

Microsoft 程序设计系列

Microsoft® Press

Designed for
Microsoft®
Windows NT®
Windows 98



CD-ROM
Included

Microsoft®

Internet Explorer 5

程序设计

[美] Scott Roberts 著
北京博彦科技发展有限公司 译

用Internet Explorer 5
快速创建
Internet和在线商务
应用程序



清华大学出版社
<http://www.tup.tsinghua.edu.cn>

Microsoft Internet Explorer 5 程序设计

[美] Scott Roberts 著

北京博彦科技发展有限公司 译

清华大学出版社

(京)新登字 158 号

Microsoft Internet Explorer 5 程序设计
Programming Microsoft Internet Explorer 5
Scott Roberts

Copyright © 1999 by Microsoft Corporation.

Original English Language Edition Copyright © 1999 by Microsoft Corporation.
Published by arrangement with the original publisher, Microsoft Press,
a division of Microsoft Corporation, Redmond, Washington, U.S.A.

本书中文版由 Microsoft Press 授权清华大学出版社出版。

北京市版权局著作权合同登记章 图字 01-2000-2240 号

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: Microsoft Internet Explorer 5 程序设计
译 者: 北京博彦科技发展有限公司
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: 刘映欣
印 刷 者: 清华大学印刷厂
发 行 者: 新华书店总店北京科技发行所
开 本: 787×960 1/16 印张: 29.5 字数: 643 千字
版 次: 2001 年 2 月第 1 版 2001 年 2 月第 1 次印刷
书 号: ISBN 7-900630-80-5
印 数: 0001 ~ 4000
定 价: 59.00 元

前 言

我刚加入 Microsoft 时,有幸被聘为 Internet 客户端应用程序开发支持小组中的一名技术支持工程师。令我感到幸运的是自己能有机会与那些为当今最杰出的 Web 浏览器(Microsoft Internet Explorer)创建 Internet 应用程序的客户在一起工作。在我从事技术支持的工作过程中,看到了那些客户使用 Microsoft 的产品和技术创建的第一流的应用程序。他们经常与我们联系,查询一些如何使用 Internet Explorer 在他们的应用程序中实现多种不同特性的信息,这些特性包括 DHTML 脚本编程和 WebBrowser 控件。随着时间的流逝,我也学到了许多有关编写 Internet Explorer 应用程序和使用 Internet Explorer 组件的知识,每天我都用这些知识帮助客户创建尽可能好的应用程序。但是,工作中总有一个问题困扰着我,那就是我只能在客户向开发支持提出问题时才能帮助他们。

我花了许多时间和精力寻找一本可以推荐给客户的较全面介绍 Internet Explorer 程序设计的书,可是始终没有找到。因此我决定自己写一本书,这本书以我在作为技术支持工程师阶段学到的知识以及转移到从事 Internet Explorer 开发工作以来所掌握的知识为基础,由我自己的多年经验和那些为本书提供建议、例程、注解等同仁的经验所组成。我希望这本书无论在现在还是将来都能对您创建 Internet Explorer 应用程序有所帮助。

在编写本书的整个过程中,有许多人以不同的方式给予了支持,如解答我的问题、提供例程等等。我想首先感谢整个 Microsoft Internet Explorer 开发工作组,他们创建了目前为止市面上最好的浏览器。没有这个杰出的产品,浏览 Web 站点就不会如此有趣,本书也就不会存在了。

其次我要特别感谢我的朋友 Nick Dallett,他从前是 Internet Client Developer Support Team 中的一员,现在属于 Microsoft Windows Update 工作组。Nick 花了大量的时间帮助我从技术方面对每章和每个例程进行检查,而且他还提供了许多自己的例程。Nick 用了许多不眠之夜和周末为本书创建了诸多例程: DHTML Ledger、DHTML Memo、DHTML Soup、DHTML Ledger Behavior、Ledger HTA、Memo HTA 和 Page Controller。有了这些例程,编写本书就快多了。

我还要感谢 Internet 客户端应用程序开发支持小组中的下列同事: Mark Harper,他检查了第 5 章中有关 Code Download 部分的内容,并且他的注释起到了很好的帮助作用; Jason Strayer,他解答了一些非常难的 Internet Explorer、DHTML

Object Model 和 Visual C++ 方面的问题;Mark Davis,他提供了第 11 章中的 AtlBehave 例程,并回答了一些有关 OC96 规范、DHTML Object Model 和 Visual C++ 的问题;Jess Judge,他提供了第 11 章中的 ClientCaps 和 LinkBuilder 例程,并回答了一些常见的有关 Client Capabilities 和 Persistence Behaviors 方面的问题;Dave Templin,他调试了我的 WebBand 例程,并为一些疑难问题的解答提供了建议;Heidi Housten,他为第 11 章提供了 TOC Behavior 例程;Rene Schuchter,他回答了一般的 Code Download 方面的问题;Sharon Malmquist,他总是督促我做一些更有意义更好的事情,如果没有 Sharon,我可能就没有足够的动力写这本书了。我还要感谢 Internet 客户端应用程序开发支持小组这个集体,他们投入了非常多的时间帮助客户理解如何为 Microsoft Internet Explorer 开发应用程序,并一直坚持从事这一伟大的工作。

对于 Internet Explorer SDK 文献工作组,我要感谢 Jerry Drain 和 Dennis McCreery,他们帮助解决了与 Internet Explorer 有关的头文件和库的问题。另外,我要感谢 Matt Oshry,他帮我解决了许多与文献有关的问题,Matt 曾无数次地主动帮助我。我只想说整个 MSDN Online Web Workshop 工作组提供了我所知的最好的文献产品。

对于 Internet Explorer 的开发者,我想感谢 Dave Massy,他是 DHTML Behaviors 的 Program Manager,他总是主动来帮助我,Dave 协助我检查第 11 章并介绍了一些 IDocHostUIHandler 中新添加的内容,对此我深表感谢。我要感谢 Phil Cooper,他检查了第 4 章并回答了许多有关 HTML Applications 和 DHTML 对象模型的问题,正是他创建了 HTAs。谢谢您,Phil,感谢您聘用我作为 Trident Extensibility 工作组中的一名开发者。一直以来我对这份工作都非常有兴趣,我希望永远都会继续下去。

我还想感谢以下人员,他们的工作帮助我提高了本书的质量:Microsoft Museum 的 April Hill,他为第 1 章提供了有关 Internet Explorer 上市的日期;Eric Berman 回答了有关 Web Accessories 的问题;Sundaram Ramani 提供了他的 HTML-Spy HTA 例程;Strohm Armstrong 为 Explorer Bars 和 Desk 区做了大量的工作;Alan Auerbach 回答了有关 WebBrowser 控件的问题和事件,Alan 是另一位主动帮助我的人。

对于 Microsoft Press,我想感谢 Eric Stroo,他帮助我建立起此项目的工作;Victoria Thulman,本书的项目编辑,在书的结构和内容上给予我很大的帮助;Julie Xiao,本书的技术编辑,对所有技术细节和规则帮我进行了一校和二校。另外,还要感谢 Joshua Trupin 和 Joanne Steinhart,感谢他们在 *MIND* 杂志上发表了本书的部分内容。

最后,我要感谢我的妻子 Andrea。我想把本书献给她,以表达我对她不尽的感激之情。在我写书的这几个月中,她从未抱怨过一句,她总是支持我所做的每项工作。她的支持减轻了我一天天的工作压力,使我能够提前两个月完成本书的编写工作。谢谢你,谢谢你总是和我在一起,爱着我和我们的孩子,我要用意大利语再说一次:Ti amo moltissimo。

目 录

第 1 部分 使用 HTML 和 DHTML

第 1 章 Internet Explorer 的发展	1
1.1 Internet Explorer 版本 1 和版本 2	2
1.2 Internet Explorer 版本 3	3
1.2.1 ActiveX 控件	3
1.2.2 脚本编程支持功能	4
1.2.3 Internet Explorer 对象模型	4
1.2.4 WebBrowser 控件	4
1.3 Internet Explorer 版本 4	5
1.3.1 动态 HTML	5
1.3.2 数据捆绑	6
1.3.3 增强型 WebBrowser 控件	6
1.3.4 Explorer Bars	6
1.3.5 浏览器助手对象	8
1.4 Internet Explorer 版本 5	8
1.4.1 DHTML Behaviors 及 HTML Components(HTCs)	10
1.4.2 可持久性	10
1.4.3 客户机功能	10
1.4.4 HTML Application	11
1.4.5 打印	11
1.4.6 动态属性	11
1.4.7 XML	11
1.4.8 增强的运行效率	12
1.4.9 使用 Internet Explorer 5 编程的原因	12
第 2 章 HTML 与脚本制作	15
2.1 HTML 简介	16
2.1.1 标记	16

2.1.2	标题	18
2.1.3	文本格式化	19
2.1.4	超级链接	23
2.1.5	图像	23
2.1.6	有序和无序的列表	24
2.1.7	选项框	26
2.1.8	表格	27
2.1.9	内部控件	28
2.1.10	注释	30
2.1.11	框架	30
2.1.12	HTML 表单	32
2.2	脚本编程基本内容	33
2.2.1	给网页添加脚本	34
2.2.2	程序语句	35
2.2.3	注释	35
2.2.4	变量	36
2.2.5	数据类型	39
2.2.6	数组	42
2.2.7	运算符	43
2.2.8	表达式	47
2.2.9	利用条件程序语言控制程序流	48
2.2.10	循环	52
2.2.11	子程序和函数	56
2.3	创建简单的脚本应用程序	61
第 3 章	用 DHTML 进行程序设计	65
3.1	DHTML 对象模型	66
3.1.1	对象	66
3.2	DHTML 对象	71
3.2.1	window 对象	72
3.2.2	document 对象	81
3.2.3	navigator 对象	88
3.2.4	event 对象	90
3.2.5	form 对象	93
3.2.6	对 DHTML 而言其他一些重要的 HTML 元素	94

3.3 其他优秀的 DHTML 特性	97
3.3.1 级联样式表	97
3.3.2 动态属性	100
3.3.3 打印支持	100
3.3.4 鼠标捕获	102
3.3.5 上下文菜单	104
3.4 举例: DHTML Memo 和 DHTML Ledger 程序	104
第 4 章 编写 HTML 应用程序	109
4.1 创建 HTA	110
4.1.1 < HTA: APPLICATION > 标记	111
4.2 创建带有客户区域的 HTA	117
4.3 其他 HTA 例子	120
 第 2 部分 创建 HTML 应用程序	
第 5 章 创建 ActiveX 控件	123
5.1 ActiveX 控件是什么?	124
5.1.1 在 Web 网页中插入 ActiveX 控件	125
5.1.2 给客户机下载 ActiveX 控件	128
5.1.3 在 Web 上获得安全性保障	137
5.2 创建 Visual Basic ActiveX 控件	142
5.2.1 添加方法	146
5.2.2 添加属性	147
5.2.3 激活事件	147
5.2.4 设置线程模型	148
5.2.5 编译	149
5.2.6 使用 PDW 组装 Visual Basic 控件	150
5.2.7 测试	154
5.2.8 为 Visual Basic 控件编写脚本	155
5.2.9 在 HTML 中进行初始化	157
5.2.10 打印	158
5.3 创建 ATL ActiveX 控件	159
5.3.1 添加方法	166
5.3.2 添加属性	168

5.3.3	激活事件	169
5.3.4	ATL 控件的加密	172
5.3.5	从 HTML 中初始化	173
5.3.6	打印	176
5.3.7	为 ATL 控件编写脚本	178
第 6 章	掌管 WebBrowser 控件和自动化 Internet Explorer	181
6.1	Internet Explorer 体系结构	183
6.1.1	WebBrowser 控件宿主	183
6.1.2	SHDOCVW	184
6.1.3	MSHTML	184
6.2	WebBrowser 控件和 Internet Explorer	184
6.2.1	接口	186
6.3	掌管 WebBrowser 控件	200
6.3.1	使用 Visual Basic	200
6.3.2	使用 Visual C++	206
6.4	自动化 Internet Explorer	219
6.4.1	使用 Visual Basic	219
6.4.2	使用 Visual C++ 和 COM APIs	225
6.5	添加高级功能	235
6.5.1	从 ActiveX 控件中访问 Internet Explorer 的 IWebBrowser2 接口	235
6.5.2	掌管 WebBrowser 控件时访问框架的 IWebBrowser2 接口	239
6.5.3	调用 Find、View Source 以及 Internet Options	241
6.6	发布 WebBrowser 控件	243
第 7 章	Internet Explorer 事件	245
7.1	事件和对外接口	245
7.1.1	IConnectionPointContainer	246
7.1.2	IConnectionPoint	247
7.2	捕获事件的方法	248
7.2.1	在 Visual Basic 中捕获事件	248
7.2.2	在 C++ 中捕获事件	251
7.2.3	在 ATL 中捕获事件	255
7.2.4	在 MFC 中捕获事件	262
7.3	由 Internet Explorer 5 激发的事件	268

7.3.1	BeforeNavigate2	271
7.3.2	CommandStateChange	273
7.3.3	DocumentComplete	275
7.3.4	DownloadBegin	278
7.3.5	DownloadComplete	278
7.3.6	NavigateComplete2	278
7.3.7	NewWindow2	279
7.3.8	ProgressChange	282
7.4	事件顺序	282
7.5	仅由 Internet Explorer 使用的事件	283
7.6	从 ActiveX 控件中处理 Internet Explorer 事件	284
第 8 章	访问 DHTML 对象模型	291
8.1	复习 DHTML 对象模型	291
8.2	从 Visual Basic 中访问 DHTML 对象模型	292
8.2.1	从 Web 网页中获得数据	296
8.2.2	使用 Visual Basic 向 Web 网页中动态地插入数据	302
8.3	从 Visual C++ 中访问 DHTML 对象模型	305
8.3.1	从 Web 网页上获得数据	310
8.3.2	使用 VC++ 向 Web 网页中动态地插入数据	318
8.4	处理 DHTML Object 事件	322
8.4.1	在 Visual Basic 中捕获 DHTML Objects 事件	322
8.4.2	在 Visual C++ 中为 DHTML Objects 进行事件捕获	324
8.5	从 Visual Basic Activex 控件中访问 DHTML 对象模型	329
第 9 章	高级掌管接口	333
9.1	IDocHostUIHandler	334
9.1.1	GetHostInfo	335
9.1.2	ShowContextMenu	338
9.1.3	TranslateAccelerator	340
9.1.4	扩展 DHTML 对象模型	343
9.1.5	AtlBrowser	345
9.2	ICustomDoc	348
9.3	WBCustomizer 对象	349
9.4	IDocHostShowUI	351

9.5 控制下载和执行的内容	353
第 10 章 Web Accessories	357
10.1 理解 Explorer Bar 和 Desk 区	358
10.1.1 Explorer Bands	358
10.1.2 Comm 区	360
10.1.3 Desk 区	361
10.2 使用 DHTML 和脚本创建 Explorer Bars	362
10.3 使用 C++ 和 COM 创建 Explorer Bars 和 Desk Bands	366
10.3.1 实现所需的 DLL 函数	367
10.3.2 实现需求接口	373
10.3.3 可选接口的实现	380
10.3.4 给容器发送命令	383
10.3.5 为区对象实现 Windows 程序	384
10.3.6 在区对象中掌管 WebBrowser 控件	386
10.4 添加工具栏按钮	389
10.4.1 显示 Explorer Bars	391
10.4.2 运行 COM 对象	392
10.4.3 运行脚本	393
10.4.4 运行可执行的文件	393
10.5 添加主菜单项	394
10.6 添加上下文菜单项	394
第 11 章 DHTML Behaviors 和 HTML 组件	397
11.1 使用默认 Behaviors	398
11.1.1 客户端功能	399
11.1.2 持久性	403
11.2 使用由 HTML Components 创建的 Behaviors	407
11.2.1 将 Behaviors 添加到 HTML 元素中	407
11.2.2 使用 HTC 在脚本中创建 Behaviors	409
11.2.3 访问包容文档的 DHTML 对象模型	414
11.2.4 HTC 的缺点	417
11.3 使用 Binary DHTML Behaviors	418
11.3.1 用 C++ 实现 Binary DHTML Behaviors	419
11.4 使用其他的 Behavior 样例程序	426

第 12 章 Browser Helper Object	429
12.1 Internet Explorer 如何启动 BHO	430
12.2 创建 BHO	431
12.2.1 IEHelper	431
12.2.2 实现 IObjectWithSite	433
12.3 用 BHO 处理 Internet Explorer 事件	435
12.4 使用 BHO 来控制 Internet Explorer	440
12.5 注册和取消注册 BHO	440
附录 疑难解答和技术支持	443
击键问题	443
MFC	443
ATL 和标准 C++	446
Win32 SDK 模态对话框	448
区对象	450
技术支持资源	453
新闻组	453
文献	453
Microsoft Knowledge Base	453
Microsoft Developer Support	457

第 1 部分 使用 HTML 和 DHTML

第 1 章

Internet Explorer 的发展

从计算机时代开始,人们就一直在寻找各种方法,力求简化计算机的应用。最初,技术人员基于文本的终端设备,在用户及其计算程序之间提供了主要链接。尽管程序开发者努力简化这种接口,但它仍然仅仅是计算机黑客(那些迷恋纯技术的人)乐于使用的工具,他们从中找到了用计算机工作的乐趣。直到出现了诸如 Microsoft Windows 这样的击键式操作系统,用户才能够以富有成效的、直观的方式与终端进行交互。鼠标和图形用户接口(GUI, graphical user interface)从根本上改变了人们对计算机操作的看法,使计算机得到成千上万人的接受和喜爱。

随着 Internet 和 World Wide Web 的流行,计算机的实用性跃上了新的台阶。通过一系列超级链接,用户可以从一个信息页(Web 页)漫游到另一个页面。而浏览这些网页时存在的唯一问题就是位于其上的信息是静态的,用户不能用在 Microsoft Windows 应用程序上所使用的方式与网页信息进行交互。为满足用户的交互需要,HTML 2.0(Hypertext Markup Language 2.0,超文本标记语言 2.0 版)中引进了表单,使用 HTML 表单,用户即可将信息发送给网页的拥有者。Web 上信息双向传输的实现,为 Internet 应用程序的实现铺平了道路。

随着时间的推移,计算机操作日益简化,但这种简化也增加了内在的复杂性,同时为满足用户需要,计算机程序也设计得越来越复杂。因此,所需要的逻辑结构、图形表现、子程序的堆积也都越来越复杂,这是对程序设计者和程序开发者的挑战。而 Internet 应用程序的开发却没有追随这样的趋势,当操作任务对用户而言变得简单的同时,对程序设计者和程序开发者而言任务也变得简单了。

1.1 Internet Explorer 版本 1 和版本 2

Microsoft Internet Explorer 在许多方面促成了这种进步。版本 1(如图 1-1 所示),在 1995 年 8 月 24 日作为 Windows 95 的组成部分面市,它简化了 Internet 应用程序的创建。

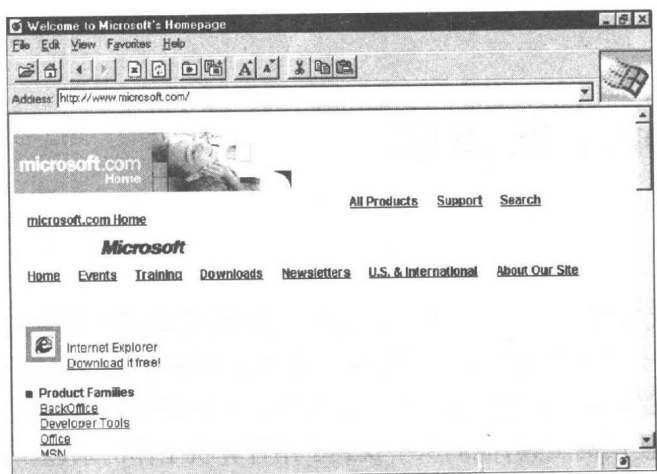


图 1-1 Internet Explorer 版本 1

使用 Internet Explorer 版本 1 和版本 2(1995 年 11 月 17 日面市,如图 1-2 所示),及使用标准的 HTML,可以创建基于 Web 的应用程序。因此,仅仅使用由 HTML 所创建的 Web 网页,Internet Explorer 就可以成为在世界范围内传播信息的工具。

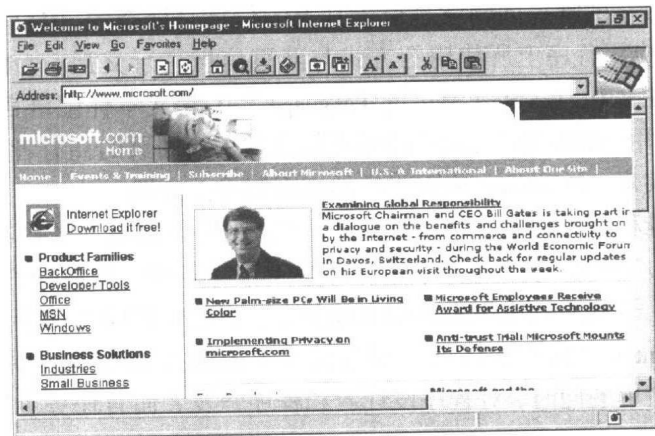


图 1-2 Internet Explorer 版本 2

Web 网页用户也可通过 HTML 表单与您进行交互。Internet Explorer 提供了一种创建基于 Web 应用程序的途径,该应用程序容许客户不必给您的公司打电话就能实现诸如“在线定货”之类的任务。这样的应用程序,不仅为您的公司节省了大量资金,也为客户节省了大量的时间,同时减少了他们的抱怨:信号不再忙碌,不必再排长队等候了。

1.2 Internet Explorer 版本 3

现在您也许会想:“既然 Internet Explorer 版本 1 和版本 2 带来了如此多的好处,Microsoft 又何必必要走得更远呢?”也许您能猜到答案:您所使用的 Windows 应用程序可以比用基本的 HTML 页做更多的工作。例如,当创建 Windows 应用程序时,您可以拥有任意数目的 ActiveX 控件,Internet Explorer 版本 1 和版本 2 甚至不具备掌管 ActiveX 控件的能力,于是 Internet Explorer 版本 3 登上了舞台。Internet Explorer 版本 3(如图 1-3 所示)于 1996 年 8 月 13 日发布,它的出现进一步推动了用 Internet Explorer 为 Web 创建功能完善的应用程序。版本 3 引进的一些新特性有助于实现 Internet Explorer 的可编程性和可重用性。



图 1-3 Internet Explorer 版本 3

1.2.1 ActiveX 控件

Internet Explorer 版本 3 最重要的特性之一,就是它在 Web 页上掌管 ActiveX 控件的能力。ActiveX 控件是通过组件对象模型(COM, Component Object Model)提供了某些功能性的动态链接库(DLLs, dynamic-link libraries)。使用 ActiveX 控件能够在自己的 Web 网页上做任何与在普通 Windows 应用程序上相同的工作,

然而,这些新特性也带来了安全性方面的风险。因此,版本3引入了安全性模型的概念,以防止人们创建使计算机瘫痪的 ActiveX 控件。我们将在第5章讨论创建 ActiveX 控件的方法。

1.2.2 脚本编程支持功能

在过去,系统不得不盲目地将数据传输给服务器,由服务器上运行的程序来检验用户输入的数据,如果某些数据错误或遗失,系统将重新显示带有问题解释的 Web 网页。如果用户不得不经常输入数据并重新提交表单,以确保所有信息都正确无误,就会给用户带来很大的麻烦,而且也很浪费时间,尤其当用户通过迟缓的调制解调器线路来连接 Internet 时。

Internet Explorer 版本3开始使用 Microsoft JScript 和 Microsoft Visual Basic Script (VBScript),同时引入了脚本编程支持功能。现在,您可以在 Web 网页所包含的 ActiveX 控件上调用方法和访问属性;而且,在将信息提交给 Internet 服务器进行处理之前,用户可在表单中验证数据的正确性。如某项数据出错,用户就会立即察觉并输入正确数据。

1.2.3 Internet Explorer 对象模型

前面已经提到,使用版本1和版本2时,Web 网页上显示的全部信息都是静态的:设计页面时,必须确定该页面上将包含何种信息;而一旦 Internet Explorer 显示页面后,就无法访问更改其上的信息了。

当页面加载时,使用脚本编程支持以及 Internet Explorer 对象模型(后来命名为 DHTML Object Model),可以决定页面上所要显示的信息。另外,还可以通过 DHTML 对象(Dynamic HTML,动态 HTML),从 Web 网页的脚本上访问信息,这样就能实现给服务器发送信息之前先验证表单数据的正确性。

1.2.4 WebBrowser 控件

到目前为止,我们仅考察了 Internet Explorer 的可编程性。那么,如何在自己的 Windows 应用程序中引入 Internet 的功能呢?例如,您也许想为学校的孩子们创建他们自己的 Web 浏览器;您也许想使用这种类型的 Web 浏览器来控制孩子们可查看的 Web 网页;或者想使自己的 Web 浏览器非常便于孩子们使用。当然,您可能不想重新创建 Web 浏览器,因为如果要那样做,您必须是精通多种技术的专家,包括通信、COM、C++、Internet 标准和协议以及安全性等等。

市场上有许多种 Web 浏览器,难道就没有简便的方法重新使用某些已有的功能吗?其实从 Internet Explorer 版本3开始就有了。Internet Explorer 版本3是