



清松电脑系列丛书

VBScript 可视化程序设计

刘炳文 编著

清华大学出版社

<http://www.tup.tsinghua.edu.cn>



VBScript 可视化程序设计

刘炳文 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

VBScript 是重要的 Web 开发工具,是 Microsoft ActiveX 和 COM 技术的重要组成部分。本书以 Visual InterDev 和 ActiveX Control Pad 为基础,深入浅出地介绍了用 VBScript 开发 Web 应用程序的方法和技巧。主要内容包括:Visual InterDev 6.0 和 ActiveX Control Pad 编程环境、ActiveX 技术、HTML 语言、脚本语言 VBScript 基础知识、基本输入输出、对象以及 VBScript 的控制结构、数组、过程定义和调用、事件处理和脚本对象模型、部分内部控件、ActiveX 控件以及键盘事件过程和鼠标事件过程。本书内容通俗易懂,编排合理,概念清楚、准确,适合自学,并注意了对 Visual Basic 的继承性。

本书可供从事 Web 开发的专业人员和网络爱好者自学,还可作为计算机应用人员和大专院校师生的参考书。

版权所有,翻印必究。本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

VBScript 可视化程序设计/刘炳文编著. - 北京:清华大学出版社,1999
ISBN 7-302-02166-X

I . V … II . 刘 … III . VBScript-程序设计 IV . TP312

中国版本图书馆 CIP 数据核字(1999)第 00525 号

出 版 者:清华大学出版社(北京清华大学学研楼,邮政编码:100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑:曹美芳

印 刷 者:清华大学印刷厂

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 印张:30.5 字数:720 千字

版 次:1999 年 12 月第 1 版 1999 年 12 月第 1 次印刷

书 号:ISBN 7-302-02166-X/TP·2168

印 数:0001~5000

定 价:42.00 元

前 言

VBScript 是重要的 Web 开发工具之一,是 IIS、ASP、CGI 程序设计的基础,是 Microsoft ActiveX 和 COM 技术的重要组成部分。它利用和继承了 Microsoft 著名编程工具 Visual Basic 已有的成果,致力于开发 WWW 页面的制作市场。作为一种脚本编制语言,VBScript 已受到越来越多的页面设计人员的欢迎。

在 Web 应用程序中,VBScript 是 HTML 文档的一部分,它是依托 HTML 而存在的,也就是说,必须把脚本代码放在 HTML 文档中,否则将无法执行。VBScript 程序是纯文本代码,并且可以用简单的文本编辑器(如 NotePad)编写。早期 Web 页面的开发完全是“文本”化的,通常用 NotePad 编辑 HTML 文档程序。每个程序要经过编辑、保存、运行(在浏览器中)、修改等步骤,有时候,可能需要在不同的环境中反复多次,很不方便,而且不能进行可视化设计。

为了方便 Web 应用程序的开发,Microsoft 推出了一系列可视化开发工具,包括 Visual InterDev、ActiveX Control Pad、FrontPage 等,利用这些工具,可以实现 Web 应用程序的可视化设计。但是,不管使用哪一种工具,HTML 和脚本语言(如 VBScript、JavaScript 等)都是必要的基础工具。本书的“可视化”是通过两个开发工具,即 Visual InterDev 和 ActiveX Control Pad 来实现的。

Visual InterDev 是一个功能强大的、用于在 Internet 和 Intranet 领域建立和维护动态 Web 站点的软件,是快速设计、建立和调试 Web 应用程序的综合集成开发环境。它是 Microsoft 公司 Visual Studio 的成员之一,类似于 Visual Basic 和 Visual C++ 等传统的应用程序开发工具。Visual InterDev 是综合性的 Web 开发系统,含有完整的设计、数据库开发、站点管理和可视化的编辑工具。Visual InterDev 为用户提供了一个易于使用的、可视化的、简单完整的开发环境,可用来建立完善的 Web 应用程序,具有较高的效率。

尽管 Visual InterDev 具有较强的功能,但在程序的可视化设计方面,它并不是完美无缺的。例如,页面上的对象不能精确定位,不能像 Visual Basic 那样随心所欲地进行界面设计。而利用 Microsoft 的一个小应用程序,即 ActiveX Control Pad,就能实现 Web 页面的可视化设计,并且可以用与 Visual Basic 中在窗体上画控件类似的操作设计界面。利用 ActiveX Control Pad,结合 HTML 代码、ActiveX 控件、HTML 布局以及 VBScript 或 JavaScript,可以十分方便地建立 Web 页面。

VBScript 程序设计包括两个方面,一是编写客户端程序,另一个是编写服务器端程序。本书通过 Visual InterDev 和 ActiveX Control Pad 这两个编程工具,系统、全面、深入浅出地介绍了用 VBScript 实现客户端 Web 应用程序设计的各个方面。从内容上看,全书大体上可分为三部分。其中第 1 到 4 章为第 1 部分,主要介绍了 ActiveX 技术、HTML 以及脚本语言 VBScript 的基础知识。第 5 到 10 章为第 2 部分,在这部分中,介绍了 Visual InterDev 6.0 的编程环境,以这个环境为基础,介绍了 VBScript 的基本输入输出、对象、表单元素以及 VBScript 的

控制结构、数组、过程定义和调用、事件处理和脚本对象模型。在最后一部分(第 11 到 14 章)中,介绍了简单而实用的 Web 应用程序开发工具 ActiveX Control Pad,并结合这个工具介绍了部分内部控件、ActiveX 控件以及键盘事件过程和鼠标事件过程。

目前国内图书市场上有关 VBScript 程序设计方面的著译颇多,本书在内容、编排及叙述等方面与这些书有较大的不同,除内容通俗易懂,编排合理,适合自学和概念清楚、准确,按教材风格编写外,同时强调对 Visual Basic 的继承性。它既适用于初学者,又可供具有一定 Web 程序设计经验的读者参考。

目前国内越来越多的人转向 Web 应用程序开发,VBScript 已成为重要的基础工具,希望本书能满足广大读者用 VBScript 开发 Web 应用程序的强烈愿望,对 Internet 在我国的推广应用起到一些作用。由于笔者水平有限,编写时间紧促,不当之处在所难免,衷心希望专家和广大读者不吝赐教,批评指正。

刘炳文

1999 年 9 月于北京

目 录

第 1 章 ActiveX 技术	1
1.1 从剪贴板到 ActiveX	1
1.2 ActiveX 技术	3
1.2.1 ActiveX Internet 模型	3
1.2.2 ActiveX 模型	5
1.3 ActiveX 与其他技术	7
1.3.1 ActiveX 与 COM	7
1.3.2 ActiveX 与 Java	8
1.4 为什么要使用 ActiveX	10
第 2 章 Internet 与超文本标记语言(HTML)	12
2.1 Web 与 URL	12
2.1.1 什么是 Web	12
2.1.2 什么是 URL	13
2.2 超文本标记语言(HTML)简介	15
2.2.1 HTML 文档的基本结构	15
2.2.2 分行、居中与注释	17
2.3 标题与列表	19
2.3.1 标题	19
2.3.2 列表	20
2.4 表格与文本	23
2.4.1 表格	23
2.4.2 文本	28
2.5 输入(INPUT)标记与内部控件	30
2.5.1 输入(INPUT)标记	30
2.5.2 内部 HTML 控件	32
2.6 窗体与图像	38
2.6.1 窗体	38
2.6.2 图像	40
2.7 MARQUEE 标记	41
2.8 框架	43

2.8.1	定义框架	43
2.8.2	浮动框架	46
2.9	超链接	48
第 3 章	HTML 与脚本语言	50
3.1	脚本语言	50
3.1.1	脚本宿主与脚本引擎	50
3.1.2	VBScript 与 JavaScript	51
3.2	HTML 与 VBScript	55
3.2.1	脚本代码在 HTML 文档中的位置	55
3.2.2	模块	57
3.2.3	事件处理	57
3.3	VBScript 语言与 Visual Basic	59
3.3.1	VBScript 与 Visual Basic 的区别	59
3.3.2	可以在 VBScript 中使用的 Visual Basic 关键字	61
3.3.3	VBScript 中不能使用的 Visual Basic 的功能和关键字	66
3.4	编写简单的 VBScript 程序	68
第 4 章	VBScript 基础	71
4.1	数据类型	71
4.2	常量和变量	73
4.2.1	常量	73
4.2.2	变量	75
4.3	运算符与表达式	78
4.3.1	算术运算符	79
4.3.2	关系运算符与逻辑运算符	81
4.3.3	表达式的执行顺序	83
4.4	子类型测试与转换	84
4.4.1	测试子类型	85
4.4.2	数据类型转换	88
4.5	常用内部函数	90
4.5.1	转换函数	91
4.5.2	数学函数	93
4.5.3	字符串函数	94
4.5.4	日期和时间函数	98
4.5.5	随机数函数	102

第 5 章 Web 应用程序开发工具——Visual InterDev 6.0	104
5.1 Visual InterDev 6.0 简介	104
5.1.1 Visual InterDev 6.0 的主要功能	104
5.1.2 Visual InterDev 6.0 的主要特点	106
5.2 Visual InterDev 6.0 编程环境	108
5.2.1 Visual InterDev 6.0 的用户界面	108
5.2.2 工具栏	111
5.3 建立 Web 工程	116
5.3.1 几个基本概念	116
5.3.2 用 Web 工程向导建立一个新的工程	117
5.3.3 改变现有的设置	123
5.4 窗口	125
5.4.1 工程浏览器窗口	125
5.4.2 属性窗口和工具箱窗口	131
5.5 HTML 编辑器	135
5.5.1 视图、工具栏和文档大纲	135
5.5.2 页面属性设置	140
5.6 文本与列表	142
5.6.1 文本的输入与格式化	143
5.6.2 列表	145
5.6.3 文本滚动	147
5.7 超链接与书签	148
5.7.1 建立超链接	149
5.7.2 建立书签	150
5.8 图像	151
5.8.1 插入图像	151
5.8.2 设置图像属性	152
5.9 HTML 表格	153
5.9.1 Visual InterDev 中的表格	154
5.9.2 表格的建立和基本操作	155
5.9.3 表格属性设置	157
第 6 章 VBScript 的基本输入输出	161
6.1 语句	161
6.1.1 赋值语句	161
6.1.2 注释语句	162
6.2 信息框	163
6.2.1 简单信息框	163

6.2.2	MsgBox 函数	164
6.2.3	MsgBox 语句	168
6.3	InputBox 函数	169
第 7 章	VBScript 对象与表单元控件	173
7.1	对象	173
7.1.1	VBScript 的对象	173
7.1.2	属性设置	175
7.2	控件的画法和基本操作	177
7.2.1	控件的画法	177
7.2.2	控件的基本操作	178
7.3	在页面中建立表单	180
7.4	表单元控件简介	182
7.4.1	文本框	183
7.4.2	按钮	184
7.4.3	复选框和单选按钮	185
7.4.4	下拉菜单和列表框	186
第 8 章	VBScript 的控制结构和数组	188
8.1	选择结构程序设计	188
8.1.1	单行结构条件语句	189
8.1.2	块结构条件语句	189
8.2	多分支结构程序设计	193
8.3	For 循环结构程序设计	198
8.4	While 循环控制结构	202
8.5	Do 循环控制结构	205
8.6	多重循环	210
8.7	数组	214
8.7.1	数组的基本概念和定义	214
8.7.2	静态数组与动态数组	217
8.7.3	数组的清除和重定义	219
8.7.4	数组的基本操作	220
8.7.5	For Each...Next 语句	224
8.7.6	数组的初始化	229
第 9 章	过程	231
9.1	过程的定义	231
9.1.1	Sub 过程的定义	231

9.1.2	Function 过程的定义	233
9.2	过程的调用	234
9.2.1	调用 Function 过程	234
9.2.2	调用 Sub 过程	237
9.3	参数传送	239
9.3.1	形参与实参	239
9.3.2	引用	240
9.3.3	传值	242
9.3.4	数组参数的传送	243
9.4	递归	244
9.5	程序举例	247
第 10 章	VBScript 事件处理与脚本对象模型	258
10.1	VBScript 与浏览器事件	258
10.1.1	事件驱动	258
10.1.2	事件处理过程的调用方式	259
10.2	常用事件处理过程	262
10.2.1	Load 事件和 UnLoad 事件	262
10.2.2	与鼠标有关的事件	263
10.2.3	与焦点有关的事件	267
10.2.4	Submit 事件	270
10.3	脚本对象模型	275
10.3.1	脚本对象的属性	275
10.3.2	脚本对象的方法	277
10.4	Window 对象	279
10.4.1	Window 对象的主要属性	279
10.4.2	Window 对象的方法	280
10.5	Document 对象	285
10.5.1	Document 对象的属性	285
10.5.2	Document 对象的方法	289
10.5.3	文档的颜色设置	290
10.5.4	用 Document 实现块输出	291
10.6	Document 对象举例	293
10.7	其他对象	300
10.7.1	History 对象	301
10.7.2	Navigator 对象	303
10.7.3	Location 对象和 Link 对象	305
10.8	菜单	306

第 11 章 ActiveX Control Pad	311
11.1 ActiveX Control Pad 的构成和安装	311
11.2 文本编辑器	312
11.2.1 文本编辑器的基本功能	312
11.2.2 文本编辑器菜单命令	313
11.3 对象编辑器	318
11.4 布局编辑器	322
11.4.1 HTML 布局编辑器的构成	323
11.4.2 控件的基本操作	326
11.4.3 HTML 布局菜单命令	331
11.5 .alx 文件与脚本向导	333
11.5.1 .alx 文件	333
11.5.2 脚本向导	338
11.6 建立 .alx 文件	342
11.6.1 设计界面	342
11.6.2 编写代码	344
11.7 建立 HTML 文档	346
11.8 建立图像地图	350
11.8.1 建立界面并确定图像内各区域的位置	350
11.8.2 编写代码	353
第 12 章 常用内部控件	361
12.1 标签	361
12.1.1 标签的属性和事件	361
12.1.2 程序举例	365
12.2 文本框	371
12.2.1 文本框的属性、事件和方法	371
12.2.2 文本框的应用	374
12.3 命令按钮、复选框和单选按钮	378
12.3.1 属性和事件	378
12.3.2 命令按钮、复选框和单选按钮的应用	379
12.4 列表框	382
12.4.1 列表框的属性、事件和方法	383
12.4.2 列表框应用举例	386
12.5 组合框	388
12.6 图像框	392
12.7 滚动条	394
· VII ·	

12.8	旋钮控件	400
第 13 章	ActiveX 控件	403
13.1	对象标记(< OBJECT >)	403
13.2	在 HTML 中使用 ActiveX 控件	405
13.2.1	Label 控件	405
13.2.2	Marquee 控件	407
13.2.3	菜单	410
13.3	弹出菜单控件	414
13.3.1	弹出菜单控件的属性、方法和事件	415
13.3.2	弹出菜单控件举例	417
13.4	计时器(Timer)控件	421
13.4.1	用计时器控件显示数字时钟	422
13.4.2	用计时器控件处理动画	425
13.5	TabStrip 控件	429
13.5.1	TabStrip 控件的属性和方法	429
13.5.2	TabStrip 控件举例	433
13.6	Active Movie 控件	444
第 14 章	键盘和鼠标事件过程	451
14.1	KeyPress 事件	451
14.2	KeyDown 和 KeyUp 事件	453
14.3	鼠标事件	463
14.3.1	鼠标位置	464
14.3.2	鼠标键	466
14.3.3	转换参数(Shift)	469
14.4	鼠标光标的形状	471
14.4.1	MousePointer 属性	471
14.4.2	MousePointer 属性设置	472

第 1 章 ActiveX 技术

ActiveX 是近年来新升起的一颗“网络新星”，它是 Microsoft 的 Internet 长远战略的基石之一。Microsoft 试图通过 ActiveX 把 Internet 网或 World Wide Web 网与桌面(包括 Windows 桌面和 Machintosh 桌面)乃至 UNIX 合并到一起,组成一个“无缝”的集成化环境,从而使 Internet 可以像普通的硬件或局域网一样,成为从桌面访问的一个资源。

随着 Web 的发展和开发,ActiveX 技术也在不断更新和进步。在这一章中,对 ActiveX 及其相关技术作一个简单介绍,使大家对 ActiveX 有一个总体印象。

1.1 从剪贴板到 ActiveX

不少人对 ActiveX 还不很了解,但对剪贴板想必非常熟悉,而且经常使用。实际上,ActiveX 技术就是从剪贴板发展而来的。这一节将介绍 ActiveX 的产生过程。

在实际应用中,常常需要在两个应用程序之间交换信息。DOS 环境下的信息交换比较繁琐,往往要通过中间文件来实现。而在 Windows 中,应用程序之间的信息交换大致经历了以下几个阶段。

1. 剪贴板

Windows 环境下信息交换最原始的方式是通过剪贴板(Clipboard)来实现的。剪贴板是一个临时的存储区,为了把文本或图形从一个应用程序(源)拷贝到另一个应用程序(目标),可以先把要拷贝的内容放到剪贴板中,然后再“粘贴”(Paste)到目标处,如图 1.1 所示。

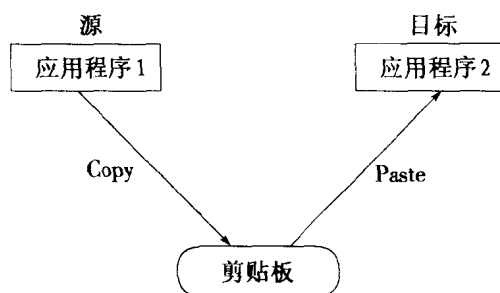


图 1.1 通过剪贴板在两个应用程序之间传送数据

例如,为了把 NotePad 中指定的文本传送到 Word 中,可以先在 NotePad 中选择要传送的文本,并执行“编辑”菜单中的“复制”命令,把这部分文本放到剪贴板上,然后打开 Word,执行“编辑”菜单中的“粘贴”命令,即可完成。

2. 动态数据交换(DDE)

用剪贴板传送数据的操作比较简单,但需要对数据进行手工复制(或剪切)和粘贴。为了能直接把数据从一个应用程序传送到另一个应用程序,Microsoft 公司又引入了动态数据交换(DDE)协议。利用 DDE,可以通过程序代码使两个应用程序之间进行通信。在使用 DDE 的两个应用程序中,其中一个为信息接收方,称为客户端;另一个为信息提供方,称为服务器端,如图 1.2 所示。

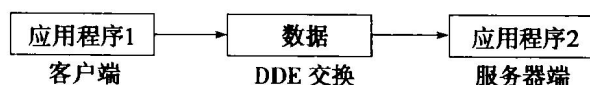


图 1.2 通过 DDE 进行通信

3. 对象链接与嵌入(OLE)

DDE 在两个应用程序之间提供了强有力的通信协议,但是,编写和调试基于 DDE 的程序是一件相当困难的事情。为此,Microsoft 公司又开发出了对象链接与嵌入(OLE)技术,它通过一种“以文档为中心”的计算模型,使得一个应用程序可以使用其他应用程序的功能。例如,通过 OLE,可以在 Word 文档中插入一个 Excel 工作表,并可使用 Excel 的所有功能。也就是说,如果在编辑 Word 文档时需要修改工作表,则可以通过双击工作表启动 Excel,对该表进行修改,完成修改后退出 Excel,并返回到 Word。

一般来说,当在文档中嵌入了一个诸如 Excel 这样的对象时,如果希望能对该对象进行修改,则要求系统中有相应的软件。而在今后,文档中的对象将带有自己的功能,无需接收方的系统中有相应的软件。

在使用 OLE 时,既可以在文档中嵌入一个对象,也可以在其中链接一个对象。当嵌入一个对象时,将在文档中存放一个此对象的拷贝,如果以后在文档外对此对象进行了修改,则文档中的对象将不会受到影响;而如果在文档中链接了一个对象,则只在文档中存放一个对象文件的索引,每次打开文档时,都要把该对象相应的文件装入文档中,因此,如果修改了对象,则可在文档中保持该对象的最新内容。OLE 容器(客户端)是一个嵌入或链接了对象的应用程序,而 OLE 服务器则是建立这个被链接或被嵌入对象的应用程序。例如,当把 Excel 工作表嵌入或链接到 Word 文档后,Word 是客户端,Excel 为服务器端,如图 1.3 所示。

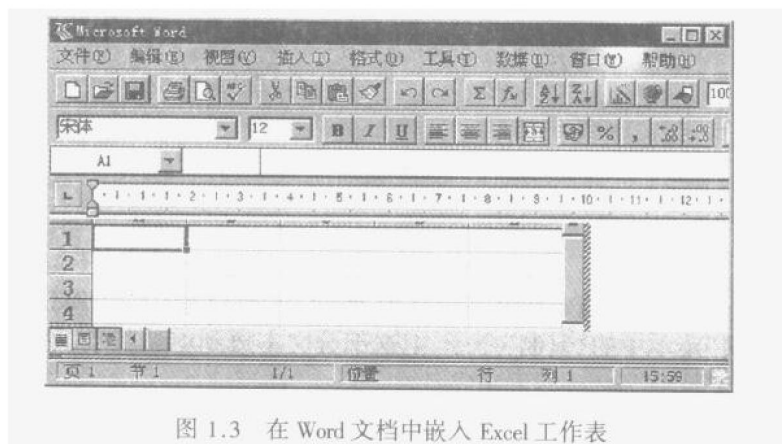


图 1.3 在 Word 文档中嵌入 Excel 工作表

4. ActiveX

随着 Internet 的迅速发展与普及,人们把注意力转向了网络,需要开发出一种新的对象技术。在这种情况下,作为一种为网络定义的 OLE,Microsoft ActiveX 应运而生。目前还无法给 ActiveX 下一个完整而明确的定义,只能把它看作是一种用于 Internet 的 OLE 技术。

1.2 ActiveX 技术

如前所述,ActiveX 是一种基于 Internet 的技术。有了这种技术,就可以实现用户计算机和信息之间的无缝链接,而不必关心这些信息存放在什么地方。在这一节中,将介绍 ActiveX Internet 模型和 ActiveX 的构成。

1.2.1 ActiveX Internet 模型

早期的计算模型是基于主机的,用户通过终端来运行存放在主机中的程序,随着工作站和计算机网络的应用,出现了客户/服务器计算模型。在这种计算模型中,用户运行存放在自己 PC 机中的应用程序,并通过网络向服务器发送请求信息,在客户端主要处理用户界面及用户输入,而服务器端则处理客户端的请求,客户端与服务器端通过网络进行通信,如图 1.4 所示。

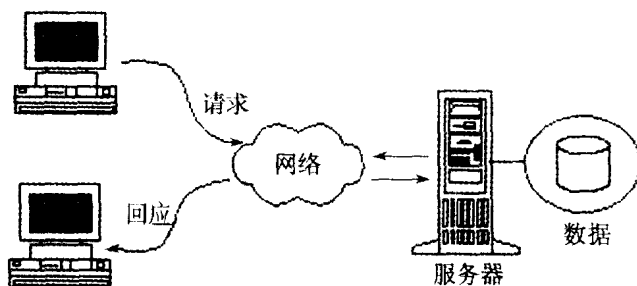


图 1.4 传统的客户/服务器结构

随着 Internet 的广泛应用,超文本传输协议(HTTP)模型继续发展了客户/服务器模型。在基于 HTTP 的模型中,客户端(通常为浏览器)与服务器端进行交互,在一般情况下,这种交互包括客户端向服务器请求指定的 Web 页。由于 Web 页由 HTML 文档组成,所以这种模型有时也称为 HTTP:HTML 客户/服务器模型,万维网就是基于 HTTP 的,如图 1.5 所示。

从用户的观点看,HTTP 是非动态的,它只提供很少的交互甚至根本没有交互功能。也就是说,用户只能简单地查看 Web 页中的内容,而不能通过 Web 页中的元素进行交互。这种交互能力可以通过公共网关接口(CGI)、Perl 及其他编程语言来实现。

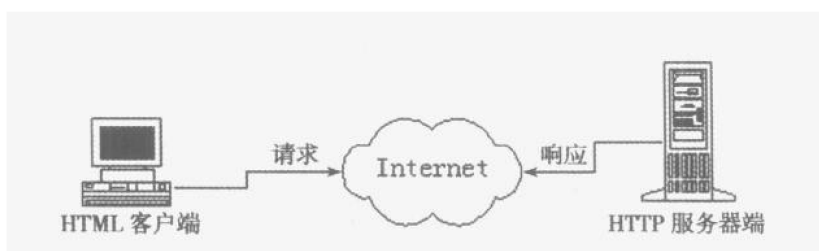


图 1.5 交互式 Web 模型

利用 HTML,可以建立一个表单,这样,用户通过浏览器就能与服务器进行交互。当用户单击浏览器中表单的 Submit 按钮时,浏览器将该表单发送给服务器,服务器运行一个程序(通常用 Perl 编写)来处理所提交的表单。根据服务器端程序的设计,可能生成一个基于 HTML 的响应,并由服务器发送回客户浏览器。

基于 CGI 的表单处理程序有一个缺点,就是每次当客户端启动一个 CGI 脚本时,服务器都要建立一个新进程。随着进程的增多,服务器的处理能力会大为下降。使用 CGI 的客户/服务器模型如图 1.6 所示。

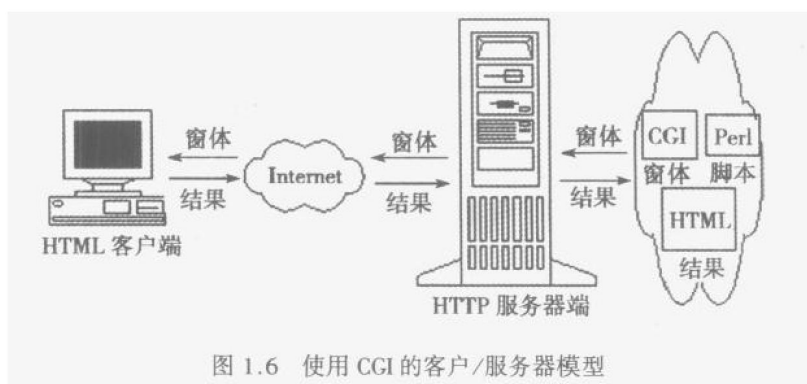


图 1.6 使用 CGI 的客户/服务器模型

为了克服上述模型的缺点,Microsoft 推出了专门为 Internet 设计的新型的计 算模型,即 Active 平台,如图 1.7 所示。Active 平台由 Active Desktop(客户端)和 Active Server(服务器端)组成,他们在功能上是对称的,这意味着在客户端和服务端都可以使用 ActiveX 控件及脚本处理,即都可以充分利用 ActiveX 控件所具有的功能。

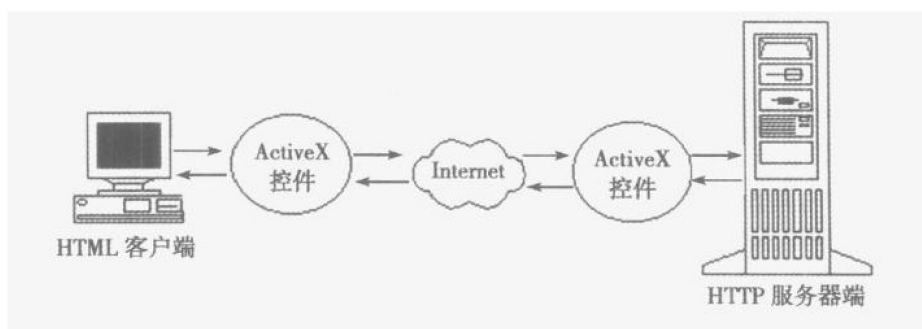


图 1.7 Microsoft ActiveX Internet 模型

1.2.2 ActiveX 模型

ActiveX 是一个“迷宫”，这个“迷宫”每天都在变，变得越来越大，越来越复杂；它由许多技术组成，而 ActiveX 是把这些技术连在一起的“粘接剂”。

ActiveX 包括以下几种技术：

- ActiveX 控件
- ActiveX 脚本处理
- ActiveX 文档
- ActiveX 服务器框架 (ActiveX Server Framework)
- ActiveX 会议 (ActiveX Conference)
- Java 虚拟机 (Java Virtual Machine)

下面对这几种技术作简单介绍。

1. ActiveX 控件

ActiveX 控件是一个提供特定功能的二进制对象，可以像一个计算机应用程序一样，把 ActiveX 控件添加到客户/服务器应用程序、Web 应用程序、Java 小应用程序以及使用 JavaScript 和 VBScript 的 Web 页中。

可以通过两种方式加强客户端的网页交互功能，即组件 (Component) 和脚本 (Scripting)。对于网页组件来说，Navigator 提供的是 applet，而 Explorer 提供的则是控件和 applet。当 HTML 网页需要用到某个组件时，Explorer 便可以加载该组件，为它安排显示区域，然后执行，这些组件就像是在浏览器里执行的小程序。

ActiveX 控件是一种开放结构式的 Internet 对象引用技术，它不是某一个特定语言的产物，但提供了非常实用的网页组件对象 (Component Object)。用不同语言编写的 ActiveX 都可以应用在 Microsoft 的 Explorer 浏览器上，每个控件就如同一个具有既定功能的 IC (集成电路)，通过不同的连接和组合，可以建立满足各种需要的应用程序。

ActiveX 控件可用于各种语言环境中，也可以用于 Web 页面中。可以通过 Visual Basic (控件建立版 CCE)、Visual C++、Borland C++ 以及 Borland Delphi 来建立 ActiveX 控件。

2. ActiveX 脚本处理

ActiveX 脚本处理是一个在诸如 VBScript、JavaScript 这样的脚本语言中配合使用了 ActiveX 控件的处理过程，通过使用脚本语言，可以使应用程序更具交互能力。例如，可以用脚本语言来处理鼠标单击以及鼠标在 Web 页上某个特定区域移动之类的鼠标事件。

不同的浏览器支持不同的脚本语言。Netscape 支持 JavaScript，而 Microsoft 的 Explorer 则支持 VBScript 和 JavaScript。不管使用哪一种脚本语言，都可以与网页中的对象以及浏览器环境彼此进行交互操作。

Explorer 提供了十分灵活的脚本语言处理机制，可以用 JavaScript 和 VBScript 来编写处理网页的脚本程序，并能直接测试。我们将在本书中介绍脚本语言 VBScript 的程序设计。