

万水 Netscape 从入门到精通系列

Official Netscape JavaScript Book

Netscape JavaScript 指南

冯志强 孙江宏 刘芝泉 译
曹 康 审校



美] Peter Kent 著
John Kent

[http://www.netscapepress.com/
support/javascript/](http://www.netscapepress.com/support/javascript/)



中国水利水电出版社

TP312
KT/1

万水 Netscape 从入门到精通系列

Netscape JavaScript 指南

[美] P. 肯特 John 肯特 著
冯志强 孙江宏 刘芝泉 译
曹 康 审校

中国水利水电出版社
1997

0049109

内 容 简 介

本书全面介绍了 WWW 上的最新编程语言 JavaScript。全书共 19 章,可分为五个部分: JavaScript 简介和一些应用实例;变量、表达式与运算符、数组、条件与循环,以及函数;对象、属性、方法,以及事件处理程序;用 JavaScript 控制窗口、文档、框架和窗体,以及为用户通信;最后部分为全书综合示例的一个地区代码应用程序。另外,后面附录部分为读者更有效地利用本书及深入学习 JavaScript 提供了不可多得的帮助和指导。

本书内容详实,叙述深入浅出,示例丰富,可操作性强,还有相配套的 Online Companion(联机伴侣)站点,所以特别适合于广大初学编程的用户,同时也是有经验的程序员从事 Web 页面编程的优秀参考书。

Peter Kent, John Kent: Official Netscape JavaScript Book

"Original English language edition published by Ventana Communications Group, Inc., P. O. Box 13964, RTP, North Carolina 27709 - 3964. TEL: 919/544 - 9404, FAX: 919/544 - 9472.

Copyright © 1996 by Ventana Communications Group. All rights reserved."

北京市版权局著作权合同登记号:图字 01-97-0064

版权所有,翻印必究

图书在版编目(CIP)数据

Netscape JavaScript 指南/(美)肯特(Kent, P.), (美)肯特(Kent, J.)著;冯志强等译. - 北京:中国水利水电出版社, 1997.3

(万水 Netscape 从入门到精通系列)

书名原文: Official Netscape JavaScript Book

ISBN 7-80124-388-9

I. N… II. ①肯… ②肯… ③冯… III. 程序语言, NetscapeJavaScript - 指南 IV. TP312-62

中国版本图书馆 CIP 数据核字(97)第 02984 号

书 名	Netscape JavaScript 指南
作 者	Peter Kent John Kent 著
译 者	冯志强 孙江宏 刘芝泉
审 校	曹 康
出版发行	中国水利水电出版社(北京市三里河路 6 号 100044) 北京万水电子信息有限公司(北京市西城区三里河一区 39 栋 100045)
排 版	北京万水电子信息有限公司
印 刷	北京 714 印刷厂印刷
规 格	787×1092 毫米 16 开本 19.75 印张 446 千字
版 次	1997 年 4 月第一版 1997 年 4 月北京第一次印刷
印 数	0000—5000 册
定 价	36.00 元

出版者的话

近年来,中国水利水电出版社及其与美国万国集团的合资公司北京万水电子信息有限公司一直致力于电脑图书的出版。1994年,我社购买了美国 MIS 出版公司的 5 本 Windows 自学教程的中文版版权;1995年,我社又与美国万国集团联袂购买了美国 Peachpit 出版公司的 7 本电脑书(其中有 6 本是 Macintosh 苹果电脑参考书)的版权;1996 年我们已经组织数套电脑丛书,如《中文版软件详解丛书》、《万水流行软件用户伴侣丛书》、《万水电脑彩色与平面设计丛书》、《万水网络与数据库丛书》等。这些图书有的已经出版,有的将在近期内陆续推出。已经出版的图书,在出版界和电脑用户中产生了良好的影响。

1997 年我们正在组织并逐步推出《万水计算机与通信系列》、《万水电脑动画与多媒体系列》、《21 世纪高新技术指南系列》、《万水 Netscape 从入门到精通系列》、《万水计算机辅助技术系列》、《万水 Windows 软件精通指南系列》、《万水苹果电脑软硬件指南系列》、《万水国际经济合作指南系列》、《万水实用英语指南系列》、《万水现代实用写作系列》等电脑及其他方面的图书。

目前,中国水利水电出版社与北京万水电子信息有限公司筹划出版一套《万水 Netscape 从入门到精通系列》。该系列是对以计算机、网络技术著称的美国 Netscape 公司的一系列软件的介绍。Netscape 的网络浏览器可与微软件网络浏览器并驾齐驱。《Netscape JavaScript 指南》是本系列中的一种,全面介绍了 WWW 上的最新编程语言 JavaScript。

编写、出版这样的系列对我们来说尚属首次,错漏、失当之处在所难免。我们诚恳地欢迎您对本书提出修改意见,以便我们再版时进行修正;我们也热诚地欢迎您就本系列提出选题建议或自荐、推荐作者,以使其尽可能地臻于完善;如对本系列图书其它各册感兴趣,也请随时同我们联络。联络地址为:北京万水电子信息有限公司,北京市三里河一区 39 栋,邮政编码:100045,电话:(010)6852.8689,传真:(010)6853.3313, E-Mail: mchannel@public3.bta.net.cn;或:中国水利水电出版社对外合作部,北京市三里河路 6 号,邮政编码:100044,电话:(010)68358031 转,传真:(010)6835.3010。

中国水利水电出版社
北京万水电子信息有限公司
1997 年 1 月

作者简介

John Kent 居住于英国伦敦,拥有名为 CISS Ltd. 的一家软件开发公司。他采用大多数流行的语言在大型机、PC 以及 UNIX 系统中编程,这些语言包括 SQL、Visual Basic 和 C++。

Peter Kent 居住于科罗拉多州的 Lakewood,是一位自由作家和信息咨询者,在技术编著和软件界面设计方面具有十五年的工作经验。他的二十五本著作中最成功的一本是 *PGP Companion for Windows* (Ventana 出版)。

致 谢

感谢 John,如果没有他的参与就不可能有此书的面世。同时感谢我的妻子和孩子们,他们因为我的写作而放弃了度假的时间。

—— Peter Kent

感谢我的妻子 Jayne 所做的一切,还要感谢我的弟兄和合作者 Peter。编著本书是一项艰巨的工作,但我们一起合作到最后。

—— John Kent

目 录

序 言	1
0.1 准备工作	1
0.2 内容简介	2
0.3 关于 Online Companion	4
0.4 JavaScript 是活的	4
第一章 JavaScript 是什么	6
1.1 首先有了 Oak	6
1.2 Java 到底是什么	7
1.3 JavaScript 的产生	8
1.4 Java 与 JavaScript 的比较	9
1.5 从哪里得到 JavaScript	10
1.6 JavaScript 有什么用处	11
1.7 让 Java 流行	20
1.8 JavaScript 浏览器打开了吗	20
1.9 进入下一章	22
第二章 JavaScript 速成诀窍	23
2.1 在 Web 页面中置入修改的日期	23
2.2 读者来自于哪个页面	25
2.3 打开附属窗口	26
2.4 游动按钮	28
2.5 告别消息	31
2.6 状态条消息	31
2.7 向窗体添加消息	32
2.8 自动转发读者	33
2.9 自动转发——使用 Confirm 对话框	34
2.10 进入下一章	35
第三章 起步阶段：脚本、函数和注释	36
3.1 运行 JavaScript 的两种方式	36
3.2 <SCRIPT>标记	37
3.3 使用脚本源文件	39
3.4 函数简介	40

3.5	注释标记<!--和/-->	41
3.6	JavaScript 注释	42
3.7	错误警告——请使用 HEIGHT 和 WIDTH 特性	43
3.8	进入下一章	43
第四章	变量与字面量——存储数据	44
4.1	变量命名	46
4.2	变量类型	47
4.3	变量作用域——变量可用于何处	50
4.4	为何要使用 var	53
4.5	什么是字面量	54
4.6	进入下一章	58
第五章	处理值的表达式与运算符	59
5.1	在表达式中使用运算符	59
5.2	字符串运算符	63
5.3	条件运算符	66
5.4	运算符优先顺序	66
5.5	类型转换	68
5.6	进入下一章	70
第六章	条件与循环——判定和控制脚本	71
6.1	if 语句	71
6.2	else 子句	74
6.3	条件运算符	75
6.4	if 语句的另一种格式	81
6.5	循 环	81
6.6	进入下一章	88
第七章	函数的深入讨论	89
7.1	定义函数	89
7.2	函数可以返回值	91
7.3	将值传入函数	95
7.4	利用函数工作	97
7.5	内置函数	99
7.6	进入下一章	106
第八章	故障排除及问题的避免	107
8.1	不只是用户程序有错误	107

8.2	避免问题	107
8.3	追查问题	111
8.4	进入下一章	115
第九章	建立数组	116
9.1	创建一个数组	116
9.2	从数组中获取数据	119
9.3	数组匹配	123
9.4	捷 径	125
9.5	真正的数组对象	126
9.6	进入下一章	127
第十章	对象、属性和方法	128
10.1	属性——属于对象的变量	129
10.2	方法——属于对象的函数	132
10.3	字符串变量是对象	133
10.4	对象的层次结构——document 对象	133
10.5	对象类与实例	140
10.6	混乱的名字——location	141
10.7	层次结构	143
10.8	本章摘要	143
10.9	进入下一章	144
第十一章	内置对象的深入讨论	145
11.1	Date 对象	145
11.2	Date 方法	145
11.3	Math 对象	147
11.4	string 对象	149
11.5	进入下一章	155
第十二章	JavaScript 事件	156
12.1	事件处理程序	156
12.2	新的事件处理程序	167
12.3	进入下一章	168
第十三章	高级议题	169
13.1	创建用户自己的对象	169
13.2	使用 cookie	173
13.3	性 能	176

13.4	脚本测试.....	177
13.5	在 JavaScript 中使用 Java Applet	177
13.6	属性是对象的数组元素.....	178
13.7	进入下一章.....	179
第十四章	使用 JavaScript 控制窗口和文档	180
14.1	window 对象	180
14.2	window 对象的方法	184
14.3	打开附属窗口.....	184
14.4	窗口配置.....	185
14.5	使用链接时的问题.....	188
14.6	关闭附属窗口.....	189
14.7	向窗口写入内容.....	190
14.8	从另一窗口写入内容.....	191
14.9	document 对象	193
14.10	控制背景颜色	197
14.11	创建“滑动显示”	198
14.12	进行选择	200
14.13	进入下一章	203
第十五章	使用带有框架的 JavaScript	204
15.1	框架的基础知识.....	204
15.2	框架是对象.....	207
15.3	frames 数组.....	207
15.4	改变框架的内容.....	210
15.5	调用其它框架中的函数.....	213
15.6	进入下一章.....	214
第十六章	窗体与 JavaScript	215
16.1	窗体对象.....	215
16.2	从窗体传递信息.....	216
16.3	不必传递引用.....	218
16.4	在窗体间传递信息.....	219
16.5	窗体的有效性验证.....	220
16.6	文本是否为数字.....	222
16.7	长度检查.....	224
16.8	进入下一章.....	225

第十七章 与用户通信	226
17.1 合并用户的信息	226
17.2 状态条消息	229
17.3 使用状态条消息	231
17.4 写入当前页面	232
17.5 进入下一章	234
第十八章 地区代码应用程序	235
18.1 地区代码应用程序与数据	236
18.2 数据在哪里	236
18.3 窗 体	239
18.4 函 数	241
18.5 Info()和 Help()——观看窗口信息	241
18.6 搜索数组	246
18.7 换行问题	254
18.8 进入下一章	255
第十九章 现成的脚本	256
19.1 寻找 JavaScript	256
19.2 添加时钟	257
19.3 滚动状态条消息	259
19.4 不要偷,要借脚本	261
19.5 进入下一章	261
附录	263
附录 A 关于 Online Companion	264
附录 B JavaScript 对象与数组	265
附录 C JavaScript 属性	280
附录 D JavaScript 事件处理程序	285
附录 E 保留字	287
附录 F 符号参考	288
附录 G JavaScript 颜色	290
附录 H 寻找更多信息	295
附录 I 本书所用的脚本索引	300

序 言

欢迎阅读这本介绍 JavaScript 的《Netscape JavaScript 指南》。本书是为非程序员,为数百万个 Web 开发者而编写的,这些开发者知道如何创建 Web 页面,久闻 JavaScript 的大名,并且想来看看 JavaScript 到底怎么样。

也许读者以前从未有过编程经验,但这并不能阻挡住您学习和使用 JavaScript。虽说 JavaScript 也有相当复杂的一面——许多最高级的 JavaScript applets 要求开发者具备程序员的专业技巧。然而,即使用户现在不用这些一本正经的专业技巧,也能够完成许多编程工作了。用户为了使用 JavaScript,不一定非得使自己变成一个程序员。

本书中,我们逐渐地引入了各种编程概念,这样做的目的,是为了让非程序员们开始使用 JavaScript 时,能够学到他们需要知道的内容。本书还提供了许多脚本,读者可以取出这些脚本,并把它们直接引入到自己的 Web 页面,然后稍作修改,就有了能够马上使用的短小 JavaScripts。可以在页面中键入这些脚本,或者到本书的 Online Companion(在站点 <http://www.netscapepress.com/support/javascript>)中,把这些脚本从 Web 页面示例中拷贝出来(本书的每个示例都有一个 Online Companion 引用号,它可以帮助读者找到正确的页面)。读者甚至可以把这些样本从 Online Companion 页面下载到自己的硬盘(这些页面是以 .ZIP、.SEA 以及 .TAR 形式提供的,因而可以一次把它们全部取出)。

0.1 准备工作

为了使用本书,读者需要准备几件东西。首先,必须要有一个兼容 JavaScript 的浏览器——就是说,一个具有执行 JavaScript 能力的 World Wide Web(全球信息网)浏览器。在本书写作期间,Netscape Navigator 2.0 或更新版本(请保持使用最新的浏览器,因为它将具有一些增强功能,还有对错误的修复)是有效的浏览器。可以从 <http://www.netscape.com/>和 <http://www.netscape.home.com/>下载该浏览器。阅读本书时,其它兼容 JavaScript 的浏览器可能已经出现,但要明白,本书是基于 Netscape 的。不同的浏览器显示 HTML 页面的方式略有不同,同样,不同的 JavaScript 浏览器在显示方式上肯定也会有略微差别。因为不同的开发组织会引发不同的错误和特征,这是不可避免的。

读者至少还需要了解一些有关 HTML 的知识。本书不是讲 HTML 的,因而假定读者已经知道如何创建 Web 页面——例如,在谈论 HTML 窗体,或者段落格式时,本书没有给出细节的描述。在本书讲到当用户单击按钮,如何使用 JavaScript 执行一个动作时,并没有完整解释其中有关窗体的知识。本书涉及了较宽的领域,而没有考虑这些基础知识,有时只是出于完成工作的要求,才解释这些所需的 HTML 信息。如果读者需要更多信息,则要参考有关 HTML 的书(例如 *HTML Publishing on the Internet*,作者为 Brent Heslop 和 Larry

Budnick, 由 Ventana 出版), 或者参考 HTML 的联机文档。

当然, 读者还要连接到 Internet 上, 虽然这一点并不是绝对必要的。一旦有了本书和一个浏览器, 就能在自己的硬盘上完成任何喜欢的工作。但是, 如果连接到 Internet 上, 就能访问 Online Companion, 也能观看 Web 中的所有 JavaScript 示例。

是否需要 JavaScript Developers Kit 呢? 在许多编程语言中, 为了创建程序, 需要一个特殊工具包, 其中包括了各种各样工具, 用来帮助用户使用该系统。但是, JavaScript 没有这种工具包。JavaScript 是 ASCII 文本——它不用编译成可执行程序文件, 可以在任何文本编辑器或字处理器中创建 JavaScript (当然, 要假定以 ASCII 文本形式保存它们)。实际上, 用户常常使用相同的应用程序来创建 JavaScript 和 Web 页面的源文档, 例如 Windows Notepad、HotDog、HTML Assistant, 或者其它应用程序, 因为 JavaScripts 只是写入 HTML 文档的附加指令 (然而, 请注意, 有些编写工具不接受程序无法识别的标记, 因此, 不能使用这些工具来创建包含 JavaScripts 的 Web 页面)。

0.2 内容简介

第一章是对 JavaScript 的介绍。该章解释了 JavaScript 是什么, 它与 Java 的区别, 并给出了几个用 JavaScript 完成的示例。

第二章让读者马上就使用 JavaScript。读者将学到一些使用 JavaScript 所能完成的简单工作, 例如, 把一个文档修改的日期添加到自己的 Web 页面, 或者在状态条中显示一条消息。本章没有花费太多时间确切地解释这些工作是如何进行的, 因为它们很简单, 读者或许可以猜出来。本章的目的是在进入更复杂的内容学习之前, 告诉读者在 JavaScript 中完成一些基本工作是非常简单的。

第三章开始真正地学习 JavaScript。本章讲了如何把脚本输入 Web 页面中, 函数是什么以及如何使用函数, 还有如何输入注释。

第四章继续学习 JavaScript。本章讲了 JavaScript 存放数据的方式, 即放在变量中, 还讲了如何使用字面量向这些变量中输入数据。

第五章讲了如何运用各种技巧来处理数值。

第六章解释了实际的编程方法——如何利用循环和条件表达式来使脚本做出实际的判定。

第七章重新讨论了函数, 介绍了使用函数的细节问题。例如, 读者将学习当调用函数时, 信息是如何传递给函数的。

第八章讨论了如何修正脚本中存在的一些问题, 脚本中肯定会存在问题的。本章还讨论了预先避免故障的办法, 以及怎样把脚本写得易于阅读和理解。

第九章讲了如何创建数组, 数组是能够拥有多个值的变量。本章还讲了如何匹配数组, 通过匹配, 可以按照另一个数组中的值而把信息从这一数组中取出。例如, 搜索姓, 然后从另一数组中把与之相应的地址取出。

第十章在面向对象的编程中是一堂比较容易的课。JavaScript 包含对象——事物, 每个,

事物都有它自己的属性,与属性相关的函数(叫做方法)用来“做事情”。

第十一章更详细地解释了 JavaScript 的对象,并涉及几个重要的对象。

第十二章讲解了如何使用事件处理程序控制脚本。当进行单击按钮、指向一链接,以及按 Tab 键等操作时,可以让脚本“做些事情”。

第十三章将步入 JavaScript 最深奥的领域——这里有各种非常高级的议题,甚至许多 JavaScript 程序员都从来不想接触这些议题。读者将会了解到 JavaScript 能够完成的其它工作的概况。

第十四章讲解了如何利用脚本打开附属窗口,并向这些窗口内写入文本或加载 html 文件;如何把确认框添加到链接上;如何让用户从一定的目标范围内做出选择,以及其它内容。

第十五章讲解了如何使用 JavaScript 在窗口中控制框架——例如,怎样在一个框架中执行某个动作,从而把内容写到另一个框架中。

第十六章描述了如何编写使用窗体的脚本。例如,从窗体中获得信息,再将这些信息传递到函数。还介绍了如何验证窗体中数据的有效性,以确保用户输入的是正确的数据。

第十七章介绍了通过消息来与用户通信的各种不同的方法。

第十八章汇编了前面学到的所有内容。本章创建了一个实际的应用程序,能够用来搜索 North American Numbering Plan(遍及美国、加拿大和加勒比海的许多岛国)中的地区代码。用户键入一个地区代码、城市、州/省/岛国名,程序就会搜索相关的信息。本章中将引导读者逐行地分析脚本,并且对程序的组成原理给出了详尽的解释。

第十九章介绍了把脚本添加到用户 Web 页面上的简便的方法——借用脚本!用户可以访问一些脚本库,从中取出一些脚本并把它们组合到自己的文档中。

本书还包括几篇附录。为了使用 JavaScript,必须了解许多技术细节,或者当读者需要时,应该知道从哪里找到这些细节。

附录 A 中的内容在下面的 0.3 节中详细讨论。

附录 B 是 JavaScript 对象的概要。本附录列举了全部对象,并介绍了每个对象所具有的属性,以及可用于每个对象的方法和事件处理程序,而且,在某些例子中,还说明了父对象——某个被描述的对象是其父对象的一个属性。

附录 C 是 JavaScript 全部属性的概要,介绍了每个属性的用途,并说明了该属性可用于哪些对象。

附录 D 介绍了 JavaScript 事件处理程序,并说明了在什么条件下使用它们。

附录 E 是保留字的简短列表。保留字是用户在声明变量、定义函数,或者创建对象时不能使用的单词。

附录 F 列举了出现在 JavaScript 中的全部符号,对每个符号的功能都给出了一个简短的概括。

附录 G 是一份用于 JavaScript 中的全部颜色的列表,还给出了与各种颜色相关的十六进制代码(然而这些代码很少使用——只用颜色的名字就行了)。

附录 H 介绍了有关 JavaScript 的高级信息的来源。学完本书之后,读者可以到 JavaScript 新闻组、Web 页面、邮件列表等等之中继续学习。

附录 I 是一份把本书中的全部脚本按照用途排列的列表。读者可以查看某项技巧,例如,导航按钮、状态条消息等等,以及与它们所在脚本相关的 Online Companion 索引号,因而,可以把脚本从本书的 Web 文档中拷贝出来。

0.3 关于 Online Companion

本书有一个配套的 Online Companion,它是一个位于 Ventana Web 站点的 Web 页面的特殊集合。这些 Web 页面包含了帮助读者使用本书的信息,并让读者在学完本书之后,继续学习有关 JavaScript 的知识。本书中的所有示例都可在 Online Companion 中找到——读者可以先阅读书上的示例,然后到 Online Companion 中试用实际的脚本。也可以把所有的 Web 页面下载并保存到自己的计算机硬盘上,这样,即使退出 Internet,也可以使用这些脚本。

除了一些样本脚本之外,还能找到几个指向其它有用的 JavaScript 站点的链接。在这些站点中能找到更多的样本、新闻发布、指导,以及关于 JavaScript 的最新信息。我们已经编写了一个书签文件,因而可以把所有链接从附录 H 拷贝到用户的书签系统中。这里还有一张颜色表,向用户展示了各种可用颜色——单击一个颜色,页面的背景颜色就会改变成那种颜色。Online Companion 还提供了 Object Hierarchy 页面,帮助用户理解和学习 JavaScript 对象层次结构。

把 Web 浏览器指向 <http://www.netscapepress.com/support/javascript>, 然后就能到达 Online Companion。

0.4 JavaScript 是活的

像大多数的编程语言一样,JavaScript 的变化也非常快。大多数编程语言在发布之后都是固定不变的,因而,直到下一个版本发布之前,用户无须担心变化。但是,JavaScript 差不多是个活生生的事物,一旦写出脚本之后,脚本运行的方式就不再是死板的了。不同浏览器解释脚本的方式会略有不同,甚至同一浏览器的不同版本间都会有差别。我们使用的是 Netscape Navigator 2.01 系列浏览器,但是也包括 Navigator 3.0 beta 3 和 beta 4。到编写本书时,beta 4 是最新的有效浏览器。在这两个版本间存在一些重要差别。

技巧: 创建 JavaScript 时,不要走到最前沿,而要慢上一步——或者两三步。若是用最新的 JavaScript 功能来编写脚本,则这种脚本所在的 Web 站点观察者们无法使用这些功能,甚至这些新功能还会导致问题的发生。

我们建议使用最新版本的 Netscape,以便确保能够得到 JavaScript 的“最好的”版本。毫无疑问,Netscape Navigator 的后续版本使用 JavaScript 的方式会略有不同。

读者或许会发现书上讲的一些内容与实际的运作略微不同,如果发生这种情况,请告诉我们。可以这样来同我们联系:

71601.1266@compuserve.com(Peter Kent)和
100553.1346@compuserve.com(John Kent)

我们将用所发现的任何改变来更新 Online Companion。

现在就让我们开始吧！先来慢慢地解释一下 JavaScript 是什么，并展示几个 JavaScript 高级应用示例，然后告诉读者如何能够迅速地使用 JavaScript 把一些简单的内容加入自己的 Web 页面。好了，现在就请翻到第一章，让我们开始学习吧！

第一章 JavaScript 是什么

在 Web 上工作了一两个多星期的用户,很可能会碰到“JavaScript”和“Java”术语。它们是什么,用户拿它们又能做什么?如果买了此书就会得知,使用 JavaScript 和 Java 能够创建从 Web 浏览器上运行的小程序。然而,在开始学习 JavaScript 之前,先让我们花些时间看一看 JavaScript 是什么,它与 Java 的比较,以及用它能做些什么。

1.1 首先有了 Oak

先要从 Oak 说起。故事开始于 World Wide Web(WWW)诞生前的 1990 年。当时,Sun Microsystems 公司组建了一支特殊的小组。这个小组从公司的其它部门分离出来,具有相当大的自主权和一定程度的秘密性。他们的目标是要运用曾经激励过 Apple 和 NeXT Computer 等公司的灵感和首创精神(这些品质正是许多人感到 Sun 公司所缺乏的)来创建一种新的产品:一种能运行在所有东西上的新操作系统,甚至包括那些人们通常不会想象到需要软件的东西:冰箱、电视、收音机、烤面包器、门锁等等。他们认为下一次计算革命浪潮应该是消费性电子产品的扩张。那时,软件将进入每一个电子产品,而他们的新操作系统更是首当其冲;另一方面,这种操作系统必须具有可靠性,像将要使用它的电子器件一样可靠。一些高级别的内部程序错误对许多 PC 软件来说已经可以作为常规接受,而对消费性电子产品来说则是不允许出现的。

如果所有这些器件都使用同样的软件,如果这种软件非常灵巧,能让所有这些器件都容易使用,生活将会更加简单。因为使用不同器件的方法都是相通的,所以用户不再需要面临各种不同的使用方法。这个小组开始工作时使用的语言叫做 Oak。他们也装配了一个电子器件,成为第一台运行 Oak 的设备:一块从微型电视上取下的 LCD 屏幕,从 Nintendo's Game Boy 取下的扬声器,前端有一块触摸屏,以及压缩到微型内壳中的 Sun 工作站的配置等等。

然而,对于消费性电子产品而言,Oak 不仅仅是一种操作系统,它还能链接所有东西。所有电子器件都能联接入网,Oak 则是把它们粘接在一起的电子万能胶。所有的电子设备和数字设备都能通过 Oak 运行。Sun Microsystems 公司对 Oak 感到非常振奋,因而创立了一个新公司 FirstPerson 来开发和推进这个系统。

但是 Oak 根本没有找到自己的家。有些公司对它表示出兴趣,例如 France Telecom 想把它用于自己的 Minitel 数据系统上;Mitsubishi 想用于本公司的电子器件上;Time Warner 想用于其交互式电视控制盒上,等等,但都没有付诸实施。或是因为设备过于昂贵,或是因为那些有潜力的客户最后买了其它公司的产品。

最后,FirstPerson 回到了现实基础上,他们决定忘掉消费性电子产品而集中精力于个人电脑,然而他们已经无法跟上 Sun 工作站要求的步伐。因此,1994 年的年初,FirstPerson 步

履维艰。

然而,数月之后,作为 Sun 创始人之一的 Bill Joy 使得 Oak 重新崛起。他认识到 Oak 能用于 Internet 上,并计划让 Sun 继续开发 Oak 并将它公布。他说:“让我们给予它特许权。”换句话说讲,就是把 Oak 公布于众,让足够多的人使用它,那么 Sun 最终将“拥有”这个特殊的市场。1995 年初,Oak 按照 Internet 上的使用要求进行了修改,定名为 Java。

于是开发了一个特殊的程序 HotJava,它能够实际地运行用 Java 创建的程序。当然,后来将 Java 解释器集成到 Netscape Navigator 中,也就是说,Navigator 本身可以运行 Java 应用程序——Java applets。随着 applets 的普及,HotJava 成为 Navigator 用户的多余之物。

1.2 Java 到底是什么

Java 是一种编程语言,某种程度上像 C++,但却倾向于具有更高的可靠性。别忘了,Java 最初是为消费性电子产品市场设计的,那里比消费性软件市场更难容忍不可靠性。

Java 能允许用户创建程序,但却需要一个操作系统来运行程序。一般说来,如果要创建一个字处理器,分别运行于 Macintosh 机器和 Sun 工作站上的 DOS 和 Windows 系统,则需要创建四个不同版本(至少四个,也许更多:一个用于 Windows 3.1,一个用于 Windows 95 等)。但是 Java 却不同,Java 的观点是程序员只创建一个版本的程序,然后在 Java“解释器”(一种例如 HotJava 或 Netscape Navigator 的程序)上运行。不同的解释器用于不同的计算机和操作系统,例如一种 Netscape Navigator 运行于 Macintosh 机上,一种运行于 Windows 3.1,一种运行于 Windows 95 等。这样,一个 Java 程序就能运行于一组不同的操作系统上,只要用户拥有一个将运行于自己计算机中的解释器。

这些 Java 解释器并非真正的操作系统,虽然其原理从某种程度讲是相同的。操作系统是计算机硬件与程序之间的接口,程序员不用直接对键盘、视屏等计算机硬件编程,而只须对操作系统编程,让操作系统完成与硬件“对话”的例行工作。Java 解释器以同样原理工作。它位于操作系统与 Java 程序之间。程序员不再需要对不同操作系统编写不同的程序版本,现在只须编写一个单一程序,就能与任何计算机系统上的任何 Java 解释器对话。Java 解释器在 Java 程序与操作系统之间建立联系,将一方的语言翻译成另一方可以理解的语言。

Java 的支持者们声称这是一项突破性技术进步,因为它打破了操作系统强加于软件的束缚。“解释型”语言其实已经存在了一段时间,但它们都不像 Java 那样成熟。Java 是一种强有力的编程语言,使用它可以创建十分成熟的程序,并且将其运行于各种不同的计算机上。一个程序不再仅仅是 Macintosh 程序、Windows 程序,或者 UNIX 程序,现在它叫 Java 程序。只要有可用的 Java 解释器,它就能运行于任何这些系统上。

Java 要把我们带到哪里

许多人预言 Java 是结束“压缩式包装”软件的起点。最终,用户将使用用 Java、JavaScript 和类似的语言所创建的程序。用户不用购买字处理器或电子表格并把它们加载到自己的硬盘中,而只须登录到 Internet 站点,并从那里运行程序,然后按照使用时间或次数付费。