

Designed for



Microsoft®
Windows NT®
Windows 98

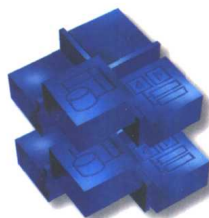


附 赠
CD-ROM

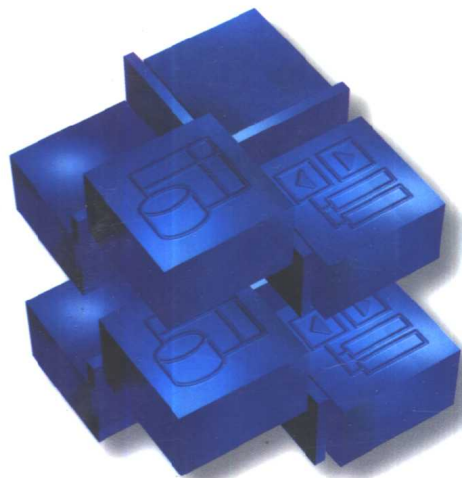
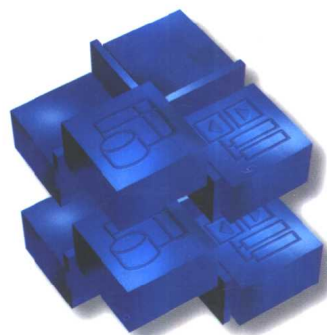
Microsoft Windows Script Host 2.0 Developer's Guide

微 软 公 司
核心技术书库

(德) Günter Born 著 马朝晖 等译



Windows Script Host 2



开发人员指南



机械工业出版社
China Machine Press

Microsoft Press

微软公司核心技术书库

Windows Script Host 2

开发人员指南

(德) Günter Born 著

马朝晖 等译



机械工业出版社
China Machine Press

本书全面介绍Microsoft Windows Script Host(WSH) 2的基本概念和高级应用。主要内容包括: WSH简介、开发工具、对象编程、交互式脚本、高级特性、编程技巧等。本书内容丰富、深入浅出, 对理解和掌握WSH、深入开发Windows应用, 提供了有效的参考。本书 附带光盘包含: WSH 2简单脚本定制工具、WSH对象参考等内容。

Günter Born: Microsoft Windows Script Host 2.0 Developer's Guide.

Copyright © 2000 by Günter Born.

Original English language edition copyright © 2001 by Microsoft Corporation; Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A. All rights reserved.

本书中文简体字版由美国微软出版社授权机械工业出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2000-3121

图书在版编目(CIP)数据

Windows Script Host 2开发人员指南/(德)保尔(Born, G.)著; 马朝晖等译. - 北京: 机械工业出版社, 2001. 6

(微软公司核心技术书库)

书名原文: Microsoft Windows Script Host 2.0 Developer's Guide

ISBN 7-111-08880-8

I. W… II. ①保… ②马… III. JAVA语言-程序设计 IV TP312

中国版本图书馆CIP数据核字(2001)第19375号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 张鸿斌

北京市密云县印刷厂印刷·新华书店北京发行所发行

2001年6月第1版第1次印刷

787mm × 1092mm 1/16 · 24.75印张

印数: 0 001-5 000册

定价: 55.00元(附光盘)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

译者序

从Windows 98开始，Windows操作系统中有了一个新工具，称为Windows Scripting Host (WSH)。到了版本2，Windows Scripting Host改名为Windows Script Host，它是一种操作系统级的主机脚本语言。它提供了两种脚本语言：VBScript和JScript。利用WSH，系统管理员可以实现自动的解决方案以管理用户账号、备份文件等，超级用户可以定制Windows系统和自动完成一些任务。总之，WSH有助于提高系统管理的效率。WSH提供了访问系统中COM对象的方法。因此，脚本可以通过COM对象提供的特性来完成各种任务。也可以将新的COM对象加入系统，利用这些对象，可以增加很多脚本化的功能，而无需改动底层的脚本语言。

但是，市面上专门讨论WSH的书籍很少，大多数系统管理员和程序员对WSH的了解很不够，大家在管理Windows操作系统时只是使用Windows提供的管理工具或第三方工具。这些工具虽各有优势，但也各有缺陷和局限。有什么工具能比我们自己创建的工具更能满足我们的需要呢？

其实，WSH并不神秘，你只要稍微研究一下就会发现这一点。如果你能在WSH上花一点工夫，不但能在你的“武器库”里添加一件利器，而且会使你的工作更轻松。（程序员的工作不就是让人们的工作和生活更轻松吗？）当你设计了一个工具，让它替你进行繁重、重复的工作的时候，就像看着自己设计的机器人在料理家务。相信我，使用WSH，这并不难做到。

我们很高兴能接到翻译本书的任务，这对于我们是一个学习的机会，更不用提能为大家提高技艺做出贡献给我们带来的成就感。本书的翻译工作主要由马朝晖和陈美红承担，潘浩、楼涵、董小蕾、王悦、李军、罗伟、鲍广华、瞿兰、陆明参与了部分翻译、录入和校对工作。

由于时间仓促，加之我们的水平有限，翻译过程中难免有不当之处，恳请读者批评指正。

译者

2000年12月9日

47.3

前言

当Microsoft Windows 3.1发布的时候,我希望它配有一种操作系统级的主机脚本语言。其他操作系统提供了这种语言,但Microsoft Windows 3.1只包含了旧的MS-DOS系统的批处理文件。基于很多原因,我不喜欢使用第三方的解决方案,如PowerBASIC和Rexx。在Microsoft Windows 95出现之后,我有时使用Microsoft Visual Basic、Visual Basic for Applications (VBA)和HTML脚本语言。1998年,当为德国微软出版社撰写Microsoft Windows 98手册时,我发现Windows 98有了一个新工具,称为Windows Scripting Host (WSH),它提供了两种脚本语言:Microsoft Visual Basic脚本版本(VBScript)和Microsoft JScript。

当年晚些时候,我开始为微软出版社撰写Microsoft Windows 98 Power Toolkit。我希望在书中广泛地介绍WSH,并且说明如何使用脚本语言自动地完成一些工作,如备份文件、映射网络驱动器和打印机以及显示用户对话框。但是,我找不到任何与WSH相关的文件或者任何随Windows 98发行的与脚本编程有关的帮助文件。Windows Script Host Reference基本上是白纸一张,相关的书籍和网站仅仅涉及了使用VBScript和JScript进行HTML脚本编程。所以我开始研究随Windows 98一起发行的WSH例程,浏览Microsoft的脚本网站,并开始编写我自己的WSH例程。

我尽了很大努力去解决以下问题:在给定的情况下使用什么样的对象,怎样得到属性和方法的信息,如何用VBScript或者JScript实现一个脚本。调试我的第一个脚本是个痛苦的过程。WSH 1通常含义模糊地报告在某一行为运行期错误,或者只是脚本没有按我的希望运行(当时,我还不知道在脚本环境中怎么使用Microsoft Script Debugger)。遇到运行期错误的时候,我就会用Notepad一遍一遍地打开脚本程序,手工寻找出错误的那一行,一次次地通过查错-修改-出错的过程改进代码。几个小时以后,我发现这种方法不可能是进行脚本编程的正确方法。

当访问WSH新闻组时,我发现很多人遇到了同样的问题。新闻组的参与者Ian Morrish、Clarence Washington Jr.和Michael Harris教了我很多有用的知识,但我还是需要了解更多的细节,尤其是在高级编程方面。我曾经撰写过关于VBA编程和HTML脚本的书籍,所以很熟悉使用HTML脚本和ActiveX编程进行网站开发的问题。我知道一些编辑工具支持脚本编程,也知道怎么样使用Visual Basic中的Object Browser这一类脚本调试器和工具了解对象及其方法和属性。在相当舒适的编程环境中,我可以极大地提高工作效率并生成脚本。

最初,我使用典型的“宏程序员”方法——尝试将简单的语句组合在一起,构成脚本程序。有时这种方法是起作用的,但是通常会遇到语法错误或运行期错误。同时,我也没有使用那些功能强大的语句,以完成文件拷贝、程序启动、得到用户输入信息等任务。在WSH新闻组中,有人推荐了一些第三方工具和语言,其中包括丰富的语句集合。但是,根据我过去的经验,一种语言提供的语句集合越丰富,其学习和使用的难度就越大。

以VBA的编程经验为基础,我开始研究脚本编程背后的学问。我发现,为了完成参数读取、

文件拷贝等任务，不必将语句“埋藏”在脚本语言当中，因为WSH提供了访问系统中COM对象的方法。因此，脚本可以通过COM对象提供的特性来完成这些任务。也可以将新的COM对象加入系统，利用这些对象，可以增加很多脚本化的功能，而无需改动底层的脚本语言。例如，如果一个对象可以实现拼写检查的功能，你只需在脚本中使用它就可以了。当更加熟悉WSH时，我发现WSH是Microsoft近年来最为炙手可热的技术。WSH将众多对象（如Microsoft Office拼写检查器、文字处理器、浏览器和电子邮件系统）结合起来，构成一个新的、定制的应用软件。只使用几行代码，即可迅速而简单地实现一个复杂的应用。

目前为止，我已经收集了很多资料，得到了一些经验，并写出了很多例程。我还为Microsoft Windows 98 Power Toolkit写了关于WSH的简单片段。在1998年法兰克福图书展览会上，我遇到了微软出版社的编辑Thomas Pohlmann和Stephen Guty，一个念头出现在我的头脑中，这就是写一本关于WSH脚本编程的书。六周后，我的德文手稿完成了。此书的第一版于1999年1月出版了，这是全世界第一本关于WSH 1的出版物。当年春天，WSH 2的测试版发布了。我们决定到Windows 2000发布之后再出版WSH 2的书籍。我利用业余时间研究了WSH 2和其他一些热门技术，如ActiveX Data Objects (ADO)、Active Directory Service Interfaces (ADSI)和Windows Management Instrumentation (WMI)。同时，我为杂志写了面向WSH初学者的系列文章，在我的网站上开了WSH专题区，并花了很多时间在WSH的新闻组上，更多地了解脚本程序员遇到的问题。

我收集到的资料足可以完成两本书——一本面向初学者和中等读者，另一本面向高级用户、探讨脚本技术的书。微软出版社将在今年晚些时候出版第二本书《Advanced Development with Microsoft Windows Script Host 2.0 (Microsoft Windows Script Host 2.0的高级开发)》。此书面向那些高级脚本程序员，他们希望深入地研究WSH方面的课题。《Advanced Development with Microsoft Windows Script Host 2.0》将介绍如何使用更多的对象来扩展脚本的能力。书中将涉及一些特别的专题，如.wsf文件的XML结构、.wsf文件未公开的技巧。书中也将介绍怎样使用一些附加的工具来进一步了解对象及其方法和属性，这些工具包括OLE/COM Object Viewer或Object Browser。书中介绍了其他有用的技术，使用这些技术，可以完成以下功能：通过WSH脚本访问Microsoft Office应用程序，使用Microsoft Excel电子数据表、Office拼写检查器、Microsoft Word文件和Microsoft Outlook这些对象。书中还介绍了使用脚本语言，通过Microsoft Access或ActiveX Data Objects (ADO)读写数据库的方法。此书也讲解了系统管理员如何通过WSH脚本访问Active Directory Service Interfaces (ADSI)，从而维护用户账号、服务和共享。书中介绍了新技术：基于Web的企业管理 (Web-Based Enterprise Management, WBEM)和微软对这种技术的解决方案，叫作Windows Management Interface (WMI)。在书里介绍了如何使用脚本访问网络和设备信息以及终止正在运行的进程。书中介绍了如何使用Visual Basic 5 Control Creation Edition (CCE)生成ActiveX控件，用这些控件实现以下功能：访问Windows API，生成ActiveX表单，访问剪贴板，控制CD，等等。我在书中将提供一些ActiveX控件的细节和源代码。本书讲解了WSH脚本编程的初级问题，而《Advanced Development with Microsoft Windows Script Host 2.0》将提供WSH高级程序员所需的信息。你将学到那些专业开发者所用的技巧，以便发掘WSH的最大潜力。

本书的目的

如果你是有经验的脚本程序员，那么应该阅读《Advanced Development with Microsoft Windows Script Host 2.0》。本书是面向其余那些对WSH感兴趣的读者，包括：

- 系统管理员，希望通过实现自动的解决方案管理用户账号、备份文件等，还有想了解脚本编程的读者。
- 超级用户，希望使用脚本语言定制Windows系统和自动完成一些任务。
- 程序员，希望很快地了解WSH编程，并通过一些想法和例程理解其他人如何使用WSH。
- 管理员，希望了解WSH的功能，使用WSH脚本提高系统的效率。

本书涉及的范围

本书并不是所有与脚本编程相关的课题的完整参考书，只是简单的、一步一步的初级读物。同时，因为大多数有经验的用户和系统管理员了解一点批处理和宏编程，但不熟悉面向对象的编程，本书介绍了如何通过VBScript和JScript来使用对象和进行编程。

读完本书之后，你可以了解WSH的功能，了解VBScript和JScript的基础，可以使用WSH提供的对象的方法和属性编写脚本程序。

你可以一页不少地阅读本书，但其实不需要这样做。你可以根据自己的水平和需要，将注意力放在某些部分和章节上。

本书分为四部分。第一部分“脚本编程的介绍”讲解了一些基础问题。我希望那些从未编写过程序的读者从这一部分开始阅读。第1章介绍了WSH和如何执行脚本。你将了解各种脚本引擎，学到如何向脚本程序传递参数，如何使用选项，如何设置脚本属性。第2章介绍了简化脚本开发的工具。你可以学到怎样使用脚本编辑器，如何安装ActiveX控件，如何调试脚本程序。从未接触过面向对象编程的读者应该阅读第3章，这一章解释了对对象、方法、属性、集合和对象模型这些概念。第4章介绍了VBScript，而第5章介绍了JScript。

第二部分“交互式的脚本”说明如何得到用户的输入，并在对话框和表单中输出结果。第6章介绍用于在对话框中观看结果的方法和函数，包括Echo方法、VBScript的MsgBox函数和Popup方法。第7章讲解如何使用WSH的对象。给出的例子使用WScript对象来获得脚本和语言的属性，检验提交给脚本的参数，访问环境变量。这一章介绍了以下内容：如何在脚本中生成和释放对象实例，如何在脚本中使用Run方法启动程序。第8章介绍如何使用VBScript的InputBox函数生成输入对话框，如何扩展JScript（它没有InputBox函数）使之实现同一功能。这一章提供了一些例子来讲解使用Microsoft Internet Explorer对象提高脚本的功能。第9章介绍如何使用Internet Explorer对象生成对话框和输入表单。

第三部分“高级脚本编程”介绍了更多的对象。第10章介绍使用WSH脚本管理Windows桌面和Start菜单中的快捷方式。第11章介绍了高级WSH任务，例如获取用户名、域名或计算机名，这一章讲解了使用WSH脚本在网络环境中进行打印机和驱动器的映射，使用WSH脚本访问Windows注册表。第12章讲解了如何使用FileSystemObject对象处理驱动器、文件夹和文件。例如，如何使用脚本枚举文件和文件夹，查询驱动器属性，备份文件。第13章涉及了WSH 2的一些特性，例如

延迟脚本的运行, 播放声音, 使用脚本运行多个应用程序。这一章介绍了如何将一个应用程序的窗口切换到前台, 如何使用SendKeys方法模拟键盘输入。第14章提供了很多脚本编程的技术、技巧和诀窍, 介绍了如何通过脚本得到当前路径、关闭Windows、打印和调用Windows外壳。

第四部分“附录”, 包括附录A和附录B。附录A例举了Internet上的WSH资源。附录B介绍脚本的安全问题, 包括及时和最新的信息和策略, 使Windows系统在脚本病毒面前更加安全。

示例文件和工具

本书提供的示例文件可以在附带的光盘上找到。示例文件按章节分开, 保存在\WSHDevGuide文件夹中的\Chapter1、\Chapter2等子文件夹中。附带光盘提供的用户界面可以用于浏览这些示例文件, 也可以使用安装程序把示例文件安装到本地硬盘上。在Start菜单中运行D:\Setup.exe (D是光驱的盘符), 就可以按照章节将示例文件安装好, 例如, 第3章的示例将安装在\WSHDevGuide\Chapter3文件夹中。

在附带的光盘上, 有一些有用的工具和文件, 包括Windows Script Host 2。光盘上还有WSH 2、VBScript和JScript的程序员参考和帮助文件。打开光盘根目录中的Start.htm即可进入用户界面, 浏览光盘的内容。

注意 切记脚本程序可能严重地损坏你的系统。在试图更改和使用能够改变文件内容、注册表、系统设置的那些脚本程序前, 一定要备份系统文件和其他重要文件。使用书中的示例文件会冒风险, 尤其是在这些文件被改动之后。无论微软或我都不提供脚本的支持, 所以改动这些示例时要万分小心。

更多书籍

- Windows Script Host Reference 这是关于WSH对象及其方法和属性的帮助文件, 还包括WSH指南。它在本书附带的光盘中。
- VBScript Language Reference 这是权威的VBScript参考。这个帮助文件还包括VBScript指南和FileSystemObject对象指南。它在本书附带的光盘中。
- JScript Language Reference 这个帮助文件记录了JScript的语法和JScript提供的所有的方法和属性。它同时包括JScript指南和FileSystemObject对象指南。它在本书附带的光盘中。

系统要求

使用本书需要Windows系统 (Windows 95、Windows 98、带有Service Pack 4的Windows NT或Windows 2000), 系统需要安装Windows Script Host (推荐使用版本2)。一些例程需要Microsoft Internet Explorer (版本4或更高)。Windows系统提供了书写脚本的工具 (例如编辑器), 在本书附带的光盘中也有这种工具。有些工具可能要从Internet上下载。

其他信息

本书和其附带光盘的准确性已经过很多测试。微软出版社对本书的修正可在mspress.

microsoft.com/support上找到。

任何关于本书和其附带光盘的注释、问题和想法可以发送到微软出版社 (mspinput@microsoft.com), 或发送到以下邮件地址:

Microsoft Press

Attn: Microsoft Windows Script Host 2.0 Developer's Guide Editor

One Microsoft Way

Redmond, WA 98052-6399

如果你有关于本书的事情要和我联系, 可以浏览我的站点 www.borncity.de。

我要再次提示, 以上任何地址均不提供脚本支持。

目 录

译者序

前言

第一部分 脚本编程简介

第1章 Windows Script Host介绍	1
1.1 WSH可以做什么	1
1.2 升级到WSH 2	2
1.3 生成和调用脚本	3
1.3.1 编写第一个VBScript脚本	4
1.3.2 用JScript编写同样的脚本	5
1.3.3 运行WSH脚本	6
1.3.4 向脚本提交参数	9
1.3.5 新的.wsf文件格式	12
1.3.6 包含外部文件	13
1.3.7 使用类型库	14
1.3.8 调试特性	14
第2章 开发工具和技术	16
2.1 生成脚本	16
2.2 编辑脚本	18
2.3 安装和卸载ActiveX控件	24
2.3.1 注册控件	24
2.3.2 控件的卸载	25
2.4 脚本调试	25
2.4.1 程序跟踪	26
2.4.2 使用Microsoft Script Debugger	27
第3章 对象编程介绍	33
3.1 对象	33
3.1.1 一个简单对象的例子	33
3.1.2 对象模型	33
3.1.3 集合	34
3.1.4 方法	34
3.1.5 实例	35

3.2 可用的对象	35
第4章 VBScript简介	37
4.1 基本特点	37
4.1.1 语句、延续行和注释	37
4.1.2 常量和变量	39
4.1.3 操作符	45
4.1.4 控制结构	49
4.1.5 循环	50
4.1.6 函数和过程	52
4.2 高级特性	59
4.2.1 错误处理	59
4.2.2 规则表达式	61
4.2.3 类	63
4.2.4 使用With语句	67
第5章 JScript介绍	68
5.1 JScript是什么	68
5.2 JScript程序的结构	68
5.2.1 注释	69
5.2.2 语句	69
5.2.3 常量	70
5.2.4 变量	70
5.2.5 字符串中的转义符	73
5.3 操作符	74
5.3.1 赋值操作符	74
5.3.2 比较操作符	74
5.3.3 计算操作符	75
5.3.4 增量和减量操作符	75
5.3.5 逻辑操作符	76
5.3.6 操作符的优先级	77
5.4 控制结构	77
5.4.1 if语句	77
5.4.2 条件操作符	78

5.4.3 for循环	78	7.3.2 使用DisconnectObject	127
5.4.4 for...in循环	80	7.4 在脚本中启动其他程序	129
5.4.5 while循环	80	7.4.1 从VBScript中启动Notepad	130
5.4.6 do...while循环	81	7.4.2 从JScript中启动Calculator	131
5.4.7 switch语句	82	7.4.3 包含空格的路径	132
5.4.8 break和continue关键字	83	7.4.4 Quit方法	133
5.5 函数和对象	83	7.4.5 使用Run方法执行MS-DOS命令	136
5.5.1 用户定义函数	83	第8章 在WSH脚本中得到用户输入信息	139
5.5.2 内置函数	83	8.1 在VBScript中调用一个输入对话框	139
5.5.3 对象	84	8.2 在JScript中调用输入对话框	141
5.5.4 数组	84	8.2.1 WSH 2的解决方案:结合 VBScript和JScript	141
第二部分 交互式的脚本		8.2.2 使用Internet Explorer的prompt方法	143
第6章 在WSH中生成简单的对话框	87	第9章 使用表单	148
6.1 使用Echo方法	87	9.1 使用Internet Explorer创建About对话框	148
6.1.1 在VBScript中使用Echo方法	87	9.1.1 使用showModalDialog方法 显示HTML文件	148
6.1.2 在Command Prompt窗口中使用Echo	88	9.1.2 使用VBScript显示About对话框	151
6.1.3 在JScript中使用Echo	89	9.1.3 使用JScript显示About对话框	153
6.1.4 在Echo中换行	90	9.2 使用Internet Explorer和WSH创建表单	154
6.2 在VBScript中使用MsgBox函数	91	9.2.1 表单的HTML代码	154
6.2.1 定义对话框中的图标和按钮	92	9.2.2 显示表单	155
6.2.2 在按钮上设置焦点	94	9.2.3 用于显示表单的JScript WSH脚本	158
6.2.3 检查用户使用哪一个按钮来关闭 对话框	95	9.2.4 显示文件选择对话框	160
6.2.4 范例: VBScript中的一个欢迎登录 信息	97	9.2.5 改进表单	162
6.3 使用Popup方法	99	第三部分 高级脚本编程	
6.3.1 在VBScript中使用Popup	102	第10章 创建快捷方式	173
6.3.2 使用Popup的另一个JScript范例	103	10.1 快捷方式基础	173
第7章 使用WSH对象	106	10.2 快捷方式的高级应用	177
7.1 使用WScript对象	106	10.2.1 使用SpecialFolders对象	177
7.1.1 得到WSH和脚本的属性	106	10.2.2 在桌面上创建快捷方式	183
7.1.2 得到语言引擎属性	109	10.2.3 在Start菜单上创建快捷方式	187
7.1.3 访问脚本参数	110	10.2.4 使用参数创建快捷方式	192
7.2 访问环境变量	114	10.2.5 读取快捷方式属性	197
7.3 生成和释放对象	125	10.2.6 更新快捷方式	202
7.3.1 CreateObject和GetObject	126	10.2.7 创建Web站点的快捷方式	205

第11章 使用高级的WSH特性.....209	
11.1 获取用户、域或计算机名称209	
11.2 连接到网络打印机211	
11.2.1 使用AddWindowsPrinter Connection 映射打印机212	
11.2.2 使用AddPrinterConnection映射 打印机213	
11.2.3 删除打印机映射216	
11.2.4 列出所有被映射的打印机218	
11.2.5 设置默认打印机219	
11.3 映射网络驱动器221	
11.3.1 JScript实现223	
11.3.2 登录脚本225	
11.4 访问Windows注册表225	
11.4.1 在WSH中注册表访问227	
11.4.2 在VBScript中访问注册表228	
11.4.3 注册表访问的运行期错误处理231	
11.4.4 检查键是否存在233	
11.4.5 在WSH中访问注册表的技术237	
11.4.6 枚举注册表键和值241	
11.4.7 远程访问注册表243	
11.4.8 改变Windows 98安装路径244	
11.4.9 在登录时隐藏前一个用户名246	
11.4.10 在Windows 98中获取工作组名称248	
第12章 使用文件系统和I/O命令251	
12.1 FileSystemObject对象模型251	
12.1.1 创建一个FileSystemObject对象251	
12.1.2 FileSystemObject对象的方法252	
12.2 访问驱动器253	
12.2.1 列出一台机器上的所有驱动器253	
12.2.2 显示驱动器的属性257	
12.3 访问文件和文件夹263	
12.3.1 列出一个文件夹中的所有子 文件夹263	
12.3.2 创建、移动、重命名和删除 文件夹266	
12.3.3 列出一个文件夹中的所有文件273	
12.3.4 获取文件属性和日期275	
12.3.5 复制和删除文件282	
12.3.6 备份文件夹285	
12.4 访问文本文件290	
12.4.1 读文本文件290	
12.4.2 使用对话框或浏览器窗口观看 文本文件的问题292	
12.4.3 写文本文件300	
12.4.4 附加新的文本到已存在的文件303	
12.4.5 替换文件中的文本305	
12.5 使用Browse For Folder对话框308	
12.5.1 使用BrowseForFolder选择文件夹308	
12.5.2 使用BrowseForFolder选择文件313	
第13章 从脚本控制Windows和应用程序317	
13.1 延迟脚本的执行317	
13.1.1 使用Sleep方法减轻CPU负载317	
13.1.2 使用延迟解决同步进程的问题318	
13.2 使用AppActivate方法激活一个 应用程序320	
13.3 使用SendKeys模拟击键323	
13.3.1 在WSH 2中使用SendKeys324	
13.3.2 使用SendKeys操作两个应用程序327	
13.4 使用登录脚本定制Windows329	
13.4.1 建立登录脚本329	
13.4.2 使用全局登录和注销脚本331	
13.4.3 使用启动和关机脚本332	
13.4.4 Windows 95和Windows 98中的启 动和登录脚本332	
第14章 编程技术与技巧334	
14.1 运行期错误334	
14.1.1 在VBScript中处理运行期错误334	
14.1.2 在JScript中处理运行期错误334	
14.1.3 在VBScript中产生一个运行期错误335	
14.2 路径和日期335	
14.2.1 获取脚本的路径335	
14.2.2 得到当前目录336	
14.2.3 设置默认文件夹337	

14.2.4 得到当前驱动器名称	337	14.5 文件处理	357
14.2.5 计算日期差	337	14.5.1 检查文件或文件夹是否存在	357
14.3 长文件名、At命令和系统调用	338	14.5.2 检查文件夹是否是空的	357
14.3.1 使用Windows NT Scheduler 执行WSH脚本	338	14.5.3 检查一个Access数据库是否正 在使用	358
14.3.2 在脚本中使用长文件名	338	14.5.4 复制文件	359
14.3.3 使用Run方法执行系统调用	339	14.5.5 文件或文件夹的重命名	359
14.3.4 锁住Windows 2000工作站	342	14.5.6 搜索文件	361
14.3.5 调用Copy Disk对话框	342	14.5.7 列出所有快捷方式文件	361
14.3.6 调用Format对话框	343	14.6 播放声音	363
14.3.7 调用Screen Saver属性页	344	14.7 Windows外壳	366
14.3.8 调用Control Panel模块	345	14.7.1 检查外壳的版本	366
14.3.9 使用Run方法处理网络映射	345	14.7.2 安排桌面窗口	367
14.4 用户对话框和输出	346	14.7.3 打开文件夹窗口	369
14.4.1 使用制表符和换行	346	14.7.4 访问Windows外壳的对话框	370
14.4.2 显示控制台输入和输出	346	14.8 调用DUN连接	373
14.4.3 写到一行和从中读取	348		
14.4.4 程序输出的重定向	349		
14.4.5 为流使用文件	350		
14.4.6 脚本输出的日志	353		
14.4.7 从WSH脚本进行打印	356		

第四部分 附 录

附录A Internet上的WSH资源	375
附录B 脚本安全	376

第一部分 脚本编程简介

第1章 Windows Script Host介绍

在Windows 98以前, Microsoft的Windows系统几乎没有提供自动完成任务的功能, 这些任务包括文件备份和实现例行的系统管理。当然, 可以使用运行在MS-DOS提示符窗口的MS-DOS批处理(BAT)文件, 来完成特定的工作, 例如拷贝文件。但是BAT文件只能包含一系列简单的MS-DOS命令(只有简单的分支, 没有真正的循环, 还有其他缺点), 而且不支持对话框和信息框。在Windows 3.1中, 宏记录器提供了稍微多一些的功能, 可以记录和回放简单的键盘输入和鼠标点击, 但是宏记录器是不能被编程的。

用户为了寻找功能更加强大的方法来执行自动化的任务, 转向第三方解决方案, 如PowerBatch或像Delphi、Microsoft Visual Basic 和Microsoft Visual C++这样的编程环境。但是许多Windows用户发现这些解决方案既不简单, 又不免费, 所以不容易接受。

因为Microsoft Office提供了Visual Basic for Applications (VBA), 而且网络开发者知道像Microsoft Visual Basic的脚本版 (VBScript)和Netscape的JavaScript这样的脚本语言已经存在, 所以Microsoft为Windows操作系统提供脚本工具只是个时间问题。这个工具就是Microsoft Windows Script Host (WSH), 它受到了微软的Windows Script技术的影响。

WSH是单机主机, 它允许直接在操作系统级别运行脚本文件。例如, 可以在命令行界面调用脚本, 也可以在Windows Explorer中双击脚本文件来调用它。WSH对于很多只需要很少用户界面或不需要用户界面的管理任务是十分方便的。因为JScript和VBScript是可以全面地访问WSH对象和其他任何Automation对象的脚本语言, 所以WSH比MS-DOS批处理文件具有更大的通用性。

注意 在版本2之前, Windows Script Host被称为Windows Scripting Host。本书中, 二者均被称为WSH。

1.1 WSH可以做什么

可以通过很多方式调用脚本来定制Windows系统。这里是一些使用脚本自动完成任务的例子:

- 备份和恢复系统中的文件(如果仅仅需要从本机保存几个文件到网络服务器, 这项功能更加方便)。
- 通过鼠标点击关闭或重新启动Windows。也可以使用脚本完成特殊的关闭或启动任务, 例如在关闭应用程序之后备份特定的文件, 或在系统启动之后记录用户名(在Windows 95和

Windows 98中,通过脚本记录信息的功能非常有用;在Windows NT和Windows 2000中,很多记录的功能已经内置了)。

- 集成应用程序和其数据。例如,脚本可以启动一个Office应用程序,加载和处理一个文档,打印这个文档,然后关闭此应用程序。通过这种方式使用脚本,可以将文档和所选的应用程序“结合”起来。
- 在Windows NT和Windows 2000中完成系统管理的工作,例如添加、更改、删除用户账户。利用Windows NT和Windows 2000提供的Active Directory Service Interfaces (ADSI),可以使用WSH脚本自动完成这些工作。
- 通过适当的对象直接访问Windows外壳(生成快捷方式或者映射诸如驱动器、打印机等网络设备)。
- 读取Windows环境变量或者获取关于Windows的信息。
- 启动程序和控制Automation对象。
- 显示对话框,以便通知用户程序的状况或者得到用户输入。
- 访问Windows外壳和Windows应用程序编程接口(API),以便控制窗口和其他应用程序。

VBScript和JScript的一些问题

WSH和Windows 98以及Windows 2000一起出售(对于Windows 95和Windows NT 4有一个供下载的版本),WSH包括两种编程语言,VBScript和JScript。VBScript使用与Visual Basic类似的语法;它其实是Visual Basic的子集。JScript是Microsoft对ECMAScript的实现,是基于JavaScript与生产厂商无关的编程语言。

通过以上两种编程语言,即可进入脚本编程的世界了。不管怎样,Microsoft为第三方软件的生产厂商设计了用于WSH的开放接口,使他们可以集成自己的语言引擎以支持其他语言,如Perl、Tool Control Language (Tcl) 和Rexx。

1.2 升级到WSH 2

WSH 1最初随Windows NT 4 Option Pack发布。WSH 1也是Windows 98的一部分,但它并不处于激活状态,必须将它作为一个Windows附加组件来安装。WSH 2是Windows 2000整体的一部分,如果已经安装了Windows 2000,则不需再进一步安装WSH 2。



图1-1 脚本文件的扩展名和图标

如果要检查系统中是否安装了WSH,只需要打开一个包括.js文件、.vbs文件和.wsf文件的文件夹(例如,使用Windows资源管理器)。如果文件的图标如图1-1中所示,说明WSH 2已经安装了。如果.wsf文件所用的图标是未知文件类型的图标,说明系统中安装了WSH 1。如果所有的文件均为未注册文件类型,则WSH没有安装。

VBScript和Visual Basic/VBA以及JScript和JavaScript

对于使用过Visual Basic或VBA的程序员,或者为HTML文档编写过VBScript或JScript

脚本的程序员，用VBScript或JScript编写WSH脚本不会有很大难度，但是一定要切记以下几点：

- Visual Basic程序可以编译为EXE文件，但是WSH不提供这一功能。在WSH中，所有的脚本程序必须以.vbs、.js或.wsf扩展名保存。WSH使用Windows Script引擎直接解释脚本文件的内容，所以脚本可以使用简单的文本编辑器编写（例如Notepad）。
- VBScript和JScript（随WSH 2一起发布）不包括语言结构，例如Visual Basic和VBA中的Declare语句，这种语句允许访问外部的函数和过程。VBScript不包括实时错误处理的功能（例如On Error GoTo label）。无论VBScript或JScript均不支持显式的数据类型声明；相反，这两种语言将所有的变量视为Variant。
- WSH提供的对象模型与Internet Explorer提供的对象模型不同。例如WSH不支持用于用户输出的window.document对象。
- 因为WSH环境不提供扩展的用户界面（像Internet Explorer所做的），WSH对象模型不暴露用户界面事件（例如在HTML脚本中的onclick事件）。但是，WSH支持事件处理，这在下面的章节中将讲到。

要把Windows 95、Windows 98或Windows NT 4升级到WSH 2，必须手工进行安装。本书附带的光盘包括针对于这些操作系统的WSH 2安装程序。也可以在站点<http://msdn.microsoft.com/scripting>免费下载最新的WSH文件。

注意 对于Windows NT 4，必须安装Service Pack 3。在Windows 95或Windows NT 4平台上使用WSH，必须具备Internet Explorer 4或更高版本。在安装了Internet Explorer 3的Windows 95平台上运行WSH 2，必须安装DCOM。具体细节见Microsoft网站：<http://msdn.microsoft.com/scripting>。

WSH和语言引擎的版本

你必须了解WSH的版本和用来运行脚本的语言引擎的版本。Microsoft在版本3.1（随WSH 1发布）之后，发布了几个语言引擎的版本。版本3.1a修正了一些错误。和Visual Studio 6一起发布的语言引擎是版本4。Internet Explorer 5包括版本5的语言引擎，它修正了错误，并扩充了语言功能。Microsoft也提供了一个单独的版本5的语言引擎以供下载（<http://msdn.microsoft.com/scripting>）。

WSH 2是Windows Script 5.1的一部分，WSH 2包括了VBScript和JScript语言引擎5.1版。5.1版的语言引擎可以和Internet Explorer 4、5以及IIS 3、4一起使用。

1.3 生成和调用脚本

准备好编写第一个脚本了吗？在一开始，我们编写一个简单的程序，显示一个对话框，其内容是“Hello, world”。通过这个例子，可以了解基本的脚本编程方法和执行方法。

1.3.1 编写第一个VBScript脚本

生成一个简单的如图1-2中所示的对话框，只需一行VBScript语句：

```
WScript.Echo "Hello, world"
```

但是如何生成脚本文件呢？对于以上的例子，只需要Notepad就可以了。具体步骤如下：

1) 启动Notepad.exe，输入图1-3所示的语句。WScript.Echo语句是代码中最重要的一部分，因为它生成对话框。



图1-2 一个简单的对话框

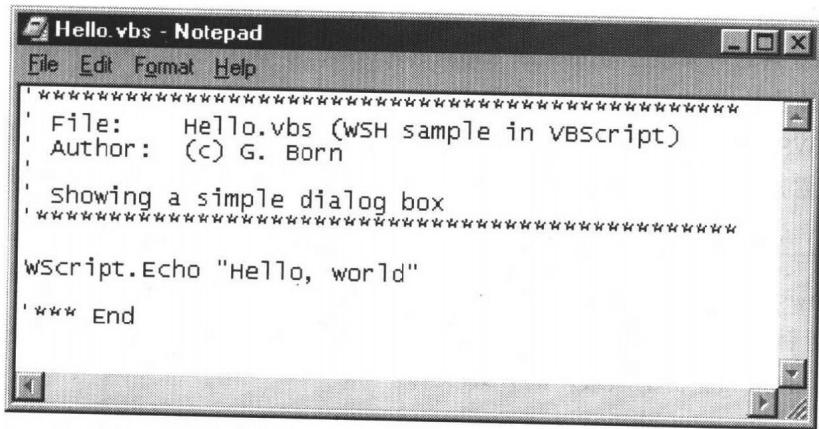


图1-3 第一个VBScript程序

2) 将编辑窗口中的内容保存到硬盘上的一个文本文件。可以使用任何文件名（如Hello），但文件扩展名必须是.vbs。

注意 我推荐使用.vbs作为VBScript文件的扩展名。这种文件类型是为WSH自动注册的。如果使用其他的文件扩展名，任何版本的WSH都不会识别文件中的脚本语言。双击一个扩展名错误（如.vb）的脚本文件，Open With对话框会出现，因为这种文件类型没有注册对应的应用程序。

保存脚本之后，在硬盘上就会有一个名为Hello.vbs的文件，其源代码见清单1-1。

清单1-1 Hello.vbs

```
*****
' File:      Hello.vbs (WSH sample in VBScript)
' Author:    (c) G. Born
'
' Showing a simple dialog box
' *****

WScript.Echo "Hello, world"
'*** End
```