

北京科海培训中心

Designed for
Microsoft®
Windows NT®
Windows 95

Visual Basic

脚本编写指南



(美) Scot Hillier 著
张威 王彬星 译
卢庆龄 审校

机械工业出版社

Microsoft

北京科海培训中心

Visual Basic 脚本编写指南

(美) Scot Hillier 著

张 威 王彬星 译

卢庆龄 审校

机械工业出版社

著作权合同登记号:图字 01-97-1831

内 容 提 要

本书详细介绍了使用 Microsoft Internet 技术,开发交互式 Web 页所需的各方面内容。内容包括:介绍 ActiveX Scripting 和 VBScript 语言;从使用 VBScript 实现自动 ActiveX 控件到使用 IDC 和 OLEISAPI 访问后台数据库,介绍 Internet Explorer 编程功能和 Internet Information Server 编程,以及介绍最新的 Internet 技术。

书中提供众多告诉你如何实现目标的实例,两个为你生成代码的向导可加速开发过程。

你想提高 Visual Basic 编程技能吗?你想建立迷人的 Web 页吗?欢迎你阅读《Visual Basic 脚本编写指南》。

图书在版编目(CIP)数据

Visual Basic 脚本编写指南/(美)伊利耶(Hillier,S.)著;

张威,王彬星译. —北京:机械工业出版社,1998.5

书名原文:Inside Visual Basic Scripting Edition

ISBN 7-111-06321-X

I. V… II. ①伊… ②张… ③王… III. Basic 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(98)第 08193 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:科培 责任校对:成昊

北京市朝阳区科普印刷厂印刷·新华书店北京发行所发行

1998 年 4 月第 1 版·1998 年 4 月第 1 次印刷

787mm×1092mm 1/16·21.375 印张·520 千字

0001—5000 册

定 价:36.00 元

前 言

和以前相比,今天有更多的人在编写代码,包括专业人员和非专业人员,并且每天都从 Internet 的开放性中得到新的信息。小商业主、非盈利组织和独立个人都在为获得信息及在 World Wide Web 上抛头露面而学习 HTML 及其他 Web 创作技术。随着 Microsoft 的 Visual Basic, Script Edition (Microsoft Visual Basic 脚本编写版) 的推出,一直在为全世界数以千计的商业提供定制的解决方案的 Visual Basic 用户群,现在可以在 Microsoft Internet Explorer 和 Microsoft Windows 平台的安全而健全的环境下将他们当前的知识应用于使用 ActiveX 技术创建灵活、强大、动态的 Web 页面中。

尽管在重要的应用程序开发中,Visual Basic 一直是,而且也将继续是一支主要的力量,但它并不将其成功归结于自己的技术优势。Visual Basic 的成功也和 Visual Basic 的开发群体有很大关系,他们志同道合,并热衷致力于用 Visual Basic 和其他的 Microsoft 开发工具创建独特而强大的应用程序。开发者们互相依靠,这样便于了解哪一项技术起作用(或不起作用),也便于借鉴其他“先行者”的经验教训。因为这种类型的反馈有助于 Microsoft 确定未来产品的方向,所以在数年来 Microsoft 得以和开发者团体紧密工作,以确保我们的产品和服务程序将继续提供成功的应用程序的组件。Microsoft 对有影响力的开发者团体以及支持信息共享并对开发者团体的自力更生品质有至关重要作用的团结推动的用户群、联机论坛、Web 页面、顾问委员会、独立简讯和讲座会有很大的义务。

世界范围内的开发者团体的核心是那些在他们自己的专业领域担任领导角色的个人。New Technology Solutions 公司的 Scot Hillier 就是这个团体中一个典型的领导者。在他的领域中,他是一个真正的专家——一个 Microsoft 赖以获得产品开发反馈的技术领头人,他正在和 Dan Mezick(也是 New Tech 的员工)一起创建世界最大的基于团体开发者的程序 Microsoft Developer Days。通过这个不断发展的与同等开发者共享最新信息的工程,Scot 和 Dan 一直在团体中奉献他们的时间、知识和精力。

借助于本书,你将得到的不只是一种和世界上最伟大的软件开发工具配合使用的脚本编写语言的手册资源,你也将得到一个新的合作伙伴,他位于,而且在未来的几年内也将继续处于世界范围的开发者团体的中心。如果你是这个团体的新成员,我向你致以热烈的欢迎。如果你早已加入这个团体,我相信你将发现你和 Scot 的合作会像 Microsoft 和他的合作一样富有成效。

序 言

本书是为那些需要着手编制 World Wide Web 的动态页面的开发者而写的。在本书的创作中,我的意图是站在一个试图从客户机/服务器的世界跳跃至 Intranet/Internet 世界的 Visual Basic 程序员的角度进行编写。

本书中我采用的方法源于我向数以千计的人讲授 Visual Basic 3.0 和 4.0 的经验。这些经验告诉我核心内容和有力的实例是最好的老师。抱着这样的看法,我主要致力于讲授 Web 开发中与技术相关的基本原则,以及提供众多告诉你如何实现你的目标的实例。

本书开始概述了可用于开发数据库解决方法的当前 Web 技术。随即是 Microsoft Visual Basic,Scripting Edition(即 VBScript 语言)的特点。如果你是一个 Visual Basic 开发者,你将对 VBScript 和 Microsoft Visual Basic for Applications(VBA)之间的重要差别感兴趣,书中对这些主要差别作了详细介绍。接下来是关于在 Web 页面中使用控件的讨论。这个讨论不仅包括可通过 HTML 使用的内部控件,而且还包括 ActiveX 组件(我们习惯称作 OCX)的使用。接着讲述将 Microsoft Internet Information Server(IIS,Microsoft Internet 信息服务器)用作一个后端组件。本书的最后给出了几个完整的数据发布项目的实例。

本书的写作过程中,我对 Internet 技术变化的步伐感到惊讶。当然,每个人都会谈论事物变化得多么快,但在本书写作过程中 Microsoft 推出的技术的数量确实让我感到惊愕。新特性不断被添加到 Microsoft Internet Explorer——这使得我为确保我的代码仍旧可以起作用而手忙脚乱。但是,随着本书的问世,它将达到帮助你开发一个完整的动态 Web 内容的目的。

本书底稿的写作只用了十周的时间,但本书从构思到面世的过程则相当长,因此当你读到这本书的时候,当前技术发生的变化已不可避免地会影响本书的内容。譬如,允许在服务器端使用 VBScript 代码的技术现在刚刚出现。服务器端的脚本编写预计允许在浏览器内和服务器上都使用 ActiveX 组件。实际上,服务器和客户机上的组件都将可以通过 Internet 通信。这种交互作用就是大家所知道的 Distributed COM(DCOM,分布式组件对象模型),它将再一次彻底改革 Internet 的开发。在本书的下版中,我一定像这样完整地论述出新的技术。

回顾所付出的全部艰辛,我必须说十周的写作周期是值得的。我当然感觉长日难捱,但我也有无穷的乐趣。Dan Mezik 确保了 my 日程安排一清二楚,这使得我除了处理技术问题以外什么都不必做,坦率地说,我认为我得到的是最好的工作。你在别的什么地方还能找到付给你报酬让你摸索最新的技术并且不必为日常琐事操劳的工作吗?

愿你欣赏本书,并祝你在开发探索中一路顺风。

目 录

第1章 Web 技术概述	(1)
1.1 Internet 概念的变化	(3)
1.2 Internet 开发工具	(3)
1.3 开发者的机遇	(7)
1.4 本书中的假定	(7)
1.5 HTML 概述	(7)
第2章 ActiveX Scripting 和 VBScript	(14)
2.1 脚本编写语言	(14)
2.1.1 JavaScript	(15)
2.1.2 VBScript	(17)
2.1.3 JavaScript 和 VBScript 的比较	(18)
2.1.4 语言可升档性	(19)
2.1.5 其他脚本编写语言	(19)
2.2 VBScript 语言	(20)
2.2.1 Microsoft Visual Basic for Applications 和 VBScript 之间的主要区别	(22)
2.3 窗体和模块	(27)
2.3.1 窗体	(27)
2.3.2 模块	(28)
第3章 理解 Automation 和浏览器结构	(30)
3.1 Internet Explorer 对象模型	(32)
3.2 使用 Internet Explorer 对象	(35)
3.3 Internet Explorer 脚本编写对象模型	(38)
3.3.1 Windows 对象	(38)
3.3.2 Document 对象	(40)
3.3.3 Form 对象	(41)
3.3.4 Location 对象	(42)
3.3.5 Link 对象	(43)
3.3.6 Anchor 对象	(44)
3.3.7 Elements 数组	(45)
3.3.8 其他对象	(45)
3.4 在 VBScript 中使用脚本编写模型	(45)
3.4.1 Windows 对象实例	(45)
3.4.2 Document 对象实例	(46)
3.4.3 Form 对象实例	(48)
3.5 内部 HTML 控件	(49)

3.5.1 内部 Button 控件	(51)
3.5.2 内部 CheckBox 控件	(52)
3.5.3 内部 Hidden 控件	(52)
3.5.4 内部 Password 控件	(53)
3.5.5 内部 Radio 控件	(53)
3.5.6 内部 Reset 控件	(54)
3.5.7 内部 Select 控件	(54)
3.5.8 内部 Submit 控件	(55)
3.5.9 内部 Text 控件	(55)
3.5.10 内部 TextArea 控件	(56)
第4章 处理 ActiveX 控件	(57)
4.1 获得 ActiveX 组件的 Class IDs	(60)
4.1.1 系统 Registry	(60)
4.1.2 ActiveX Control Lister	(62)
4.1.3 ActiveX Control Pad	(62)
4.2 ActiveX Control Pad 和 HTML Layout 控件	(64)
4.3 代码标记和 Internet 安全	(66)
4.4 本书配套光盘中的 ActiveX 控件	(67)
4.4.1 Label 控件	(67)
4.4.2 Timer 控件	(68)
4.4.3 New Item 控件	(69)
4.4.4 Stock Ticker 控件	(70)
4.4.5 Marquee 控件	(71)
4.5 扫雷和计算器实例	(72)
第5章 Microsoft Internet 信息服务器	(87)
5.1 Internet 服务 API	(87)
5.2 Internet 数据库连接器	(88)
5.2.1 安装 ODBC 资源	(90)
5.2.2 使用 IDC	(91)
5.3 详细的 IDC 实例	(99)
5.3.1 用户组事件登记实例	(99)
5.3.2 产品陈列实例	(102)
5.3.3 只有一个成员的页面实例	(108)
5.4 OLEISAPI	(111)
5.4.1 建立 OLEISAPI	(112)
5.4.2 在 Visual Basic 4.0 中建立供 OLEISAPI 使用的 ActiveX 组件	(112)
5.4.3 一个简单的 OLEISAPI 语法分析实例	(115)
5.5 一个详尽的 OLEISAPI 实例	(121)
5.5.1 OLEISAPI 扫雷游戏	(121)
5.6 IIS 向导	(131)
5.7 CGI 和 IIS	(138)

5.8 其他 IIS 服务	(138)
5.8.1 文件传输协议	(138)
5.8.2 Gopher	(139)
第6章 项目 1:Intranet 简讯	(140)
6.1 要求	(140)
6.2 设计目标	(140)
6.3 先决条件	(141)
6.4 创建数据源	(142)
6.4.1 安装 Access ODBC 驱动程序	(142)
6.4.2 创建 Access 数据库	(143)
6.4.3 定义数据源	(145)
6.5 创建 HTML 页面	(148)
6.5.1 首部	(148)
6.5.2 主体部分	(148)
6.5.3 脚本部分	(154)
6.6 调用 Internet Database Connector	(159)
6.6.1 idc 文件	(159)
6.6.2 htx 文件	(159)
6.7 发送简讯	(161)
6.7.1 建立界面	(161)
6.7.2 为应用程序编码	(163)
6.8 运行项目	(171)
第7章 项目 2:事件登记应用程序	(174)
7.1 要求	(174)
7.2 设计目标	(174)
7.3 先决条件	(175)
7.4 创建数据源	(175)
7.4.1 创建 Access 数据库	(176)
7.4.2 定义数据源	(177)
7.5 创建 HTML 页面	(179)
7.5.1 首部	(179)
7.5.2 主体部分	(180)
7.5.3 脚本部分	(186)
7.6 创建进程内 ActiveX 组件	(196)
7.6.1 Person 类	(196)
7.6.2 主模块	(208)
7.6.3 完成 ActiveX 组件	(208)
7.7 运行项目	(209)

第8章 项目 3:一个联机书店	(212)
8.1 要求	(212)
8.2 设计目标	(212)
8.3 先决条件	(213)
8.4 创建数据源	(213)
8.4.1 创建 Access 数据库	(214)
8.4.2 定义数据源	(215)
8.5 创建主页	(216)
8.6 建立搜索引擎	(228)
8.7 创建选定书名页面	(229)
8.8 图书订购查询	(235)
8.9 创建图书订购页面	(235)
8.10 购买查询	(244)
8.11 创建确认页面	(245)
8.12 运行项目	(249)
第9章 项目 4:一个用 ActiveX Control Pad 建立的交互式订购窗体	(252)
9.1 要求	(252)
9.2 先决条件	(252)
9.3 ActiveX Control Pad 的功能	(252)
9.4 HTML Source Editor	(253)
9.5 ActiveX Control Editor	(253)
9.5.1 实例 1:使用 ActiveX Control Editor	(255)
9.6 Script Wizard	(259)
9.6.1 实例 2:使用 Script Wizard	(259)
9.7 HTML Layout Control 和 HTML Layout Editor	(262)
9.7.1 实例 3A:使用 HTML Layout 控件	(264)
9.7.2 实例 3B:使用 HTML Layout Editor	(265)
9.7.3 运行应用程序	(270)
9.8 交互式订购窗体	(271)
9.8.1 bytecomp. htm 文件	(271)
9.8.2 bytecomp. alx 文件	(272)
9.8.3 添加 VBScript 代码	(277)
9.8.4 运行项目	(281)
第10章 崛起的 Internet 技术	(286)
10.1 Windows NT 4.0	(286)
10.2 Normandy	(286)
10.3 Microsoft SQL Server 6.5	(287)
10.4 Microsoft dbWeb	(287)
10.4.1 安装 dbWeb	(288)
10.4.2 dbWeb Administrator	(288)

10.4.3 dbWeb Service	(291)
10.5 未来展望	(291)

附录A HTML 入门 (294)

A.1 HTML 文档介绍	(294)
A.1.1 首部	(294)
A.1.2 脚本部分	(295)
A.1.3 主体部分	(296)
A.2 HTML 标记	(297)
A.2.1 锚(Anchor)标记	(297)
A.2.2 基本(Base)标记	(297)
A.2.3 BaseFont 标记	(298)
A.2.4 BGSound 标记	(298)
A.2.5 中断(Break)标记	(298)
A.2.6 注释(Comment)标记	(299)
A.2.7 Div 标记	(299)
A.2.8 字体(Font)标记	(300)
A.2.9 窗体(Form)标记	(300)
A.2.10 框架(Frame)标记	(301)
A.2.11 Frameset 标记	(302)
A.2.12 水平线(Horizontal Rule)标记	(302)
A.2.13 Iframe 标记	(303)
A.2.14 图像(Image)标记	(303)
A.2.15 输入(Input)标记	(305)
A.2.16 列表(List)标记	(306)
A.2.17 大帐幕(Marquee)标记	(307)
A.2.18 Noframes 标记	(308)
A.2.19 对象(Object)标记	(308)
A.2.20 Param 标记	(310)
A.2.21 段落(Paragraph)标记	(310)
A.2.22 样式(Style)标记	(311)
A.2.23 表(Table)标记	(311)
A.2.24 TextArea 标记	(314)
A.2.25 文本格式化标记	(315)
A.2.26 HTML 文档中的颜色	(315)

附录B 代码编写标准 (316)

B.1 HTML 标准	(316)
B.1.1 首部	(316)
B.1.2 脚本部分	(316)
B.1.3 主体部分	(317)
B.2 VBScript 标注	(317)
B.2.1 对象命名	(317)

B.2.2 变量命名	(318)
B.3 注释	(318)
B.4 大小写敏感	(319)
B.5 <!DOCTYPE>标记	(320)
附录 C Visual Basic 脚本编写版不支持 Visual Basic for Applications 的功能	(321)
附录D 交互式 Web 站点	(323)
D.1 和本书配合使用的站点	(323)
D.2 ActiveX 控件资源	(324)
D.3 其他站点	(325)
附录 E 词汇	(327)

第1章 Web 技术概述

在过去的几年中,我们开始相信 Internet 将渗透到现代生活的方方面面。作为软件开发者,我们知道 Internet 开发是下一次充满机会的浪潮,如果我们错过这一浪潮,就会跟不上时代的潮流。大量的对话听起来好像只是一场骗局。一个开发者——对任何人都一样——不会轻而易举地区分出事实和谎言。谎言固然丰富,但夸大其词的下面是这样的一个真理:Internet 对目前的软件开发产生了重大的影响。

Internet 是使用同样规则进行通信的数以千计的计算机构成的国际互联网络。它如此受欢迎的原因不单是有数据访问功能,而且它还有友好的浏览器界面。浏览器是这样简单的应用程序:它在计算机上运行,允许用户可在 Internet 上访问获得大量的信息。浏览器典型地用于查看 World Wide Web(WWW)上的 Web 页面。World Wide Web 是 Internet 上互联的超文本文档(Web 页面)集。典型的浏览器提供对数据的直观查看,即使用户是初学者也可以容易地理解。要在 WWW 上的 Web 页面之间漫游,只需单击显示在浏览器文本中的链接即可,图 1-1 显示了 Internet 浏览器 Microsoft Internet Explorer 的图形用户界面(GUI)。该浏览器正在显示 WWW 上的一张 Web 页面示例。

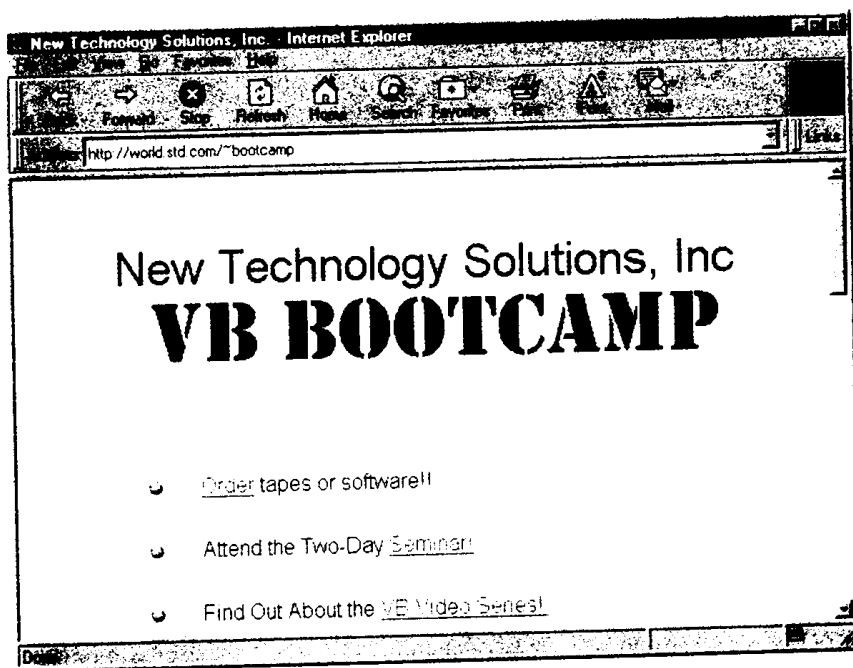


图 1-1 通过 Microsoft Internet Explorer(面向 Windows 95 的 3.0 版本)访问的 Web 页面

Internet 上的信息存放在服务器内。服务器是向连接的客户机提供服务的计算机。Web 服务器是和 WWW 相连接的计算机,它以 Web 页面的形式提供在浏览器内显示的信息。Web 上的每一个服务器用一个唯一的 Internet Protocol(IP)地址标识。要和某一 Web 服务器连接,浏览器必须指定这个服务器的 IP 地址。这个地址通常采用 Uniform Resource Loca-

tor (URL, 统一资源定位器) 的形式, URL 是标识服务器和要显示的特定数据的文本字符串。服务器和浏览器是组成 Internet 的基本组件。

连接到 Internet 上的计算机使用 protocol(协议) 进行通信。协议是计算机用于交换信息的一套规则。连接到 Internet 上的计算机采用 Transfer Control Protocol/Internet Protocol (TCP/IP) 进行通信, TCP/IP 允许数据以小的信息“包”的形式发送。Web 服务器和浏览器使用 Hypertext Transfer Protocol(HTTP, 超文本传输协议) 在 WWW 上交换信息。数据以基于文本的 Web 页面的形式从服务器传输, Web 页面通过浏览器阅读。这些基于文本的页面是用 Hypertext Markup Language(HTML, 超文本标记语言) 编写的。浏览器能够识别 HTML 并对其进行解码, 以便通过浏览器界面展示丰富的图形和文字。图 1-2 显示了在 Internet 上浏览器和服务器的相互连接的一个简单的示意图。

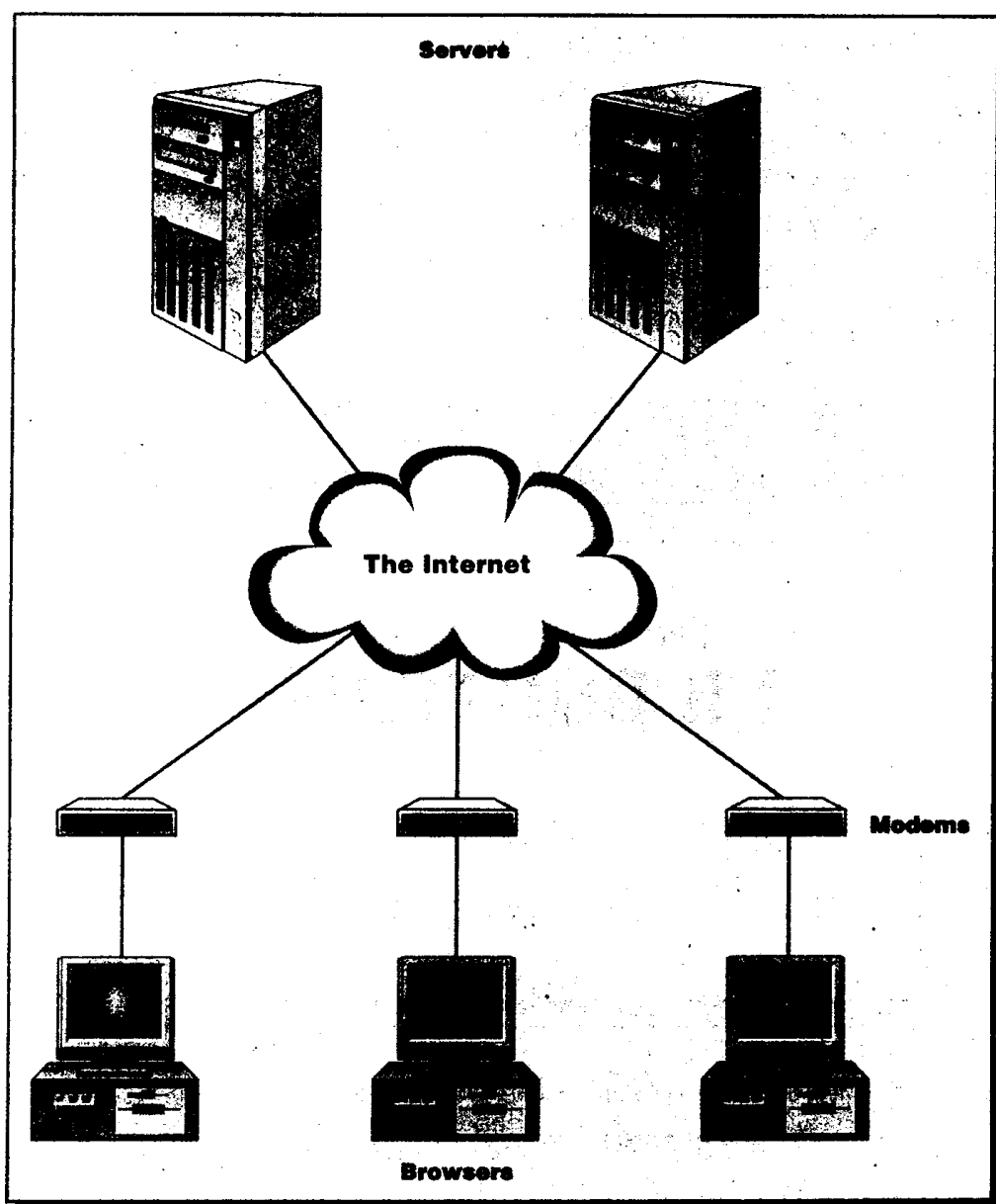


图 1-2 Internet 上的浏览器和服务器的相互连接示意图

1.1 Internet 概念的变化

虽然使用 Internet 会令人兴奋不已,但人们完全有权对它在未来的通信中扮演的角色提出质疑。Internet 会成为指导远距离商务的中心手段吗?还是一直作为专家管理的一个工具?Internet 商业在充分可行之前有许多障碍要克服。带宽、数据传输率在 Internet 上都有所限制。大多数家庭用户是通过诸如 CompuServe, America Online 或 Microsoft Network 这样的联机服务提供者连接到 WWW 上的。对服务提供者的访问是通过调制解调器完成的,调制解调器的操作速率不大于 28.8 位/秒。这样的速度导致浏览器响应很慢,使得用户有些心灰意冷,也使整个过程黯然失色。另外,Internet 上的事务并不是永远安全的,这使得用户对在 Web 上进行商务活动感到忐忑不安。

然而,在人们评价 Internet 技术的时候,他们经常忽视好的消息。尽管 Internet 上率先应用了服务器/浏览器模型,但同样的设置适用于支持应用在公共 WWW 上的 TCP/IP 协议的任何专用网络。这就意味着人们可以创建通过 TCP/IP 协议互连的计算机内部网络,利用服务器/浏览器技术可以向雇员提供同样直观的界面。这些内部网络称之为 intranets。intranet 可以克服 WWW 的传统局限,而又保持了受人欢迎的浏览器界面,用户可以利用这种界面查看数据。在 intranet 上,带宽通常不是一个问题,内部网络能以远远超过人们所能接受的速度通过浏览器向雇员提供安全的企业数据。同样雇员可以利用他们的交互式浏览器界面来放置订货、请求休假、公布公司信件等等。这种技术在许多公司里已经出现了。你从本书中学到的一切对 Internet 和 intranet 都是适用的。

关键概念: intranet 克服了 Internet 的两个主要的局限:带宽和安全问题。

1.2 Internet 开发工具

对开发者而言,利用 Internet/intranet 潮流的关键是软件工具。一个开发者需要的工具是建立在他以前的知识之上,并和现有技术可以很容易结合的软件工具。幸运的是,Microsoft Visual Basic, Scripting Edition (Microsoft Visual Basic 脚本编写版)(即 VBScript)就是这样的工具。VBScript 是 Microsoft Visual Basic for Applications 语言的一个极好的子集,Microsoft Visual Basic for Applications 在诸如 Microsoft Excel 的产品中得到了广泛的应用。VBScript 用于将静态 Web 页面转变为丰富、交互的环境,这种环境赏心悦目而且提供丰富的内容。VBScript 也可以通过扩展当前的 HTML 规范来和现有技术很好地结合。为 Internet 开发而学习 VBScript 是你编程技巧的一项投资,这种投资在几年之内都将得到回报。

Microsoft 已经创建了数量惊人的 Internet 开发工具,满足各种程度的开发者在客户端和服务端创建交互式应用程序的需要。你最初可能会认为仅仅使用 VBScript,你就不需要理解 Microsoft 的 Web 服务器技术,但你很快就会发现 Web 上没有一个开发者是孤立的。想创建优秀应用程序的开发者需要透彻理解 Internet 的全部工作部分——客户机和服务器。做为引言,让我们先概述一下 Microsoft 为创建 Web 服务器、浏览器应用程序和 Web 内容而提供的各种技术。

想完成 Web 服务器的开发者可使用从 Microsoft 获得技术。基于 Microsoft 软件的 Web

服务器的主要组件是操作系统 Windows NT 3.51 或更新版本。Windows NT 有能力行使 TCP/IP 网络服务器的作用,它是任何 Web 站点的基础。利用 Windows NT 服务器创建 TCP/IP 网络的能力当然不是什么新鲜事务,但是 Microsoft 随着 Microsoft Internet Information Server(IIS,Internet 信息服务器)的推出,又添加了一种新的重要的功能。IIS 安装在 Windows NT 服务器上,它提供允许已联网的用户访问 World Wide Web、传送文件和搜索 Web 的服务。图 1-3 显示了 IIS 的基本结构。

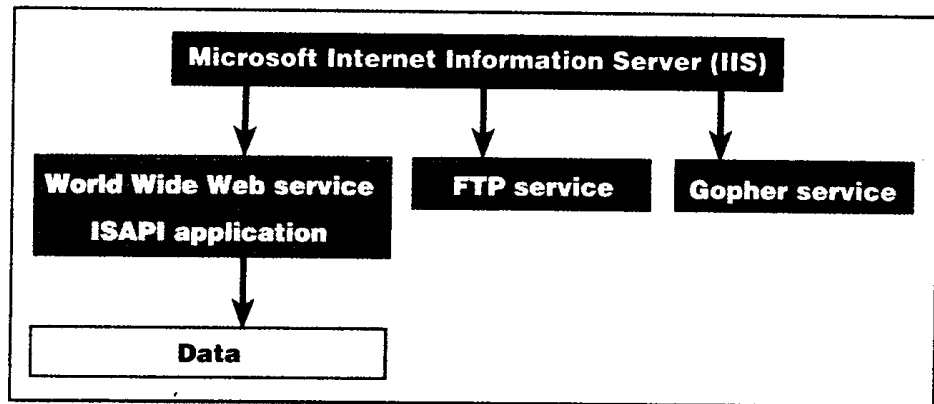


图 1-3 Microsoft Internet Information Server 的结构

在 IIS 伞型结构下,你会发现 WWW 服务、文件传送协议(FTP)服务和 Gopher 服务。这三个服务是 IIS 的核心。创建最简单的站点不需要掌握这些服务的知识。一旦 IIS 在 Windows NT 服务器上安装,你只要把 HTML 页面放置在文件夹\wwwroot 下,就会得到一个相当完善的 Web 站点。然而,完成这点工作需要下很多的功夫。IIS 提供两个强大的动态链接库(DLL):Internet Database Connector(IDC)和 OLE Internet Server Application Programming Interface(OLEISAPI)。这些 DLL 允许你方便地访问 IIS 主要支持数据库发布的最强大的功能。

当一个 Web 站点需要和服务器的一个可执行程序交互作用时,传统的途径一直是调用 Common Gateway Interface(CGI,公用网关接口)标准。例如,一个 CGI 程序在每一次浏览器和它连接时就可以运行,并向服务器传送一个数据请求,如查询。CGI 程序执行并将结果以 HTML 的形式返回到客户浏览器。然而,CGI 为浏览器发出的每一个请求都创建一个新的过程。CGI 对它自己以前的事例没有记忆,这样就导致耗费过多,响应时间缓慢。Microsoft 通过创建 Internet Server Application Programming Interface 在 IIS 中已经解决了这个问题。ISAPI 是允许程序员开发与 IIS 配合工作的 DLL 的一个接口。由于 ISAPI 应用程序是 DLL,它们装入内存并只被初始化一次。因为库已经驻留内存,浏览器对 DLL 的后续调用需要的开销较少。此外,由于 DLL 知道以前的调用,所以它可以保持整个会话过程的状态信息。这意味着服务器现在可以记忆浏览器的关键信息,提供灵活多变的编程能力和改进的性能。

一个特别的 ISAPI 应用程序是 Internet Database Connector(IDC)。IDC 是访问服务器上任意的 ODBC(开放数据库互连)数据源的一种简单机制。该技术包含三个文本文件。第一个文件是实际的 HTML 文件。例如,你可以从浏览器内的 HTML 文件启动数据库查询,方

法是单击到 Internet Database Connector 文件的链接,Internet Database Connector 文件以 idc 为扩展名。包含的第二个文件是 idc 文件,它含有诸如你想要访问的数据源和想要执行的结构化查询语言(SQL)语句等信息。服务器然后将 idc 文件映射到 IDC 的 httpodbc.dll 文件中。IDC 阅读 idc 文件中的信息,连接 ODBC 数据源,检索必要的数据并将其合并到 HTML 扩展文件中。包含的第三个文件是 HTML 扩展文件,它是将返回给浏览器的实际 HTML 文档的模板。最后,IDC 将合并后的 HTML 文档返回给服务器,服务器再将它返回给浏览器显示。图 1-4 显示了浏览器查询数据库并返回结果的数据流。

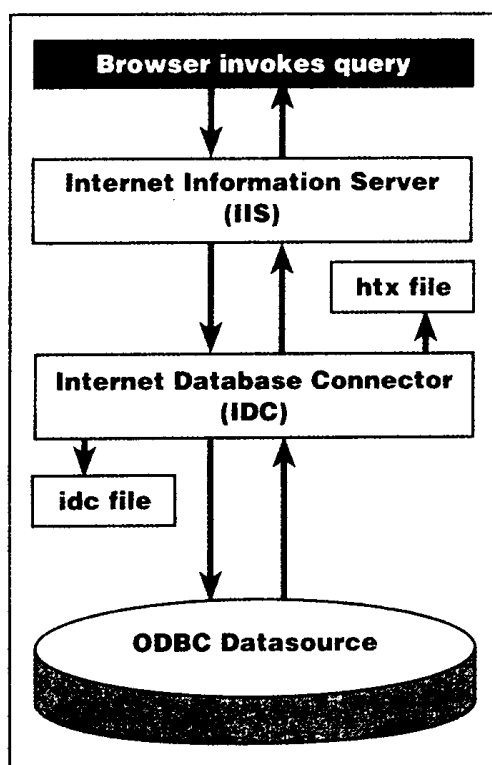


图 1-4 数据库查询的数据流

另一个 ISAPI 应用程序是 OLEISAPI,这个应用程序使 ISAPI 接口的函数功能可为任意的 ActiveX 组件所用。ActiveX 组件只不过是支持 Automation(自动化)的软件组件。Visual Basic 中的 OLE 服务器就是 ActiveX 组件的实例。你在 Visual Basic 4.0 和 Visual C++ 下都可以创建 ActiveX 组件。OLEISAPI 允许你访问你的进程内的 ActiveX 组件,这种 ActiveX 组件能够接收来自客户浏览器的请求,执行查询或其他附加功能,以及返回 HTML 文档。尽管 OLEISAPI 在功能上和 IDC 非常相似,但它给开发者更大的灵活性,这是由于它使一种开发工具(如 Visual Basic)的全部功能可为程序员所用。

在客户端,Internet Explorer 提供一个增强的环境,在这个增强的环境中,用户和 Web 信息交互作用。Internet Explorer 不只是识别标准的 HTML 标记,还识别管理 ActiveX 控件的扩展 HTML 标记。这些 ActiveX 控件(以前称作 OLE 控件)是在像 Visual Basic 这样的开发工具中使用的我们所熟悉的可重用 OLE 自定义(OCX)控件,在某些情况下,ActiveX 控件被优化以在 Internet 上传送,但它们确实只是 OCX 控件,实际上,Internet Explorer 可以支持任何 ActiveX 控件,使用这些 ActiveX 控件和编写 Visual Basic 代码一样简单。这是 VB-

Script 输入的图片方式。因为 VBScript 恰恰是 Microsoft Visual Basic for Applications 语言的一个子集,所以编写代码和使用 ActiveX 控件的语法格式和 Visual Basic 应用程序的语法格式类似。这是一个很重要的概念,因为 Visual Basic 程序员将发现 VBScript 是他们以前编程技巧的一个自然延伸。

在一张 HTML 页面中可以包含对 ActiveX 控件的引用,同样也可以包含对其他项(如图像)的引用。当包含 ActiveX 控件的 HTML 页面被 Internet Explorer 访问时,控件下载到客户机。Internet Explorer 登记下载的 ActiveX 组件并显示该页面。组件一旦下载后,就可在客户机上获得,并且不必再次下载。

除了管理 ActiveX 控件外,Internet Explorer 浏览器还有它自己的对象模型,这意味着使用 Visual Basic,通过 Automation 可以直接访问它。这就允许 Visual Basic 应用程序将 Internet Explorer 的功能作为一个组件包含。Internet Explorer 和任何其他的 Automation 组件一样,显示在 Visual Basic 环境的 References 对话框中。

Automation 也被 Internet Explorer 用来支持一项称作 ActiveX 文档的技术,ActiveX 文档是 OLE 文档的扩展。支持 ActiveX 文档使得 Internet Explorer 能够在浏览器窗口内打开一个应用程序,如 Microsoft Excel。对用户来说,这看起来就像在 Internet Explorer 内打开了一个独立的带有完整的工具栏和菜单的应用程序。

然后用户可以从浏览器上直接和应用程序进行交互。言外之意当然是用户永远不必离开浏览器。本地文件、网络文件和 Internet 全都可以在 Internet Explorer 内查看。对在桌面发生的全部工作,不论信息源如何,这种功能都可以提供一种统一的外观和感觉。统一桌面是 Internet 技术的一大进展。

也不要忘记 Visual Basic。Visual Basic 本身在你所有的 Internet 开发策略中扮演一个重要的角色。它通过一个简单的像进程内 DLL 一样运行的 ActiveX 接口和 Internet Explorer 直接对话。Visual Basic 同样也可以利用 ActiveX 控件,这些 ActiveX 控件通过 WinSock 层和 TCP/IP 协议允许直接访问 Internet,完全不需要任何类型的浏览器。Microsoft 在一个软件包中提供这些 ActiveX 控件,这个软件包就是 Internet Control Pack(ICP)。例如,ICP 包含一个为任何 Visual Basic 应用程序提供 Internet 浏览功能的 ActiveX 控件。把它从工具箱中取出,放入你的窗体,然后即可立刻浏览。

当涉及到内容时,你得到的新玩具更多。如果你使用的是 Microsoft Office 产品,如 Microsoft Word 和 Microsoft PowerPoint,你就会想马上研究 Internet Assistants for Office。这些有用的附件可帮助你直接用喜爱的 Office 工具创建 Internet 内容。生成 Web 的一张主页和编写一个新 Word 文档一样容易。这也是学习 HTML 的一个极好的方法,因为你可以生成页面,然后在一个文件编辑器(如 Notepad)中查看他们一切骇人的细节。

如果你想要的是一个高起点的工具,那么考虑 Microsoft FrontPage。FrontPage 是采用简单易用的界面创建和维护 HTML 页面的 Web 创作工具,该系统的妙处在于,举例来说,如果你在一张 HTML 页面上改变了背景,FrontPage 可以改变所有相联系的页面。此外,你还可得到帮你创建页面的一套向导和一些人们称之为 WebBots 的自动功能,WebBots 充当 CGI 脚本的替代品,这样使得非程序员也能够访问高级功能。

最后,如果你愿意,可以在任何文本编辑器下创建你的 HTML 页面。(NotePad 好像比较受欢迎)毕竟 HTML 只不过是文字。