

最新 Internet 技术基础与应用系列丛书



用VB5 进行高性能 Web开发



Visual BASIC 5 Web Development

(美) Scott Jarol 著
译友翻译组 译



机械工业出版社

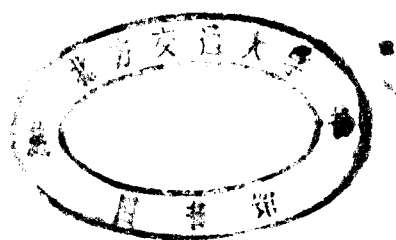


CMP

最新 Internet 技术基础与应用系列丛书

用 VB5 进行高性能 Web 开发

(美) Scott Jarol 著
译友翻译组 译



机械工业出版社

VB5.0 对 Internet 的联网协议和 ActiveX 具有内置的支持,也是 Microsoft 的 Web 对象技术。本书应用面向对象的编程技巧,开发了一些在 Web 和 Internet 编程方面的最先进的详细细目。在本书中将会学到如何应用最新的 Internet Transfer Control 浏览 Web。也将用到 VB 数据库强大的存取功能编写基于 CGI 的 Web 应用程序。另外,也将会学到如何编写自己的 Web ActiveX 控件,通过创造新的应用程序就可以掌握 Internet 上惊人的资源。

Scott Jarol: VISUAL BASIC 5 Web Development.

Authorized translation from the English language edition published by The Coriolis, Inc.

Copyright 1997 by The Coriolis, Inc.

All rights reserved.

本书中文简体字版由机械工业出版社出版,未经出版者书面许可,本书的任何部分不得以任何方式复制或抄袭。

版权所有,翻印必究。

本书版权登记号:图字:01-98-0623

图书在版编目(CIP)数据

用 VB5 进行高性能 Web 开发/(美)斯科特(Scott,J.)著;译友翻译组译.-北京:机械工业出版社,1998.6

(最新 Internet 技术基础与应用系列丛书)

书名原文:Visual BASIC 5 Web Development

ISBN 7-111-06320-1

I.用… II.①斯… ②译… III.万维网-BASIC 语言-程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(98)第 10104 号

出 版 人:马九荣(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:温莉芳 高瑞霞

北京市南方印刷厂印刷·新华书店北京发行所发行

1998 年 6 月第 1 版第 1 次印刷

787mm×1092mm 1/16·20.25 印张

印数:0 001-8 000 册

定价:53.00 元(附光盘)

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

前言

面对众多的编程语言、脚本语言、HTML 标志、API、对象规范、媒体视窗、Plug-in 和 applet 等，Web 开发者经历 Java 震惊之后，便会感受到新的关于 ActiveX 的热潮。World Wide Web 的推广引起了技术革新的风暴，使 Web 开发者面临众多的丰富资源。在如此众多的选择中，必然有许多重叠，也有不少混淆。解决任何特定问题的技术数目，就像联机数据库发布工具数目一样惊人的巨大。仅 Microsoft 就提供了几种重叠的技术，包括数据库引擎和接口、脚本语言以及 HTML 编辑工具。

如此激烈的技术竞争下，Web 开发者之间最为热烈的争论却是跨平台兼容性问题，也就是有关 Web 站点不考虑客户机、操作系统或浏览器之间的差别，在外观和功能上都相同的问题。争论者主张在各种支持高级平台相关功能的方案和技术之间折衷。

对那些只想建立联机站点的人来说，选择是惊人的。Webmaster 能掌握多少种编程语言和媒体准备工具呢？我们总想找到能把自己的知识用到各个任务之中的解决方案。那么，请想像一种能够构造包括数据库驱动 CGI 应用程序、浏览器 plug-in、客户端脚本以及各种 Internet 实用程序和服务的各种各样的 Web 应用程序的编辑语言吧！这种语言可能就是 Visual Basic。

尽管现在用 Visual Basic 编写的程序只能在 Windows 95 或 Windows NT 上运行，但不能夸大这个限制。如果你的应用程序不需跨平台兼容性的能力，那么 Visual Basic 就是你所需要的编程语言。在强大的内置功能、轻易使用 Windows API 和以 ActiveX 组件形式的广泛第三方支持等优势下，Visual Basic 使我们能创建网络或其他类型的 Windows 应用程序。即使应用程序具有确实需要跨客户端平台的功能，Visual Basic 仍然是实现基于服务器的 Web 或 Internet 站点元素的最好方式。Visual Basic 提供了一组开发如数据库驱动 CGI 脚本和特定目的 Internet 服务器等的服务器应用程序的工具，替代了晦涩的 Perl 脚本或 C 代码。如果选择实现基于 Windows 的服务器站点，Visual Basic 就是有用的工具之一。

尽管 Rosetta Stone 没有把 Visual Basic 视为 Internet 技术，但 Visual Basic 语言家族（现在包括 Visual Basic 5）、Visual Basic 应用程序和 VBScript 提供了 Web 和 Internet 编辑的广泛支持。

章节概述

首先由第 1 章快速入门探讨 Visual Basic 5 Web 编程。在第 1 章中，将试用 Visual Basic 5 最新的 Internet 编程功能，包括 ActiveX 编程和 Winsock 控件。

在第 2 章中，介绍 Visual Basic 的 Internet 和 Web 编程的各个方面。

第 3 章用 Internet 传输控件实现 Web Robot。Robot 是能在 Web 中搜索符合需求文档的特殊 Web 客户端。第 4 章再详细介绍 Web Robot 用以按搜索字符串分析文档的分析器。

在第 5、6 章中，我们实现另外一种 Internet 客户端：News Reader。但是，与大多数新闻客户端不同的是，我们的客户端嵌入了第 4 章中开发的文本分析器，使之成为智能 News

Reader。另外还将讨论 Winsock 控件，用来实现网络新闻传输协议（NNTP）。从此，便可以书写自己的 Internet 实用程序了。

第 7、8 章介绍 Visual Basic CGI 编程。Visual Basic 内置的数据库能力使数据库驱动 Web 站点的开发变为迅速和容易。用 WebSite Web 服务器产品的作者 Robert Denny 开发的代码库，我们将在 Visual Basic 中书写说明如何把整个 Web 站点封装在单个可执行文件的 CGI 脚本。另外，还讨论如何解决 CGI 棘手问题之一：建立持久客户端状态（其中讨论了 cookie）。

在第 9 章中，我们用 Visual Basic 开发 ActiveX 控件。将从头开始构造 ActiveX 控件并提出影响 Web 所用控件的特殊考虑。

最后，第 10 章书写自己的常规 Internet 应用程序，一个带独立客户端和服务器的聊天系统。一旦实现基本的系统，就把客户端转换成能在用户计算上自动下载、安装并运行的 ActiveX 控件。

目 录

前言	
第 1 章 快速入门	1
1.1 ActiveX 的新世界	1
1.1.1 快速构造 Web 浏览器	1
1.1.2 快速创建 ActiveX 控件	4
1.2 用 Winsock 进行 Internet 通信	6
1.2.1 Winsock 服务器	6
1.2.2 Winsock 客户端	8
1.3 大量强有力的工具	10
第 2 章 Visual Basic 和 Internet	11
2.1 Web 和 Internet 编程	11
2.1.1 Internet 实用程序	11
2.1.2 新的 Internet 应用程序	12
2.1.3 CGI 和数据库发布	12
2.1.4 ActiveX Web 控件	12
2.1.5 混合匹配技术	13
2.2 关于对象的陈述	14
2.2.1 Visual Basic 对象类	14
2.2.2 Visual Basic 基本对象	14
2.2.3 共享对象	15
2.2.4 谁需要指针	15
2.3 继续探讨	16
第 3 章 轻松遍历 Web	17
3.1 Agent 和 Bot	17
3.2 用 HTTP 搜集文档	18
3.2.1 请求文档	18
3.2.2 头及其字段	18
3.2.3 数据对象传输	19
3.3 运行的 HTTP	19
3.3.1 设计 HTTP 实验 (Experiment)	
工程的窗体	19
3.3.2 执行 HTTP 文档传输	20
3.4 树形化 Web	23
3.4.1 建立超链的树形图	23
3.4.2 运行 Web Explorer	24
3.4.3 设计 Web Explorer 窗体	24
3.4.4 请求 Web 文档	26
3.4.5 接收文件	30
3.4.6 抽取超链	33
3.4.7 在 TreeView 控件中显示超链	44
3.4.8 管理部分的代码	46
3.5 构造 Web Robot	48
3.5.1 运行 Web Robot	48
3.5.2 设计 Web Robot 的窗体	48
3.5.3 改变窗体代码	64
3.5.4 漫游 Web	68
3.5.5 不要让 Robot 控制服务器	75
第 4 章 VB 文档分析器	78
4.1 分析器原理	78
4.2 VB 文档分析器	79
4.2.1 执行分析器	79
4.2.2 分析器奇特的工作过程	83
4.2.3 其他操作符的过程	85
4.2.4 读取单词	88
4.2.5 clsDocuments 的其他方法	93
4.2.6 类 clsMatchLists 和 clsMatch	95
4.3 利用分析器	96
第 5 章 阅读新闻	97
5.1 网络新闻传输协议	97
5.2 Internet News 实验	101
5.2.1 运行 NNTPExp	101
5.2.2 构造窗体	102
5.2.3 Winsock 控件	103
5.2.4 NNTP 操作	103
5.2.5 frmNNTPExp.FRM 详细清单	110
5.2.6 NNTP 实验工程总结	115
第 6 章 剔除 Internet 新闻中的糟粕	116
6.1 开发完整的 Internet 新闻客户端	116
6.1.1 运行 Visual Basic News Reader	116
6.1.2 创建主窗体	118
6.1.3 保存新闻组名	119
6.1.4 从数据库到 Tree View	124
6.1.5 选择新闻组	126
6.1.6 获取新闻文章	131
6.1.7 粘贴新闻文章	131
6.1.8 过滤新闻	134
6.1.9 Connection Information 窗体	153
6.1.10 Article Posting 窗体	154

6.1.11 实用函数	155	8.2.7 工程: 用 Cookie 建立用户上下 文	233
6.2 获取 Internet 的有用文档	156	8.2.8 安全措施	243
第 7 章 无所畏惧的 CGI	157	8.2.9 设计具有用户上下文的站点	243
7.1 CGI 的基础	157	第 9 章 创建 Web 的 ActiveX 控件	244
7.1.1 HTML 窗体	157	9.1 实现 ActiveX 的历程	244
7.1.2 Windows CGI 编程	159	9.2 ActiveX 控件	245
7.2 简单的 CGI 程序	159	9.3 工程: Rollover Menu ActiveX Web 控件	246
7.3 谁需要文档文件	160	9.3.1 运行 RolloverMenu 控件	246
7.4 构造 Web 窗体	162	9.3.2 设计 RolloverMenu 控件	247
7.4.1 <INPUT>标志	162	9.3.3 命名类	247
7.4.2 <SELECT>标志	166	9.3.4 说明控件属性	248
7.4.3 <TEXTAREA>标志	168	9.3.5 设置 Procedure ID	254
7.5 测试窗体	169	9.3.6 容器器提供的属性	255
7.6 VB 和 CGI 数据库连接	172	9.3.7 用属性页面构造复合属性	256
7.7 简单的数据库驱动的 CGI 应用程序	172	9.3.8 保存和读取属性	264
7.7.1 编制 CGI 数据库实验工程的代 码	173	9.3.9 Resize 事件	269
7.7.2 构造数据库	174	9.3.10 激发事件	270
7.7.3 使用数据库	175	9.3.11 详细清单	271
7.8 用 HTML 表格返回 CGI 查询结果	179	9.4 编译 ActiveX 控件	281
7.8.1 HTML 表格	179	9.4.1 No Compatibility	282
7.8.2 HTML 表格标志	180	9.4.2 Project Compatibility	282
7.8.3 发送表格	182	9.4.3 Binay Compatibility	282
7.9 数据驱动的事件日历	187	9.4.4 把控件编译成 OCX	282
7.9.1 运行 CGI 事件日历	187	9.5 创建 Internet 安装文件	282
7.9.2 表格的烦恼	187	9.6 在 Web 文档中加入控件	287
7.9.3 表格对象模型	188	9.7 关于 ActiveX 文档的总结	288
7.9.4 用表格构造日历	199	第 10 章 网上交谈	289
7.9.5 完成 CGI 事件日历程序	202	10.1 重温 Winsock	289
7.10 结合 VB5 使用 CGI	206	10.2 Winsock Chat 应用程序	290
第 8 章 高级 CGI	208	10.2.1 运行 Winsock Chat 程序	291
8.1 Visual Basic 构造 Web 站点	208	10.2.2 设计服务器窗体	291
8.1.1 用 CGI 逻辑路径区分页面	208	10.2.3 服务器接受连接	292
8.1.2 运行 CGI Web Site 实验程序	208	10.2.4 服务器发送文本	295
8.1.3 编辑 CGI Web Site 实验程序的 代码	209	10.2.5 从客户端接收文本	296
8.2 Web 站点上的花名册	213	10.2.6 关闭服务器的连接	297
8.2.1 用 IP 地址标识客户端	214	10.2.7 设计客户端窗体	301
8.2.2 工程: 用 IP 地址建立用户上下 文	214	10.2.8 控制与服务器的连接	302
8.2.3 用逻辑路径标识符标识客户端	226	10.3 Web 中的 ActiveX Chat	308
8.2.4 工程: 用逻辑路径标识符建立用 户上下文	226	10.3.1 从 EXE 到 OCX	308
8.2.5 用 Cookie 标识客户端	229	10.3.2 用 Web Chat 连接	308
8.2.6 工程: Cookie 实验程序	231	10.3.3 编译并使用 ChatterBox 控件	316
		10.3.4 增强 ChatterBox 控件	317
		10.4 设计自己的 Internet 协议	317

第1章 快速入门

Visual Basic 5 使编写各种 Web 和 Internet 应用程序更加容易。现在，我们就从大量简单的程序开始，这些程序显示了 Visual Basic 5 编程系统的强大功能和灵活性。

Visual Basic 提供了大量对 Web 和 Internet 软件开发有深远影响的功能。像早期的发布版一样，VB5 提供了强有力的新工具，使得曾被视为神秘的编程工作简单化。如今，Redmond 中的向导加强了 Visual Basic 的以下三项关键技术：

- 支持 ActiveX 的开发
- Winsock 控件
- Internet 传输控件

现在我们就快速了解一下 Visual Basic 如何使 Web 编程变得容易。

1.1 ActiveX 的新世界

Visual Basic 的三个早期发布版都是基于新的控件体系结构 VBX 之上的。VBX 以定制控件的形式为语言提供了广泛的第三方支持。到 VB4 为止，Visual Basic 的创建者们已经结合了 Microsoft 的对象链接和嵌入（OLE）技术。现在的 Visual Basic 支持两种控件：其固有的控件和 OCX 控件，Visual Basic 固有控件封装了标准的 Windows 控件，如滚动条和下拉框；而 OCX 控件遵从 ActiveX 规范，ActiveX 规范是一种可交换程序组件的体系结构。

Microsoft 实践着他们一贯倡导的原则。Redmond 中出现的绝大多数新应用程序都是 ActiveX 组件的集合体，而我们也可以把大多数 ActiveX 组件复用到自己的应用程序中。譬如说 Internet Explorer 3.0 就是基于能搜索和显示 Web 的 HTML 文档的 ActiveX 控件之上的。如果安装了 Internet Explorer 3.0，就能很容易地把这些控件嵌入自己的 Visual Basic 工程之中。

1.1.1 快速构造 Web 浏览器

打开 Visual Basic 5 并启动一个 Standard.EXE 工程，展开窗体设计窗口中的窗体，便可开始自己的工作，如图 1-1 所示。

把光标放在 Tool box 之上并右击鼠标显示上下文菜单，然后选取 Components 以显示 Components 对话框，如图 1-2 所示。

选取 Controls 选项显示已经安装登记的控件，浏览可用的组件列表时，会发现有些以 IE 开头。这些 IE 开头的控件是由 Internet Explorer 安装的，可复用到自己的工程中。但 Internet Explorer WebBrowser 控件位于 Microsoft Internet Controls 组件列表的下端，选中时，对话框会显示它的位置，诸如：C: \ WINDOWS \ SYSTEM \ SHDOCW.DLL。WebBrowser 控件在 DLL 之中，而不是 OCX 之中，但是，它却是 OLE 组件。选取 Microsoft In-

Internet Controls 组件之后，单击 OK 关闭对话框，控件便会出现现在 Tool box 之中，用一个放大镜下的文件夹表示，如图 1-3 所示。

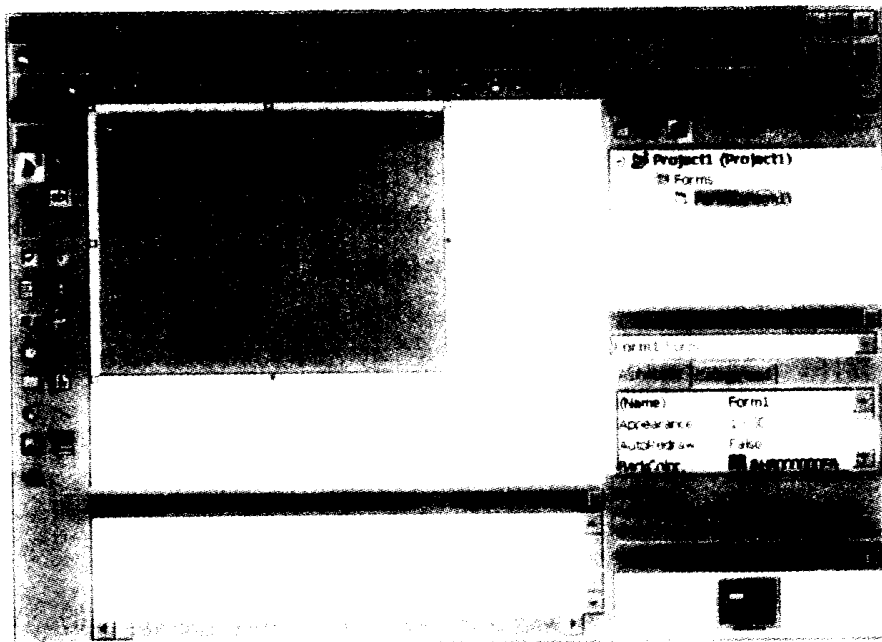


图 1-1 带新窗体设计窗口的 Visual Basic 5 开发环境

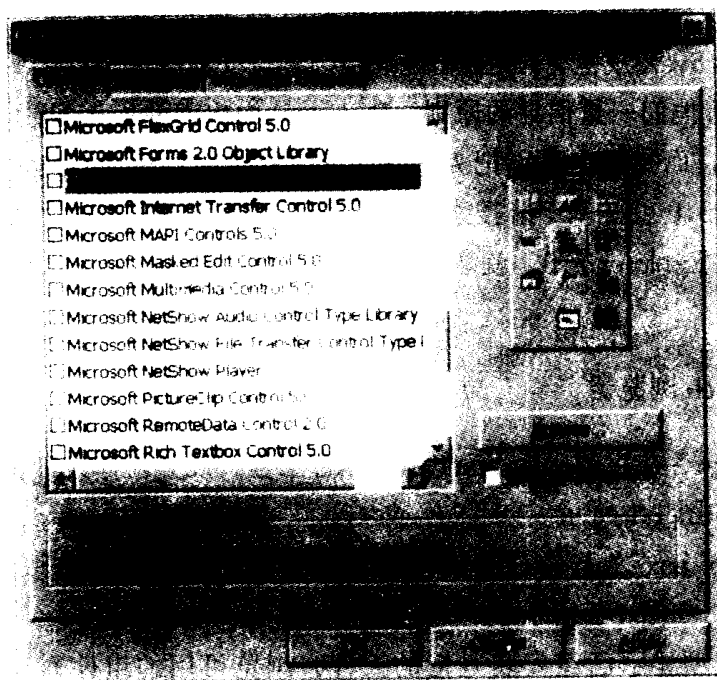


图 1-2 Visual Basic 的 Components 对话框

从 Tool box 中选择新加入的控件，并把一个实例拖放到窗体中，展开使之占据大部分可用空间。然后，双击窗体，打开 Code 窗口中的 Form_Load() 方法，加入一条语句，调用 WebBrowser 控件的 Navigate() 方法，如程序清单 1-1 所示。

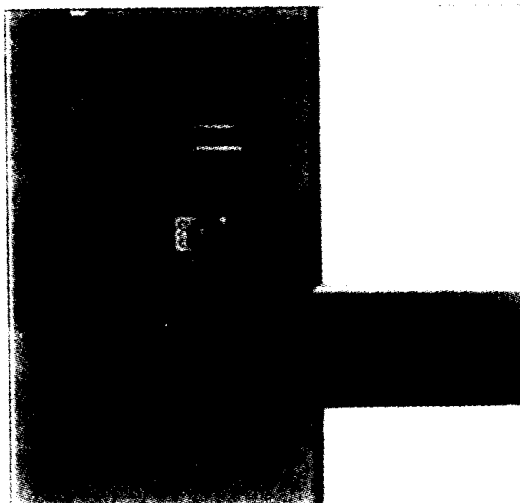


图 1-3 显示在 Visual Basic Tool Box 中的 Browser 控件

程序清单 1-1 从 Form_Load () 事件中激活 WebBrowser 控件

```
Private Sub Form_Load ()
    WebBrowser1.Navigate "http: //www.mediaterra.com"
End Sub
```

激活 TCP/IP 的 Internet 连接，再运行程序。此时，应当装载并显示 Form_Load () 事件过程中指定的 Web 文档，如图 1-4 所示。

用 Internet Explorer WebBrowser 控件，就可以构造一个基于 HTML 和 ActiveX 组件技术的用户界面的多媒体应用程序，甚至可以构造无缝嵌入 CD-ROM 和 Web 内容的产品。

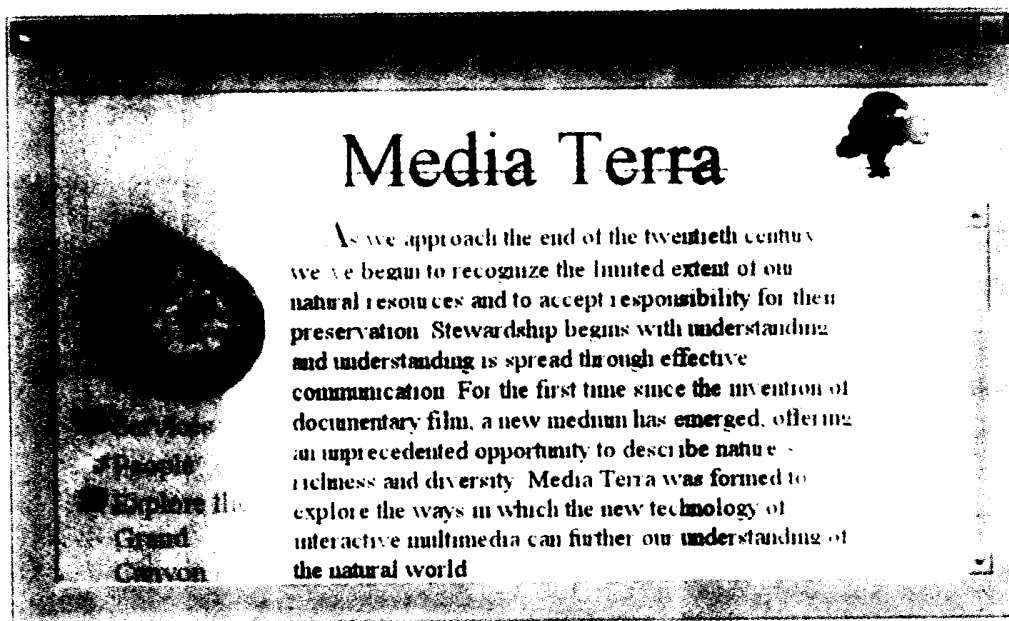


图 1-4 激活的 WebBrowser ActiveX 控件

ActiveX 组件利用了操作系统的新资源，使得不仅能用于起初设计的应用程序，也能用于其

他的应用程序。

Visual Basic 4 就引入了对 ActiveX 控件的支持——至少能用于 Visual Basic 应用程序之中。要创建新的控件，必须掌握复杂的语言如 C++，但 Visual Basic 5 能使我们在 Visual Basic 自身之中就可以创建 ActiveX 控件。

1.1.2 快速创建 ActiveX 控件

在第 9 章中，我们将详细阐述控件的创建过程。现在，我们就可以发现构造自己的 ActiveX 控件是件很容易的事。

从 Visual Basic 菜单中选择 File/New Project 项，再选择 ActiveX Control 作为工程类型。首先看到的是 User Control 设计窗，如图 1-5 所示。像窗体一样，可以在 User Control 对象的背景上可视化设计自己的控件，可以利用现存的控件，也可以用图形方法绘制。

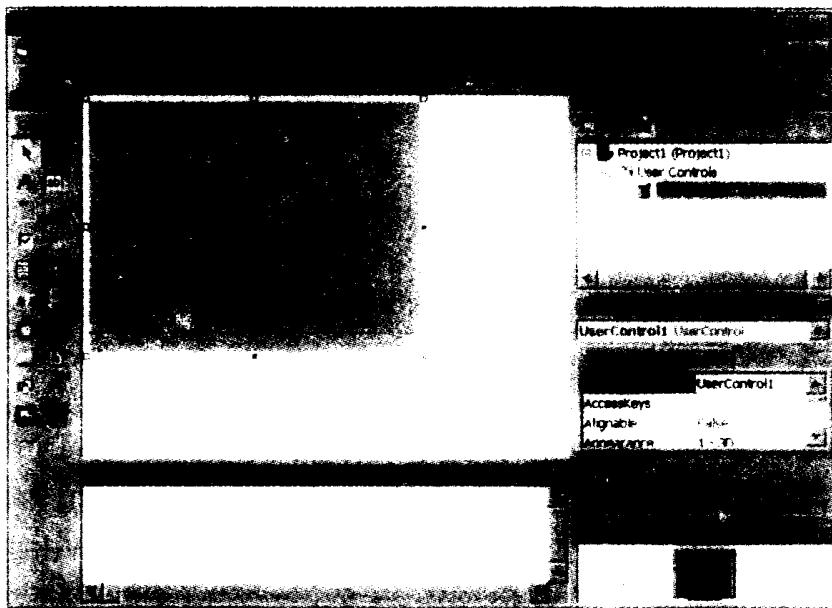


图 1-5 显示 User Control 设计窗的 Visual Basic

继续进行之前，先赋一些名称。在 Properties 窗口中，选 User Control 的 Name 属性为 Marquee Control，从 Visual Basic 菜单中选择 Project | Project1 Properties 项，在 Project Properties 对话框中，置 Project Name 为 Marquee ControlProject。最后，用文件名 Marquee ControlProject.VBP 和 Marquee Control.CTL 保存 Project。

在这个简单的工程中，我们用一个 Picture Box 控件和一个 Timer 来创建一个滚动字幕，如图 1-6 所示。在 User Control 设计窗中拖放一个 Timer 控件，将 Name 属性设为缺省值 Timer1，把 Interval 属性设为 50。然后，在 User Control 设计窗中放置一个 Picture Box，展开

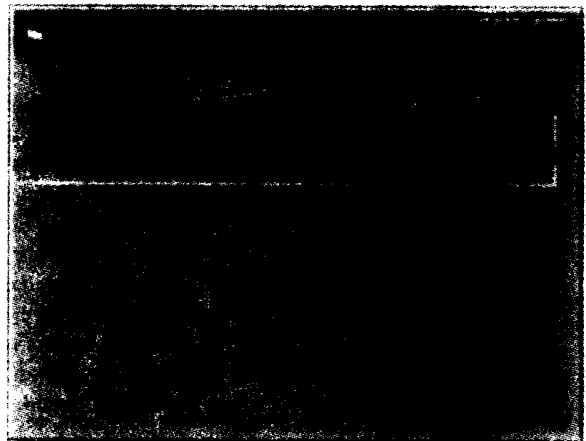


图 1-6 激活的 ActiveX 字幕控件

使之布满整个 User Control 对象，将 Name 属性设为 pbMarquee，将 BorderStyle 属性置为 1-Fixed Single。程序清单 1-2 显示了该工程的代码。

程序清单 1-2 Marquee Control.CTL 详细清单

```

Option Explicit
Private TextXPos As Long
Private TextTop As Long
Private TextSize As Long

Const Caption="What?! Another marquee?"

Private Sub Timer1_Timer ()
    If Ambient.UserMode Then
        pbMarquee.Cls
        If (TextXPos+pbMarquee.TextWidth (Caption)) < 0 Then
            TextXPos=pbMarquee.ScaleWidth
        End If
        TextXPos=TextXPos-25
        pbMarquee.CurrentX=TextXPos
        pbMarquee.CurrentY=TextTop
        pbMarquee.Print Caption
    End If
End Sub

Private Sub UserControl_Resize ()
    TextXPos=pbMarquee.ScaleWidth
    pbMarquee.Left=0
    pbMarquee.Width=UserControl.ScaleWidth
    pbMarquee.Top=0
    pbMarquee.Height=UserControl.ScaleHeight
    pbMarquee.FontSize=pbMarquee.ScaleY _
        (pbMarquee.ScaleHeight*0.8, vbTwips, vbPoints)
    TextTop= (pbMarquee.ScaleHeight\2) _
        (pbMarquee.TextHeight ("A") \2)
    If Ambient.UserMode Then
        Timer1.Enabled=True
    Else
        Timer1.Enabled=False
    End If
End Sub

Private Sub UserControl_Terminate ()
    Timer1.Enabled=False
End Sub

```

这个控件的所有动作都发生在 Timer1_Timer () 事件过程中，该过程先调用 CLS ()

方法清除 pbMarquee 的当前内容。设置 Picture Box 的当前坐标之后，用 Print（）方法显示常量 Caption 指定的字符串，按时钟慢慢地移向左边，直到滑出左边缘为止。然后，再从右边重复该过程。

输入程序清单 1-2 中的代码。这段代码定义了有完整功能的 ActiveX 控件，尽管没有很大的用途。从 Visual Basic 菜单中的 File | Add Project，加入 StandEXE，就可以测试该控件。

关闭 Marquee Control 的 User Control 设计窗时，Visual Basic 编译并登记该控件的临时版本。然后，就可以从 Tool Box 中选择该控件并拖一个实例到窗体中。运行 Standard EXE 程序激活这个新的控件。

要改变该控件显示的信息，先关掉正在运行的窗体。然后，在 Project Explorer 中，右击 Marquee Control 对象并选择 View Code，修改 Caption 的内容，再运行 Standard EXE 工程。

尽管 Visual Basic 使创建基本的 ActiveX 控件很容易，但真正有用的控件很可能有各种不同的属性、事件和方法。而写一个完整的控件就会变得相当复杂。第 9 章将探讨控件创建的几个方面，然后，明确如何把定制的 ActiveX 控件用于 Web 文档中。

1.2 用 Winsock 进行 Internet 通信

Internet 的复杂性主要来自于各种各样的通过 Internet 交换信息的应用程序，包括电子邮件、新闻和 World Wide Web。但 Internet 本身却具有良好定义的和可靠的通信协议的数字网络。

基于这个简单的基础、Windows 95 和 Windows NT 操作系统为 Internet 提供了诸如 Windows Sockets 或 Winsock 等标准接口。而 Visual Basic 则通过 Winsock ActiveX 控件为 Winsock 提供了标准接口。Winsock 的目的相当简单——为两个或多个 Internet 节点建立连接并使之交换数据。下面我们以一个简单的应用程序来看看这个过程。

1.2.1 Winsock 服务器

本工程用 TCP 协议为客户端和服务端两个节点建立连接。对实际的 Internet 应用程序而言，可能需要两个独立的应用程序。但现在为简单起见，我们在一个应用程序中，为客户端和服务端建立两个不同的窗体。

先打开一个新的 Standard EXE 工程，而后实现 Winsock 服务器，如图 1-7 所示。为此，需要如表 1-1 所列的五个控件。

1. 倾听 Winsock 连接

Internet 上的每个服务器应用程序都通过 IP 地址和端口号两个数字标识，服务器正是通过这两个数字来倾听连接请求。每个 Internet 节点，也就是说，有特定 IP 地址的计算机，都可运行多个服务器，只要这些服务器倾听的端口号不同。在程序清单 1-3 所示的服务器 Form_Load（）事件过程中，我们将 Winsock

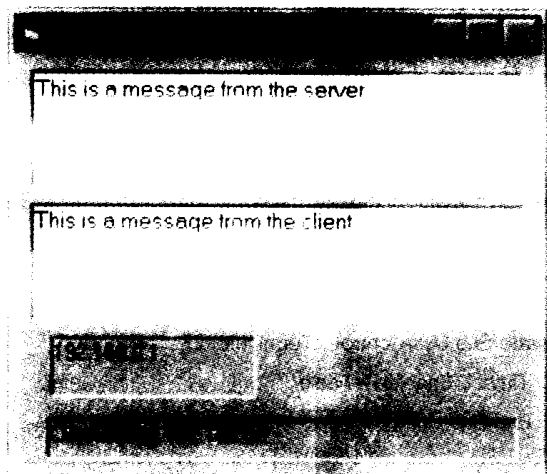


图 1-7 WinsockExperiment1.VBP
中的 frmServer

控件的 LocalPort 设为 1600。

表 1-1 frmServer1.FRM 中的控件

名 称	类 型	属 性	值
lblDataReceived	Label	Caption	“ ”
lblIPAddress	Label	Caption	“ ”
lblStatus	Label	Caption	“ ”
sktTCPChatServer	Winsock	Protocol	0-sckTCPProtocol
txtDataToSend	TextBox	MultiLine	True

由于 IP 地址由服务器运行的计算机确定，因而无需设定。调用 Listen () 方法，服务器便开始监听 Internet 的连接请求，这样便激活了服务器。

程序清单 1-3 工程 WinsockExperiment1.VBP 中的 frmServer1.FRM 详细清单

Option Explicit

Private Sub Form_Load ()

frmClient.Show

sktTCPChatServer.LocalPort=1600'Set the local port.

sktTCPChatServer.Listen 'Use the Listen method.

lblIPAddress.Caption=sktTCPChatServer.LocalIP

lblStatus.Caption="Host Name:" & _

sktTCPChatServer.LocalHostName

End Sub

Private Sub Form_QueryUnload (Cancel As Integer, _

UnloadMode As Integer)

Dim ArrayIndex As Integer

Unload frmClient

End Sub

Private Sub sktTCPChatServer_Close ()

lblStatus.Caption=_

"Disconnecting from" & sktTCPChatServer.RemoteHostIP

'Return to listening state.

sktTCPChatServer.Close

sktTCPChatServer.LocalPort=1600

sktTCPChatServer.Listen

End Sub

Private Sub sktTCPChatServer_ConnectionRequest _

(ByVal requestID As Long)

Dim Dummy As String

If sktTCPChatServer.State<>sckClosed Then

sktTCPChatServer.Close

End If

```

sktTCPChatServer.Accept requestID
lblStatus.Caption = "Connecting" & _
    sktTCPChatServer.RemoteHostIP
End Sub

Private Sub sktTCPChatServer_DataArrival (ByVal bytesTotal As Long)
    Dim DataReceived As String
    sktTCPChatServer.GetData DataReceived, vbString
    If CStr(DataReceived) = "QUIT" Then
        sktTCPChatServer.Close
    Else
        lblDataReceived.Caption = DataReceived
    End If
End Sub

Private Sub txtDataToSend_Change ()
    sktTCPChatServer.SendData txtDataToSend.Text
End Sub

```

2. 建立 Winsock 连接

当客户端向合适的 IP 地址和端口号发送连接请求时，Winsock 控件便会产生 Connection Request 事件。在 sktTCPChatServer_ConnectionRequest () 事件过程中，我们通过调用 Winsock 控件的 Accept () 方法，并将所接收的 requestID 作为事件过程的参数，就可以在服务器和客户端之间建立连接。

3. 交换数据

如程序清单 1-3 所示的事件过程 txtDataToSend_Change () 一样，使用 Winsock 控件的 SendData () 方法，就可以向客户端发送数据。Winsock 控件收到客户端的数据时，先调用程序清单 1-3 所示的 DataArrival () 事件过程，然后再通过 GetData () 方法从通信缓冲区中获取数据。

1.2.2 Winsock 客户端

对客户端而言，先创建类似于服务器的窗体，如图 1-8 所示，该窗体有两个文本框，分别显示接收和发送的信息。还需要两个 Command Button 以便连接到服务器上。表 1-2 列出了 frmClient 所需的六个控件。

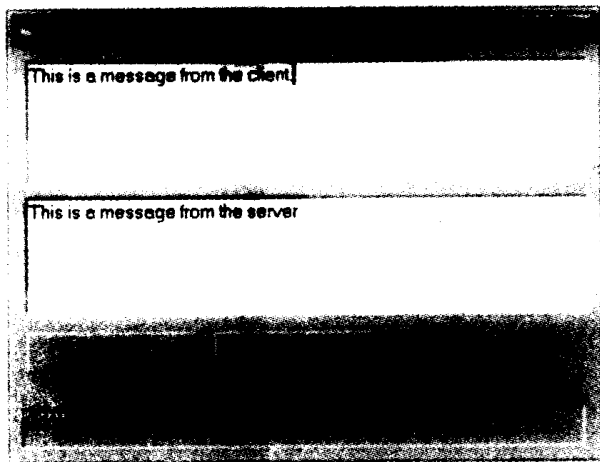


图 1-8 WinsockExperiment1.VBP 中的 frmClient

表 1-2 frmClient1.FRM 中的控件

名 称	类 型	属 性	值
cmdCloseConnection	CommandButton	Caption	"Connect"

(续)

名 称	类 型	属 性	值
cmdConnect	CommandButton	Caption	"Close Connection"
lblStatus	Label	Caption	" "
sktTCPChatClient	Winsock	Protocol	0-sktTCPProtocol
txtDataReceived	Label	MultiLine	True
txtDataToSend	TextBox	MultiLine	True

连接到服务器上

客户端和服务器的主要差别在于建立连接的方式不同。服务器通过倾听建立连接，而客户端则是用 Winsock 控件的 Connect () 方法以及 RemoteHost 和 RemotePort 来发送连接请求。本程序中，因为服务器和客户端运行在同一台机器上，所以将 RemoteHost 的值设置为 Winsock 控件所检测到的 LocalIP。在 frmClient 中，通过执行 cmdConnect _ Click () 事件过程发送连接请求，如程序清单 1-4 所示。

程序清单 1-4 工程 WinsockExperiment1.VBP 中的 frmClient1.FRM 的详细清单

```
Option Explicit

Private Sub cmdCloseConnection _ Click ( )
    lblStatus.Caption="Closing Connection"
    'Don't terminate the connection like this:
    '    sktTCPChatClient.Close
    sktTCPChatClient.SendData"QUIT"
End Sub

Private Sub cmdConnect _ Click ( )
    If sktTCPChatClient.State<>sckClosed Then
        sktTCPChatClient.Close
    End If
    With sktTCPChatClient
        .RemoteHost=sktTCPChatClient.LocalIP
        .RemotePort=1600
    End With
    sktTCPChatClient.Connect
End Sub

Private Sub Form _ Load ( )
    lblStatus.Caption="Local IP Address:"& _
        sktTCPChatClient.LocalIP
End Sub

Private Sub Form _ QueryUnload (Cancel As Integer, _
                                UnloadMode As Integer)

    sktTCPChatClient.Close
End Sub
```

```
Private Sub sktTCPChatClient _ Close ()
    sktTCPChatClient.Close
    Do
        lblStatus.Caption="Status:" & _
            sktTCPChatClient.State
        DoEvents
    Loop Until sktTCPChatClient.State=sckClosed
End Sub

Private Sub sktTCPChatClient _ Connect ()
    If sktTCPChatClient.State=sckConnected Then
        lblStatus.Caption="Connection Successful at Remote IP" & _
            sktTCPChatClient.RemoteHostIP
    End If
End Sub

Private Sub sktTCPChatClient _ DataArrival (ByVal bytesTotal As Long)
    Dim vtData As Variant

    sktTCPChatClient.GetData vtData, vbString
    txtReceived.Text=vtData
End Sub

Private Sub txtDataToSend _ Change ()
    sktTCPChatClient.SendData txtDataToSend.Text
End Sub
```

程序清单 1-4 所示的 cmd Close Connection _ Click () 事件过程关闭连接时，并不需要调用 Winsock 控件的 Close () 方法。相反，我们发送字符串“QUIT”，以便让服务器调用它的 Close () 方法来实际断开连接。建立连接和断开连接的顺序会影响 Winsock 应用程序的可用性和可靠性。在第 10 章的 Winsock Chat 应用程序中，我们将详细讨论这个问题。

客户端使用与服务器同样的 Winsock 方法来交换数据。调用 Winsock 控件的 SendData () 方法向服务器发送数据，在 DataArrival () 事件过程中调用 Winsock 控件的 GetData () 方法来获取服务器传过来的数据。

1.3 大量强有力的工具

在本章的快速进入 Visual Basic Internet 编程特性中，我们探讨了 Visual Basic 的两个有用工具，从而可以自己开发 ActiveX 组件，以便能够用友好的交互性和多媒体来扩展 Web。我们还发现可以用 Winsock ActiveX 控件书写 Internet 应用程序和服务，从而能通过 Internet 交换世界各地的信息。在以后的章节中，我们将探讨能增强我们 Internet 和 Web 经验的 Visual Basic 的其他功能。