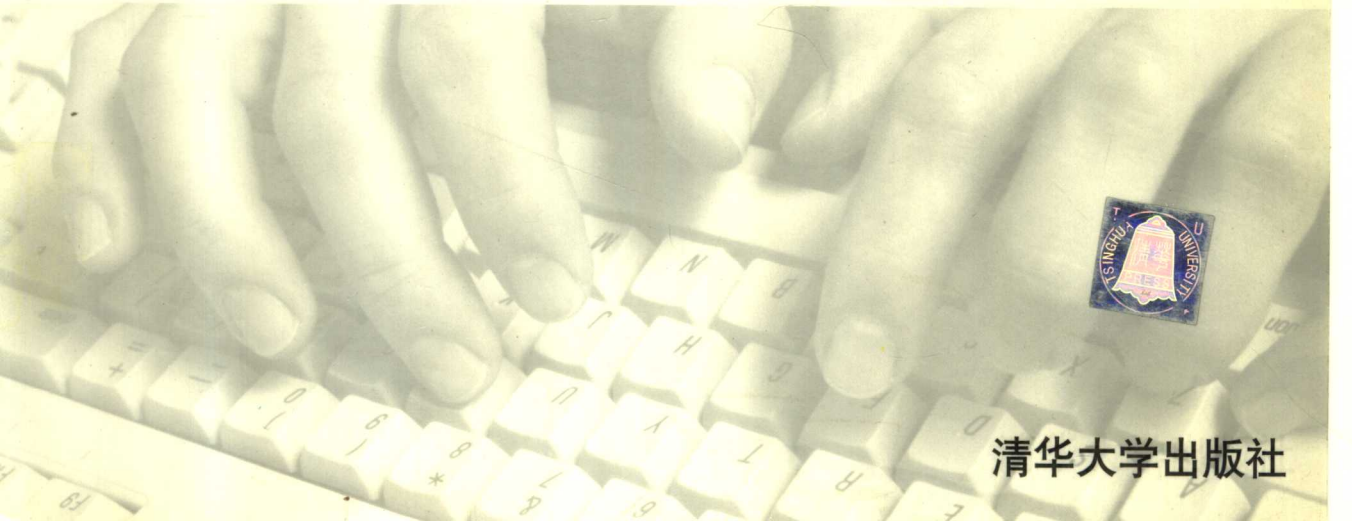




Visual Basic
高级程序设计

刘炳文 编著

Visual Basic ActiveX 程序设计



清华大学出版社

Visual Basic ActiveX 程序设计

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书较为深入地介绍了 Visual Basic 的编程技术,内容包括常用 ActiveX 控件、Windows 公共控件、对象、类和集合,以及如何建立 ActiveX DLL 和 ActiveX EXE 组件,如何建立 ActiveX 控件等;可供具有 Visual Basic 程序设计基础和一定实践经验,需要进一步深入了解 Visual Basic,进行较高层次软件开发的用户使用,也可供具有一定 Visual Basic 程序设计实践的读者自学或参考。书中提供了大量具有实用价值的应用实例。

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。

书 名: Visual Basic ActiveX 程序设计

作 者: 刘炳文 编著

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)

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

责任编辑: 焦 虹

印 刷 者: 北京昌平环球印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 **印张:** 14.25 **字数:** 328 千字

版 次: 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷

书 号: ISBN 7-302-05420-7/TP·3195

印 数: 0001~5000

定 价: 17.00 元

Visual Basic 是 Windows 操作系统平台上重要的开发工具之一。它因功能强大、简单易用而拥有数以百万计的用户。随着机器性能的提高和操作系统平台的升级, Visual Basic 的版本也在不断地更新, 逐步发展成为功能极强的大型程序设计语言。Visual Basic 具有易学习、功能强、见效快等突出特点, 因而备受人们青睐。它不仅是专业人员得心应手的开发工具, 而且易于被非专业人员掌握使用。

版本的更新和功能的增强, 使得 Visual Basic 的应用范围越来越广。它不但能满足一般应用程序设计的需要, 而且在数据库、ActiveX、多媒体、Internet、商用系统等多个领域都有着十分明显的优势。

目前国内大量的专业和非专业人员已不同程度地学习过 Visual Basic, 而且人数还在与日俱增。这些人中的大多数都希望在初步掌握了 Visual Basic 的程序设计方法之后, 能够深入了解 Visual Basic 的奥秘, 以便能开发出功能更加强大的应用程序。本书正是为了满足这些读者的需求而编写的。也就是说, 本书是给具有一定 Visual Basic 程序设计实践的读者使用的。

全书共 6 章, 主要介绍 ActiveX 程序设计。内容包括常用 ActiveX 控件、Windows 公共控件、对象、类和集合, 以及如何建立 ActiveX DLL 和 ActiveX EXE 组件, 如何建立 ActiveX 控件等。

考虑到目前图书市场上有关 Visual Basic “入门”的书比较多, 本书没有介绍 Visual Basic 程序设计的基础知识。笔者编著的《Visual Basic 程序设计教程》已由清华大学出版社出版, 《Visual Basic 高级程序设计》系列丛书的内容与该书前后衔接, 可以配套使用。

本书是《Visual Basic 高级程序设计》系列丛书中的第一本, 该系列丛书的第二本为《Visual Basic 图形与多媒体程序设计》, 第三本为《Visual Basic Internet 程序设计》, 已由清华大学出版社出版。这三本书介绍的都是 Visual Basic 较为深入的程序设计技术, 内容相对独立, 可以结合使用。

《Visual Basic 图形与多媒体程序设计》讲解了 Visual Basic 的图形与多媒体程序设计技术。主要介绍了 Visual Basic 本身提供的基本的图形程序设计方法和通过多媒体控件进行多媒体程序设计的技术, 同时介绍了 Windows 应用程序接口 (API), 并把它应用于 Visual Basic 的图形和多媒体程序设计中。

《Visual Basic Internet 程序设计》介绍的是 Visual Basic 中的 Internet 程序设计技术。内容包括: 超文本标记语言 (HTML)、脚本编写语言 VBScript、ActiveX 文档、动态

HTML 以及 Internet 控件与 Internet 程序设计。

目前, Visual Basic 在教学和软件开发中久盛不衰, 具有举足轻重的作用。随着它的进一步的推广应用, 想深入了解 Visual Basic 程序设计的人会越来越多。但愿本书能满足广大读者深入学习 Visual Basic 的强烈愿望, 对 Visual Basic 的应用和提高能起到一些作用。由于笔者水平有限, 加之时间较紧, 可供参考的资料不多, 错谬之处在所难免, 恳请专家和广大读者不吝赐教, 批评指正。

刘炳文

2002 年 6 月

第 1 章 ActiveX 技术	1
1.1 什么是 ActiveX	1
1.1.1 Windows 中的信息交换	1
1.1.2 ActiveX 简介	3
1.2 ActiveX 模型简介	4
1.2.1 ActiveX Internet 模型	4
1.2.2 ActiveX 模型	6
1.3 ActiveX 与其他技术	8
1.3.1 ActiveX 与 COM	8
1.3.2 ActiveX 与 Java	9
1.4 使用 ActiveX 的必要性	10
第 2 章 常用 ActiveX 控件	12
2.1 ActiveX 控件与可插入对象	12
2.1.1 ActiveX 控件	12
2.1.2 可插入对象	16
2.2 网格	18
2.2.1 网格的属性	19
2.2.2 网格事件	25
2.3 三维控件	27
2.3.1 三维命令按钮控件	28
2.3.2 三维组按钮控件	30
2.3.3 三维面板控件	32
2.4 量度表控件	37
2.4.1 量度表控件属性	37
2.4.2 程序举例	38
2.5 旋钮控件	41
2.5.1 事件和属性	41
2.5.2 程序举例	43

2.6	格式文本框控件	44
2.6.1	文件的装入和保存	45
2.6.2	字体设置	50
2.6.3	颜色设置	51
2.6.4	对齐与缩进	52
2.6.5	打印	54
2.7	Sysinfo 控件	56
2.7.1	属性	56
2.7.2	用 Sysinfo 控件检测系统信息	58
第 3 章	Windows 公共控件	62
3.1	概述	62
3.2	滑动器	64
3.2.1	设置滑动器的外观	64
3.2.2	滑动器的操作	66
3.3	状态条	71
3.3.1	建立状态条	71
3.3.2	面板对象的操作	78
3.4	进度条	81
3.4.1	进度条的显示位置	81
3.4.2	程序举例	82
3.5	工具条	86
3.5.1	工具条的位置与按钮对象	86
3.5.2	在工具条上添加按钮对象	88
3.5.3	按钮对象的大小和操作	92
3.5.4	显示工具提示文本	93
3.6	ImageList 控件	96
3.6.1	把图像保存到 ImageList 控件中	96
3.6.2	ImageList 控件的应用	99
第 4 章	对象、类和集合	105
4.1	Visual Basic 中预定义的类和对象	105
4.1.1	预定义对象和类	105
4.1.2	对象的操作	106
4.2	对象变量	109
4.2.1	对象变量的声明和赋值	110
4.2.2	继承	114
4.2.3	对象变量的释放	118



4.3	属性过程	119
4.3.1	属性过程的定义	119
4.3.2	属性过程举例	123
4.4	初始化事件和终止处理事件	127
4.4.1	Initialize 事件和 Terminate 事件	127
4.4.2	程序举例	128
4.5	建立自己的对象	129
4.5.1	类模块的基本概念	129
4.5.2	类的建立	132
4.6	对象浏览器	140
4.6.1	对象浏览器的基本功能	140
4.6.2	把对象描述信息添加到对象浏览器中	143
4.7	向类中添加事件	145
4.7.1	有关的语句和关键字	145
4.7.2	程序举例	146
4.8	集合	149
4.8.1	建立集合并向集合中添加项目	149
4.8.2	集合成员的删除和检索	153
第 5 章	建立 ActiveX DLL 和 ActiveX EXE 组件	159
5.1	ActiveX 组件	159
5.1.1	基本概念	159
5.1.2	在建立组件时应注意的几个问题	160
5.2	ActiveX EXE 组件	164
5.2.1	建立 ActiveX EXE	164
5.2.2	在其他应用程序中使用 ActiveX EXE 组件	166
5.3	建立含有窗体的组件	169
5.4	ActiveX DLL 组件	173
5.5	程序举例	178
5.5.1	建立 DLL	178
5.5.2	测试 DLL	181
第 6 章	建立 ActiveX 控件	184
6.1	引例	184
6.1.1	建立 Visual Basic 控件	184
6.1.2	建立 ActiveX 控件	190
6.2	基础知识	191
6.2.1	基本概念	191

6.2.2	UserControl 对象	192
6.3	用向导为 ActiveX 控件添加属性	194
6.3.1	建立控件	194
6.3.2	添加属性	196
6.3.3	测试用户控件	203
6.4	属性页	206
6.4.1	建立属性页	206
6.4.2	设计属性页	208
6.5	用户控件的方法和事件	213
6.5.1	为控件添加方法和事件	213
6.5.2	测试修改后的控件	217

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 或 World Wide Web 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，我们将对 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 什么是 ActiveX

Visual Basic 程序员想必早已感觉到，在软件开发中，ActiveX 几乎无处不在，如 ActiveX DLL、ActiveX EXE、ActiveX 控件、ActiveX 文档、ActiveX 数据对象(ADO)等。那么，什么是 ActiveX 呢？它与以前的技术有什么联系呢？下面就来介绍这个问题。

1.1.1 Windows 中的信息交换

在实际应用中，常常需要在两个进程之间交换信息。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，进程之间的信息交换较容易实现；随着版本的提高，它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)中，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。

例如，要把 NotePad 中指定的文本传送到 Word 中，可以先在 NotePad 中选择要传

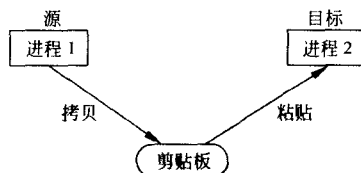


图 1.1 通过剪贴板在两个进程之间传送数据

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

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进程传送到另一个进程中,Microsoft 又引入了动态数据交换(dynamic data exchange,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,如图 1.2 所示。



图 1.2 通过 DDE 进行通信

动态数据交换是一种由 Microsoft Windows 操作系统支持的机制,它允许两个应用程序通过连续和自动地交换数据互相进行“对话”。DDE 使得应用程序之间数据的手工剪切和粘贴实现了自动化,为更新信息提供了一种较快的工具。

用动态数据交换可以使 Visual Basic 应用程序自动地接收从其他应用程序传送过来的数据,而当其他应用程序中的数据改变时,Visual Basic 应用程序中相应的数据将自动更新。反过来,也可以把 Visual Basic 应用程序中的数据传送给其他应用程序。因此,通过动态的数据链接,可以在两个不同的进程之间建立起一条“数据通道”,用来实现信息的动态交换和传递。

3. 对象链接与嵌入(object link and 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,因为必须了解服务器方的“数据格式”。为此,Microsoft 公司又开发出了对象链接与嵌入技术,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。DDE 是“数据”交换,而 OLE 可以看做“对象”交换。

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,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,如图 1.3 所示。

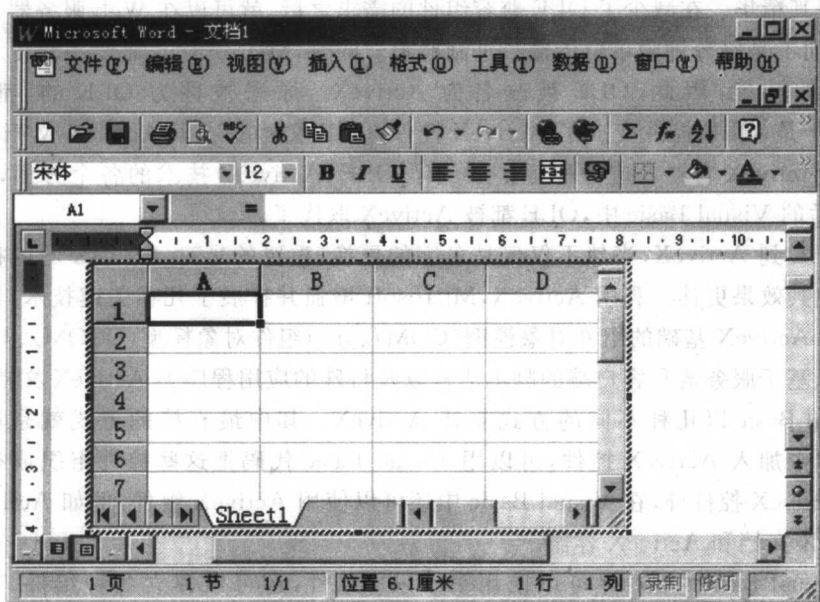


图 1.3 在 Word 文档中嵌入 Excel 工作表

1.1.2 ActiveX 简介

随着 Internet 的迅速发展与普及,人们把注意力转向了网络,这就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。也就是说,ActiveX 就是 OLE,或者更确切一点,ActiveX 所使用的核心技术是 OLE,而 ActiveX 这一术语实际上是 Microsoft 为了把 OLE 技术拓展到 Internet 而推出的新的名称。

在 Windows 的早期版本(Windows 3.0)中,Microsoft 就已致力于对象技术的使用。应用程序共享对象的最初规范是对象的链接与嵌入。如前所述,利用 OLE 技术,可以把 Excel 电子表格直接拖放到 Word 文档中,或者把两份文档“链接”起来;当修改其中的一份文档时,另一份文档可以自动更新。在 Visual Basic 环境下,程序员可以用内部控件和 VBX 控件组装应用程序。随后,Microsoft 又引入了两项新技术,淘汰了原来的 VBX 模式,并用 OLE 控件对象(即 OCX)取代。OLE 技术的最大贡献在于,实现了不同程序设计语言之间组件的共享。也就是说,用某种程序设计语言建立的组件,可以在另外一种程序设计语言中使用。事实上,在 4.0 版以前(或以后)的 Visual Basic 中,程序员所使用的 OCX 都是用 C++ 编写的,每个 OLE 组件都提供了新的功能,从而有效地扩展了操作系统。

OLE 的主要缺点是臃肿笨拙。当用 OLE 嵌入其他应用程序时,所需要的内存量较大,而且比非 OLE 应用程序要慢。这意味着,在完全理解 OLE 的操作和它所提供的功能之后,如果使用这种技术,就必须有内存空间大、运行速度快的计算机作为后盾。为了把 OLE 用到数据传输速度比本地硬盘慢若干倍的 Internet 上,Microsoft 对 OLE 进行了精

简,只保留其精华。在减少了 OLE 兼容组件的需求之后,就可以在 Web 服务器与浏览器之间以秒而不是以分钟为单位传输二进制对象。同时,Microsoft 也需要一套技术来增强 Windows 的功能,因此,OLE 被改名为 ActiveX。原来被称为 OLE 的,都改称为 ActiveX,于是,OLE 控件变成了 ActiveX 控件,OLE 文档变成了 ActiveX 文档,OLE 容器变成了 ActiveX 容器。这种称呼扩展到了 OLE(ActiveX)技术的各个方面,因此,在 5.0 版以后的 Visual Basic 中,OLE 都被 ActiveX 取代了。

从 OLE 到 ActiveX,当然不仅仅是名字的更换,重要的是对 ActiveX 的内核作了重新修正,使其效果更佳。利用 ActiveX,Microsoft 增强并扩展了几项关键技术,包括源于 OLE 作为 ActiveX 基础的组件对象模型(COM)、分布组件对象模型(DCOM)、与 Web 兼容的控件、基于服务器和客户端的脚本语言以及特殊的应用程序类 ActiveX 文档。

Visual Basic 以几种不同的方式支持 ActiveX。其中最直接的方式就是在 Visual Basic 窗体中加入 ActiveX 控件,可以用 Visual Basic 代码把这些控件组织成有用的程序。除 ActiveX 控件外,在 Visual Basic 中还可以使用 ActiveX 组件,例如 ActiveX 服务器、ActiveX 文档和 ActiveX 容器。

在 Visual Basic 中,不但可以使用 ActiveX 组件,还可以建立各种组件。事实上,Visual Basic 为每种 ActiveX 工程都提供了一个模板,这些模板可以从可视化模块开始(例如 ActiveX 控件),也可从代码模块开始(例如 ActiveX EXE 或 ActiveX DLL)。

ActiveX 为信息交换提供了一系列技术,诸如 ActiveX 控件、ActiveX 脚本处理、ActiveX 文档以及 ActiveX 代码组件(ActiveX code component)等,其中 ActiveX 代码组件又分为进程内组件(DLL)和进程外组件(EXE)。用 ActiveX 控件和进程内组件可以实现“对象类型”的共用,而进程外组件除了可以提供对象类型的共用之外,还可以实现对象及数据的共用。

1.2 ActiveX 模型简介

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连接,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。

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”。Internet 就是基于 HTTP 的,如图 1.5 所示。

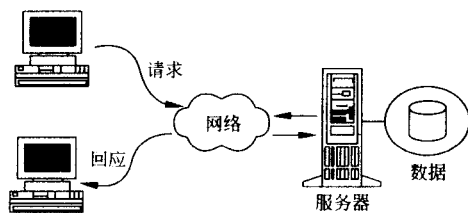


图 1.4 传统的客户/服务器结构

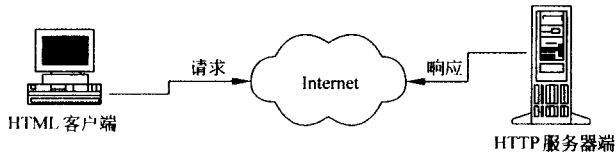


图 1.5 交互式 Web 模型

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。

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提交(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,可以生成一个基于 HTML 的响应,并由服务器发送回客户浏览器。

基于 CGI 的窗体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。

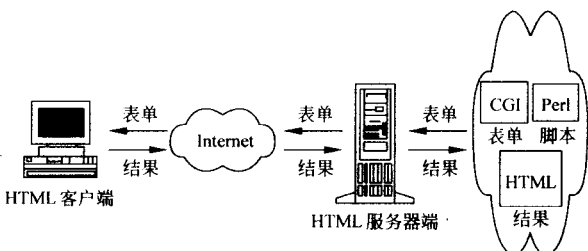


图 1.6 使用 CGI 的客户/服务器模型

为了克服上述模型的缺点,Microsoft 推出了专门为 Internet 设计的新型的计算机模型,即 ActiveX 平台,如图 1.7 所示。ActiveX 平台由 Active Desktop(客户端)和 Active Server(服务器端)组成,它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。

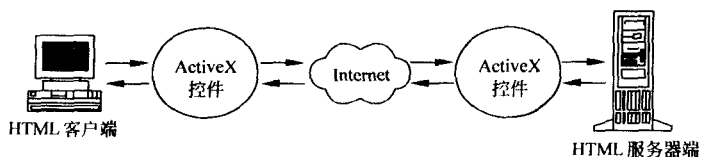


图 1.7 Microsoft ActiveX Internet 模型

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。ActiveX 由许多技术组成，而 ActiveX 模型是把这些技术连在一起的“粘接剂”。

ActiveX 模型包括以下内容：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)。

下面对这些内容进行简单介绍。

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一样。

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。

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。我们将在第 6 章介绍如何用 Visual Basic 建立 ActiveX 控件。

ActiveX 控件可用于各种语言环境中，也可以用于 Web 页面中。本书将介绍由系统提供的常用 ActiveX 控件。笔者编著的《Visual Basic 图形与多媒体程序设计》和《Visual Basic Internet 程序设计》分别介绍了与多媒体程序设计与与 Internet 程序设计有关的 ActiveX 控件。

2. ActiveX 脚本处理

ActiveX 脚本处理是在诸如 VBScript、JavaScript 这样的脚本语言中配合使用 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。《Visual Basic Internet 程序设计》一书将介绍脚本语言 VBScript 的程序设计。

3. ActiveX 文档

ActiveX 文档是一种 ActiveX 表示法的非 HTML 文档。如同用 Word,Excel 以及 PowerPoint 建立的 Microsoft Office 文档一样,通过 ActiveX 文档,可以将一个 Microsoft Office 文档加入到一个 Web 应用程序中,然后通过指定的浏览器来使用。ActiveX 文档所提供的功能具有极大的包容性,这对于团体用户来说是十分有用的。在一个企业中,可以通过 ActiveX 文档在企业的 Intranet 上为 Office 文档提供一个浏览器界面,例如,可以将员工手册或人才档案之类的文档加到 ActiveX 文档中。

ActiveX 文档具有以下优点:

- 易使用 通过浏览器界面使用,一般用户都比较熟悉。
- 投资少 只需购买一套 Office。
- 开销低 通过在企业内部的 Intranet 上提供诸如员工手册和政策方针之类的资料,可以降低文档维护、打印及发布等方面的开销。

《Visual Basic Internet 程序设计》一书将介绍 ActiveX 文档。

4. ActiveX 服务器

1996 年 11 月,Microsoft 推出了 ActiveX 平台。这是一个对称性的编程模型,由 ActiveX Desktop(客户端)和 ActiveX Server(服务器端)构成,如图 1.8 所示。

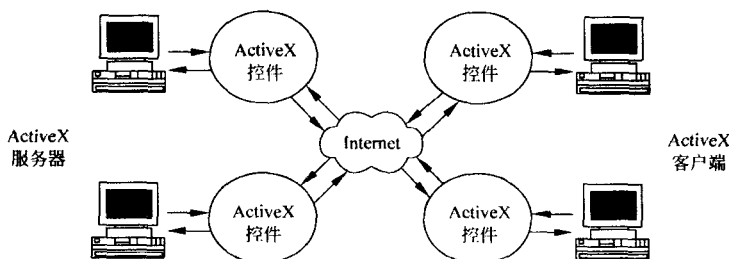


图 1.8 Microsoft 的 ActiveX 平台

在 ActiveX 平台中,客户端和服务端在本质和功能上是对称的,Active Client 和 Active Server 都含有 HTML、脚本处理和组件。服务器端的脚本处理是 Microsoft ActiveX 平台的一个功能强大的特征,通过它可以对发送请求的客户端的能力进行检测,并根据具体情况采取相应的服务。例如,如果客户端不支持 ActiveX 控件,则服务器可以向它提供静态文本和图像。Active Server 包括了 Internet Information Server(IIS)的一部分功能。

5. ActiveX 会议

为了在应用程序中加入研讨功能,Microsoft 推出了 ActiveX 会议(ActiveX

Conferencing)开发工具集(SDK),其中包括一个应用程序接口(API)。通过 ActiveX 会议 SDK,可以在 Windows 应用程序中加入诸如白板、程序共享、文件传输、语音通信、CHAT 及其他一些功能。ActiveX 会议通过 NetMeeting 来实现。

NetMeeting 是 Microsoft 推出的一个支持在局域网、Internet 或调制解调器的电话线上传输音频的 Web 会议应用程序。通过它可以实现:

- 与网络上的其他用户交谈;
- 与其他用户共享应用程序或文档;
- 在共享程序中与他人协作;
- 向其他用户发送文件。

以上介绍了 ActiveX 的几种技术,还有“Java 虚拟机”没有介绍,请见下一节。

1.3 ActiveX 与其他技术

ActiveX 不是一项孤立的技术,它与其他技术有着密切的联系。从某种程度上说,ActiveX 是建立在其他技术的基础上的。如果没有其他技术的支持,也就不可能有 ActiveX。在这一节中,我们将介绍 ActiveX 与 COM,Java 的关系。

1.3.1 ActiveX 与 COM

Microsoft 公司和 DEC 公司于 1995 年提出了组件对象模型(component object model,COM)和分布式组件对象模型(distributed component object model,DCOM)。COM 模型用于在异型网络和操作系统中建立、应用和加入可重复使用的组件对象。从理论上说,这些组件对象可以互相通信与交互,而与它们的语言、分布及原始平台无关。因此,COM 模型促进和鼓励开放式系统解决方案,其规程包括一套标准 API、一个标准的接口集以及 COM 用于支持分布式计算的网路协议。

在 COM 规程中,将组件定义为一个二进制形式的可重用的软件模块,可以在应用程序中直接使用这些组件,而不必担心它是用什么语言编写的,也不用担心它是在什么平台上开发的,只要知道组件的接口(或方法)集、调用方式、外部行为及如何与环境进行交互就可以使用它们。

目前的软件开发仍以手工方式为主,比较费力费时。利用 COM,可以把不同的组件“粘”在一起,完成软件编制,从而使开发时间大大缩短,这些组件构成了软件的功能模块。

DCOM 是一套用于分布式环境中的 COM 对象。在 DCOM 环境中,一个网络上的 COM 对象可以与另一个网络上的 COM 对象进行通信,如图 1.9 所示。

Microsoft 的 ActiveX 技术是遵循 COM 规程的。ActiveX 和 COM 的关系可以简单地描述如下:ActiveX 是建立在 COM 基础上的,它采用简化之后的 OLE,再加上网络支持,并使 OLE 随时处于动态(active)以处理信息。这种方式与 OLE 链接或嵌入的文档有所不同。

OLE 开发于 1990 年初,其着眼点在于提供复合文档的处理,即在文字处理器(如 Word)中可方便地链接(link)或嵌入(embed)电子表格(如 Excel)。OLE 的名称由此而