16
 印张:32.5
 字数:512 千字
 印数:21 001~29 000 册
 - 1999 年 3 月第 1 版
 2000 年 4 月北京第 5 次印刷
 - 著作权合同登记 图字:01-98-3015 号
 - ISBN 7-115-07715-0/TP·1062
-

定价:59.00 元

目 录

第 1 章 Web 数据库相关技术漫谈	1
1-1 网际网络相关技术漫谈.....	2
1-2 数据库相关技术漫谈.....	8
1-3 访问 Web 数据库的方法.....	10
第 2 章 建立 ASP 与 Web 数据库的运行平台.....	15
2-1 软硬件需求.....	16
2-2 IP 地址的设置.....	18
2-3 单机运行平台的建立.....	21
2-4 安装 IIS/PWS	25
2-5 网络运行平台的建立.....	29
第 3 章 从 HTML 到 ASP	33
3-1 网页是如何显示在浏览器上的?	34
3-2 设置 IIS/PWS 目录的属性	38
3-3 .asp 到底是怎样的文件?	42
3-4 制作 ASP 网页必备的知识	44
第 4 章 ASP 对象—制作网页的强力工具(一).....	51
4-1 Response 对象	52
4-2 Request 对象.....	62
4-3 Server 对象	65
4-4 Cookies 集合对象.....	70
第 5 章 ASP 对象—制作网页的强力工具(二).....	83
5-1 生命周期与有效范围.....	84
5-2 Application 对象.....	88
5-3 Session 对象	99
第 6 章 HTML 输入表单的制作	107
6-1 制作输入表单的基础.....	108

6-2	文本框.....	114
6-3	下拉菜单.....	119
6-4	复选框与单选按钮.....	125
6-5	其他类型的表关键字段.....	131
6-6	表单与 ASP 程序的组合	134
第 7 章	案例研究—访客留言板	139
7-1	访问者留言簿的组成.....	140
7-2	利用文件储存访问者留言.....	141
7-3	访问者留言簿的完整制作过程	151
第 8 章	Web 数据库的初步体验	157
8-1	以实例认识 Web 数据库的连接	158
8-2	使用 Web 数据库的基本观念	161
8-3	存取 Web 数据库的基础.....	164
第 9 章	Recordset 的浏览、排序、及查询	177
9-1	筛选及排序记录的 Select 指令.....	178
9-2	分段浏览数据记录的技巧.....	186
9-3	筛选数据记录的 Where 条件式.....	205
第 10 章	数据操作—使用 SQL 命令	213
10-1	用来操作数据的 SQL 命令	214
10-2	数据库版本的访问者留言簿.....	220
第 11 章	数据操作—使用 Recordset 对象	237
11-1	可操作数据的 Recordset 对象.....	238
11-2	SQL 指令与 Recordset 对象的比较	243
11-3	数据锁定	247
第 12 章	SQL Server 与 DSN 的设置	251
12-1	SQL Server 的 ODBC 参数	252
12-2	DSN 的意义及设置.....	255
12-3	使用其他格式的数据库.....	261

第 13 章 Command 对象.....	269
13-1 从 SQL 命令到 Command 对象	270
13-2 带有参数的 SQL 命令	276
13-3 Command 对象在 SQL Server 中的使用	282
第 14 章 案例研究—新闻讨论群组、电子购物、聊天室	289
14-1 新闻讨论组(News Group).....	290
14-2 电子购物.....	301
14-3 聊天室.....	307
第 15 章 搜索引擎.....	311
15-1 搜索文件的基础.....	312
15-2 递归搜索所有文件夹的技巧.....	324
15-3 搜索引擎的制作.....	327
第 16 章 密码验证	337
16-1 IIS 的目录安全设置.....	338
16-2 利用程序验证密码.....	341
第 17 章 初探 RDS(Remote Data Service).....	353
17-1 使用 RDS 之前的准备工作.....	354
17-2 第一个 RDS 程序.....	359
17-3 使用 RDS	368
第 18 章 RDS 与数据存取	379
18-1 那些对象库含有 Recordset 对象.....	380
18-2 数据记录的浏览及修改.....	383
18-3 存取 Web 数据库范例程序	386
第 19 章 RDS 与 Grid 组件的结合.....	397
19-1 RDS 与 DBGrid 的结合	398
19-2 RDS 与 SSDBGrid 的结合	402
19-3 VB、RDS 与 Grid 组件的结合	412
第 20 章 RDS 与 HTML 元件的结合	415

20-1 Table 与 RDS 的结合	416
20-2 其他 HTML 组件与 RDS 的结合	425
附录 A HTML 的基本概念及使用诀窍.....	435
附录 B 新概念的 VB Script 教材	443
附录 C VB Script 常用函数参考手册.....	501

1

Web 数据库 相关技术漫谈

踏入计算机业十几个年头，经常感叹技术发展太快，今天正流行的技术转眼就变成昨日黄花，作为一个技术人员，为了追赶新技术，有如“夸父追日”。

不像其他的技术会随着时间的推移而变得越来越没有声息，数据库反倒是一门历久弥坚的学科，考察其原因，则是因为技术爆炸对数据库的需求只增不减。

至于 Web 数据库，则是这几年新兴起来的技术，在因特网热潮不减的情况下，如何让原本在单机或局域网中使用的数据库移植到因特网中，也成为必然的趋势。在本章，我将与您一起谈谈 Web 数据库的相关技术。

1-1 因特网(Internet)相关技术漫谈

进入 Web 数据库领域，除了数据库方面的知识很重要之外，因特网的相关背景知识也是必要的，本节就让我们先从因特网开始谈起。

TCP/IP 通讯协议

我在此不是要介绍 TCP/IP 通讯协议，也不是要谈 Winsock(在 Windows 环境下，让 TCP/IP 通讯协议得以运行的 API 或程序库)，不过我必须强调的是作为一个网页制作者，了解 TCP/IP 的基本设置是十分必要的，虽然 TCP/IP 的基本设置没有什么特别的技术，但如果设置错误，却会造成网络无法顺利连接，其重要性不亚于其他技术，我在“建立 ASP 与 Web 数据库的操作平台”一章中，将会专门说明 TCP/IP 通讯协议的相关设置。

浏览器(Browser)

浏览器没有什么好说的，如果我是用户，就会同时拥有 Internet Explorer(以下简称 IE) 及 Netscape Communicator(以下简称 NC)，而上网时再根据情况决定使用哪一种浏览器。在浏览器的大战下，用户确实是受害者，不过今天我们的角色有点不同，作为一个网页及 Web 数据库的开发者，我们不能假设每一位用户都有两套浏览器，也不能假设用户一定使用某一种浏览器，反之，我们应该尽可能地开发出可以同时适用于 IE 及 NC 的网页及 Web 数据库。

不过对于某些特殊环境的网页倒可以例外，举例来说，贵公司现在有很多分

公司分散在各地或海内外，而您负责编写全公司的数据库系统，考虑到人员分散在各地，不易训练的情况，您不得不舍弃传统的 Windows 程序界面(例如使用 VB、VC++、Delphi 等工具来开发)，而采用浏览器为操作界面，此时我觉得可以采用 IE，因为 IE 可以在网页中嵌入 ActiveX 控件及对象，这样对于系统的开发来讲确实要容易一点。

HTML 语言

HTML(hypertext Markup Language, 超文本标记语言)是构成网页的最“基本”的元素，虽然随着技术的发展，又有各种新技术产生，例如 Dynamic HTML、Java Script、VB Script、ASP 等，但它们依然是建构在 HTML 之上的，并没有舍弃现有的 HTML。

怎样将 HTML 应用于网页的制作呢？比较令人欣慰的是现在 HTML 编辑工具已经相当普遍，要建立一个 .htm 的网页(文件)，只要使用自己最习惯的 HTML 编辑工具(例如 IE 的 ProntPage Express、NC 的网页设计师、Office 97 的 Word 等) 进行编辑，即可轻松地制作出不错的网页来。

这么说，我们是不是就可以不必学习 HTML？对一般的网页制作者来说，答案是肯定的，但是对于高级网页制作者(例如阅读本书的您就属于这一族)却是否定的，原因是我们会利用 ASP 程序动态建立 HTML 格式的文件，然后传到浏览器端，所以不得不了解 HTML 语言的语法。

虽然了解 HTML 语法是阅读本书必备的基础，但我并不主张您去买几本 HTML 方面的书籍回来，为此我特别整理了一套“即学即用”及“要用再学”的方法，并且收录于本书的附录 A “HTML 的基本概念及使用窍门”中，相信对您会有所帮助。

动态网页

HTML 是一种语言，但并不是“程序”语言，HTML 所定义的范围仅局限于如何表现文字、图片，以及如何建立文件之间的链接，而程序则是经过规划的一连串命令(或称“语句”(statement))，而这样的命令可用来驱使操作系统或应用程序执行某些工作。

由于 HTML 不是程序语言，因此由 HTML 所构成的文件并不会因时因地而产生变化，所以便有“静态”(static)网页之称。为了让网页具备动态变化的能力，不管是浏览器或 Web 服务器制造厂商，都着手将程序加到纯 HTML 文件中，而发展出动态网页技术。

要产生动态网页，一定要编写程序，至于该选择哪一种程序语言，往往是令网页制作者头痛的事情。首先我必须说明的是，程序的可执行端可分成浏览器(或称 Web 客户端)及 Web 服务器端，如果程序在浏览器上执行，则服务器必须把程序代码传下来，而浏览器也要聪明到可以执行服务器所传输下来的程序，如果程序在服务器端执行，则服务器只要把执行的结果传下来即可。下面让我来说明目前两大浏览器及 Web 服务器可执行的程序。

浏览器与编程语言

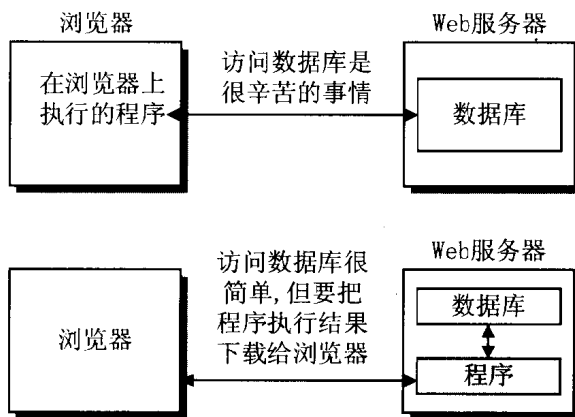
目前两大浏览器可执行的程序并不完全相同，其中 NC 可执行的程序有 JAVA Script，而 IE 除了 JAVA Script 和 JAVA 之外，还有微软极力推荐的 VB Script 和 ActiveX 组件。

程序	IE 是否可执行	NC 是否可执行
JAVA	○	○
JAVA Script	○	○
VB Script	○	×
ActiveX 组件	○	×

撇开个人的喜好，在浏览器的选择上，IE 确实比较好，不过我说过，我们

制作网页时，不能假设用户所使用的浏览器一定是 IE，因此，如果想把动态网页的程序交给浏览器来执行，又希望它们能够同时适用于 IE 及 NC，无可否认的 Java 及 Java Script 是唯一的选择，当然，也有另一派人认为将来 IE 会打败 NC，因此完全可以跟着微软走，我不是先知，更无法预知未来，这种事情还是交给广大的用户来决定。

就以上所列出的 Java Script、Java、VB Script 及 ActiveX 对象而言，Java Script 及 VB Script 的使用范围仅局限于浏览器本身，简单的说，就是它们可以通过浏览器所提供的对象来控制浏览器，进而制作出许多动态网页的效果，但它们都无法与 Web 服务器进行通讯，因此它们对于凡是与 Web 数据库有关的应用都无能为力，至于 Java 及 ActiveX 对象虽然具备与 Web 服务器通讯的能力，但相对而言程序设计要困难得多，以我的经验而论，想要利用浏览器上面执行的程序来存取 Web 数据库是一件比较辛苦的事情。



在 Web 服务器端执行的程序

相对于在浏览器上执行的程序而言，Web 服务器端的程序想要存取 Web 数据库可就轻松多了，毕竟程序与数据库是在同一台机器上(也可能不在同一台机

器上,但仍然以在同一局域网的情况居多),不过,Web 服务器端的程序必须将执行的结果传给浏览器,这算是程序的执行场所由浏览器转移到 Web 服务器所增加出来的工作。

过去在 Web 服务器端执行的程序统称为 CGI 程序,下面就让我来解释一下什么叫 CGI。

CGI(Common Gateway Interface)

对于上网族来说,CGI 好像是很技术的东西,但实际上,它的概念与 UNIX 或 DOS 的 Shell 很像,对 Shell 而言,用户是利用键盘输入指令,然后由 Shell 解释这个指令并且启动对应的程序,对 CGI 而言,指令来自于浏览器,以 CGI 接口传送到服务器,再由服务器解释后启动服务器中对应的程序。

对 Shell 指令而言,可能带有参数,当程序被 Shell 启动之后,它可以通过 Shell 所提供的接口读取指令的参数,对服务器端的程序而言,当它被启动之后,则可以通过 CGI 所提供的接口读取指令的参数,此外,读取环境变量的概念也是相同的。

最后是输出的比较,同一个程序可能被 Shell 或 CGI 所启动,被 Shell 启动的程序其输出是屏幕,被 CGI 启动的程序其输出则是网络,也就是传输给浏览器。

CGI 可以说没有任何技术可言,如果 Web 服务器提供有 CGI 接口程序(可能的形式有 source code、object code、library、dynamic linking library 等,视 Web 服务器而定),那么编写 Web 服务器端的 CGI 程序与一般的程序是没什么差别的。

CGI 的难处何在?

1. 同样的操作系统,若 Web 服务器不同,则 CGI 接口程序就不相同,过去常有读者抱怨:“Web 服务器由 WebSite 换成 IIS 之后,原来的 CGI 程序就无法使用了”,原因其实是因为不同的 Web 服务器所提供的 CGI 接口程序不同所造成的。

2. 对提供虚拟主机的 ISP 而言，通常不允许用户选择自己熟悉的语言来编写 CGI 程序，因为如果用户写的程序出了乱子，对 ISP 而言是很麻烦的事情，因此限制很多。
3. 访问数据库的程序不容易编写，大部分程序语言所定义的范围只涵盖文件的功能，至于数据库的程序库，则必须另外寻找。

ASP(Active Server Pages)

ASP 程序可以说是 CGI 程序的改进版，它跟 CGI 程序一样是在 Web 服务器端执行的，但是它有以下优点：

1. CGI 接口对象化：当我们在 CGI 程序中收集网页数据时，最麻烦的事情是解析浏览器上传来的数据(主要是为了解析特殊字符及中文码)，若使用 ASP 来收集网页数据，则 ASP 会代为解析，并且转换成对象供 Script 语言(VB Script 或 Java Script)读取。
2. 可以使用 ActiveX 对象：除了 ASP 本身所提供的对象外，凡是 Windows 环境下的 ActiveX 对象也都是 ASP 网页制作中可以引用的系统资源。而由于 ASP 程序可以使用 Windows 环境的 ActiveX 对象，因此也就能够引用 ADO(ActiveX Data Objects，存取 Web 数据库的一个十分重要的对象，稍后介绍)进而存取 Web 数据库。

想要执行 ASP，我们必须在 NT 服务器上安装 IIS(Internet Information Server)或是在 NT Workstation/Windows 95/Windows 98 上安装 PWS(Personal Web Server)，有关 IIS/PWS 的安装，请参阅第二章“建立 ASP 与 Web 数据库的运行平台”，至于如何编写 ASP 程序，则请参阅第三章“从 HTML 到 ASP”及其以后的章节。

1-2 数据库相关技术漫谈

ODBC(Open DataBase Connectivity)

如何通过网络存取服务器数据库呢？对于早期的数据库程序开发者而言，想要存取服务器数据库，唯一的途径就是使用服务器数据库厂商所提供的开发工具，这有什么缺点呢？想想看，您本来对 VB 或某一套开发工具已经很熟悉了，为了存取服务器数据库，却必须重新学习另一套开发工具，何苦呢！

后来微软发表了 ODBC，希望给前端开发工具提供一套统一的程序接口，而通过这个程序接口，便能够存取不同厂商所提供的服务器数据库(俗称“异质数据库”)，这的确是一个很好的想法，虽然一路走来，经历了 bug 缠身、系统不稳定等阶段，但到目前为止，已走向稳定成熟的阶段，而尤其重要的是，它已经成为业界存取服务器数据库的标准，这表示用户不必为了迁就服务器数据库，而改变自己习惯的开发工具。

其实 ODBC 不仅仅适用于服务器数据库，即使是单机数据库，也可以采用 ODBC 接口来存取，这表示以 ODBC 来开发单机数据库程序之后，若将来数据库扩充成为服务器数据库，则存取数据库的程序仍然可以使用。

存取数据库的对象群：DAO、RDO、及 ADO

使用 ODBC 来存取数据库固然是一种很棒的方法，但 ODBC API 并不容易使用，因此便诞生了数据库存取对象，一般来说，数据库存取对象除了会将 ODBC API 封装成为对象的一部分功能之外，也会针对某一种格式的数据库提供专用的数据库存取功能，以 DAO(Data Access Objects)为例，除了可用来存取 ODBC 数据库之外，另一项重要的功能就是存取 .mdb 格式的数据库。

DAO(Data Access Objects, 数据访问对象)

DAO 是微软公司主要应用程序及开发工具访问数据库的标准对象，截至目前为止，利用它来存取数据库的软件有 Access、Visual Basic、Visual C++、Excel、Word 等，如果能够彻底学会这一套(应该说一大套)对象，我们也可以像上述的

软件一样，完全地控制数据库。

RDO(Remote Data Objects, 远程数据访问对象)

配合 ODBC 的使用，DAO 已经具备了访问服务器数据库的能力，但是在窥视数据库大饼的情况下，微软又推出了 RDO(其实是 RDAO, Remote Data Access Objects)，主要的特点是强化 SQL 服务器(微软公司所开发的服务器等级数据库)的访问功能，并且提高执行效率。RDO 目前附属于 VB 企业版。

ADO(ActiveX Data Objects ActiveX 数据对象)

自从微软大举介入 Internet 领域后，Active 就成为微软 Internet 产品中不可缺少的名词，例如 OCX+Internet 叫做 ActiveX，Web Server Pages 叫做 Active Server Pages，自然而然地，DAO 就变成了 ActiveX DAO，取三个字母为 ADO。

数据库访问对象	主要功能简介
DAO	储存 .mdb 格式的数据库
RDO	提高 SQL 服务器的访问效率
ADO	摘取 DAO 及 RDO 的精华

ADO 有什么特点呢？千万别把它想成是全新的技术，从对象结构的角度来看，它比 DAO 提供更少的对象，从存取 SQL 服务器的角度来看，它提供的功能也不如 RDO，但是它摘取了 DAO 及 RDO 最精华的功能，成为一个小而精的更适用于 Internet 的数据库存取对象群。本书所介绍的 Web 数据库主题，其实就是以 ADO 为主轴贯穿完成的。

SQL(Structured Query Language)

在微软所推出的数据库访问对象中，又是 DAO、又是 RDO、又是 ADO，不知道未来还会出现什么 XDO，搞得我们这些可怜的工程师只要 NDO(看到 Data Objects，就喊 No!)。虽然 ADO、RDO、及 ADO 在访问数据库的概念及使用的对象方面都十分类似，但它们仍然有很多不同，学习新的对象难免要花些时间，这难免让我们考虑有没有一套技术可以通吃数据库，答案是有的，那就是 SQL。

SQL 是发送给数据库并要求数据库操作其数据的指令，它是专门为数据库设计的语言，比一般的计算机程序语言要简单得多，主要分成两部份：DDL (Data Definition Language) 及 DML (Data Manipulation Language)，其中 DDL 是建立数据表及数据列的指令群，而 DML 则是操作数据库记录的指令群，如数据库记录的添加、更新、删除、及查询等。

有一点我必须说明的是 SQL 是一种语言，不是程序，因此我们必须通过数据库软件所提供的操作接口或数据库访问对象，方可将 SQL 指令发送给数据库，进而达到访问数据库的目的，举例来说，Access 及 SQL 服务器都提供有发送 SQL 指令的操作接口，而前面所介绍的 DAO、RDO、或是 ADO，也都允许我们发送 SQL 指令。跟 ADO 一样，SQL 也是本书的主角。

1-3 访问 Web 数据库的方法

前面谈了那么多的技术名词，到底哪些东西与 Web 数据库最有关系呢？下面先听我讲解一下 Web 数据库的访问过程，便可知端倪。