

用 Java 小应用程序
丰富您的 Web 主页



〔美〕 David Gulbransen 等著
曹道卿 哈勤和 译

用 Java 编写 Web 小应用程序



浙江科学技术出版社
西蒙与舒斯特国际出版公司

David Gulbransen: CREATING WEB APPLET WITH JAVA

Authorized translation from the English language edition published by WAITE GROUP PRESS

Copyright1997 by Sams.net Publishing

All rights reserved. For Sale in Mainland China Only

本书中文简体版由浙江科学技术出版社和美国西蒙与舒斯特国际出版公司

合作出版, 未经出版者书面许可, 本书任何部分均不得以任何方式复制或抄袭。

本书封面贴有西蒙与舒斯特防伪标签, 无标签者不得销售。

版权所有, 翻印必究。

浙江省版权局著作权合同登记号 图字: 11-1998-09号

书 名	用 Java 编写 Web 小应用程序
著 者	[美] David Gulbransen 等
译 者	曹道卿 哈勤和
出 版	浙江科学技术出版社 西蒙与舒斯特国际出版公司
印 刷	杭州富春印务有限公司
发 行	浙江省新华书店
制 作	浙江科学技术出版社计算机图书工作室
读者热线	0571-5157523
电子函件	hzzjkj@public1.hz.zj.cn
开 本	787 × 1092 1/16
印 张	13.75
字 数	323 000
版 次	1998 年 8 月第一版
印 次	1998 年 8 月第一次印刷
书 号	ISBN 7-5341-1146-3/TP · 47
定 价	24.00 元
责任编辑	熊盛新
封面设计	金 晖

译者序

Java或许是当今计算机世界中被人们讨论得最多的一件新生事物了。毫无疑问，Java从它诞生的那一天起就给计算机行业吹来了一股清风；它得到了世界上几乎所有有影响的大公司的全力支持；它带来了很多新鲜而有趣思想和观念；它改变了国际互联网络和网络应用软件的面貌；它甚至改变了人们使用计算机的方式。作为一种全新的、跨越操作平台的和面向对象（OOP）的程序设计语言，Java在很大程度上平息了计算机业界对何种操作系统将最终统一计算机操作平台这一问题的喋喋不休的争论。当然，它也带来了另一类争吵，那就是关于PC（个人电脑）还是NC（网络电脑）代表着网络世界中未来电脑发展方向的争论。但不管怎么说，所有这些讨论都是有益的，因为这就意味着在信息时代中的我们能够拥有比以前更多、更好的选择；意味着我们能够接触到更丰富、更激动人心的多媒体网络资源；意味着我们可以以自己的方式和角度去看待这个精彩纷呈的世界中的各种细微变化。

谢谢Java！

本书为读者了解Java提供了一条捷径，它不是一本纯粹的关于Java编程语法的书籍，它也不同于普通的用户编程手册，它是以一个个由简到繁、由易到难的编程例子串联而成的，是引导你进入Java世界的阶梯。毫不夸张地说，任何对Java或者对计算机感兴趣的人都可以从本书中得到帮助，无论他（她）是初学者或是资深的专家。

本书的文字录入工作由浙江教育出版社的华琼小姐和杭州大学外语系的田敏捷小姐帮助完成，浙江大学的廖江波、张辉和施振展还参与了本书的翻译工作，对他们辛勤而卓有成效的工作，无论表示什么样的感谢都不过分！

由于时间仓促，也由于Java是一个全新的事物，书中难免存在谬误，恳请各位读者朋友指正。

译者

1998年5月于浙江大学求是园

目 录

第 1 章 什么是 Java	1
1.1 Java 简史	1
1.1.1 分布式计算	3
1.1.2 面向大众的编程语言	5
1.1.3 对 象	5
1.1.4 Java 适合你	7
1.2 与平台无关的方案	7
1.2.1 适应一切的二进制编码	8
1.2.2 与平台一致的用户界面	8
1.2.3 交互式的站点主页	8
1.3 小 结	11
第 2 章 使用 Java	12
2.1 小应用程序——Web 功能的扩展	12
2.1.1 在你的站点主页上增加声音	12
2.1.2 增加动画和图像	13
2.1.3 交互性	14
2.1.4 真正的用户界面	15
2.2 使用 Java 的一些限制	16
2.2.1 不能访问文件	16
2.2.2 小应用程序没有记忆能力	16
2.2.3 Java 的速度	17
2.2.4 连接的一些限制	17
2.2.5 网上安全性	18
2.3 独立的应用程序	18
2.4 Java 的未来	19
2.4.1 新的开发工具	19
2.4.2 提高速度	21
2.4.3 与其他应用程序集成	22
2.4.4 小应用程序的自动升级	22
2.4.5 基于 Java 的脚本语言	22
2.4.6 Sun 微系统公司的 Cosmo 计划	22
2.4.7 Java 虚拟机	23
2.5 小 结	23
第 3 章 Java 浏览器	24
3.1 Hot Java	24

3.1.1	用 Java 语言编制的 Java 浏览器	24
3.1.2	特色丰富、灵活性高	25
3.1.3	动态文件类型和协议	25
3.1.4	系统配置	26
3.1.5	安全性	27
3.1.6	Sun 微系统公司有关 Hot Java 的开发策略	28
3.1.7	提供 Hot Java 软件的网址	28
3.2	小应用程序浏览器	29
3.2.1	特殊功能	29
3.2.2	系统配置	30
3.2.3	安全性	30
3.2.4	提供小应用程序浏览器的网址	30
3.3	Netscape 公司的网络导航者软件	31
3.3.1	特殊功能	32
3.3.2	添加新标准	33
3.3.3	系统配置	33
3.3.4	安全性	35
3.3.5	提供 Netscape 网络导航者软件的地址	36
3.4	小 结	36
第 4 章	Java 的功能	37
4.1	增强主页的视觉和听觉效果	37
4.1.1	时钟小应用程序	37
4.1.2	杂耍小应用程序	40
4.1.3	动画小应用程序	55
4.1.4	幻灯片小应用程序	57
4.2	增强主页的功能	58
4.2.1	曲线小应用程序	58
4.2.2	股票跟踪小应用程序	58
4.2.3	远程传输小应用程序	59
4.2.4	交互式个人电脑分值牌小应用程序	60
4.2.5	自动钢琴演奏者小应用程序	61
4.3	小 结	62
第 5 章	查找和使用小应用程序	63
5.1	主要的万维网站点	63
5.1.1	Sun 微系统公司站点	63
5.1.2	Gamelan 公司站点	64
5.1.3	JARS 站点	66
5.1.4	查找引擎	68
5.2	源码或二进制代码	68
5.2.1	使用二进制代码	68

5.2.2 使用源代码	71
5.3 版权问题	72
5.4 小 结	72
第 6 章 具有 Java 特色的主页设计	73
6.1 HTML 和 Java	73
6.1.1 理解<applet>标记	74
6.1.2 用参数使小应用程序客户化	76
6.1.3 组合 HTML 和 Java	77
6.2 设定要点	78
6.2.1 美学设计	78
6.2.2 功能设计	80
6.2.3 声 音	81
6.2.4 避免过多的小应用程序	83
6.2.5 下载时间	83
6.3 小 结	83
第 7 章 Java 开发者工具箱	84
7.1 Java 开发者工具箱中的组件	84
7.1.1 Javac	84
7.1.2 java	85
7.1.3 jdb	86
7.1.4 javah	87
7.1.5 javap	87
7.1.6 javadoc	87
7.1.7 小应用程序观察器	89
7.1.8 小应用程序例程	90
7.2 JDK 的站点	91
7.3 JDK 的安装	92
7.3.1 在 Unix 系统下安装	92
7.3.2 在 Windows(NT 或 95)系统下安装	93
7.3.3 在苹果机系统上安装	93
7.4 查找 JDK 文档	93
7.4.1 Java 工具文档	93
7.4.2 Java 的 API 文档	93
7.5 如何编译 Java 源程序代码	94
7.6 小 结	96
第 8 章 Java 的语法	97
8.1 注 释	97
8.2 变 量	98
8.2.1 整 数	98
8.2.2 浮点数	99

8.2.3 字 符	99
8.2.4 布尔值	100
8.3 表达式	100
8.3.1 算术运算符	100
8.3.2 关系和逻辑运算符	101
8.4 数 组	103
8.5 语句和程序块	104
8.6 流程图	104
8.6.1 If...Else 语句	104
8.6.2 循 环	105
8.7 无指针的语言	107
8.8 小 结	108
第 9 章 Java 的对象	109
9.1 面向对象的程序设计的基础	109
9.2 类和对象	110
9.3 生成自己的类和对象	111
9.3.1 Vehicle 实例	111
9.3.2 实例变量	111
9.3.3 封 装	112
9.3.4 方 法	112
9.3.5 构造函数	113
9.3.6 创建对象	114
9.3.7 访问方法	114
9.3.8 一个小应用程序例程	115
9.4 继 承	117
9.4.1 函数的继承	118
9.4.2 继承中的多态	119
9.5 小 结	123
第 10 章 小应用程序的结构与设计	124
10.1 小应用程序软件包	125
10.1.1 getDocumentBase() 方法	127
10.1.2 getAudioChip(URL, string) 方法	127
10.1.3 getImage(URL, string) 方法	127
10.2 小应用程序的生存期	128
10.2.1 init() 方法	129
10.2.2 start() 方法	130
10.2.3 stop() 方法	130
10.2.4 destroy() 方法	130
10.3 建立一个小应用程序	131
10.3.1 参 数	133
10.3.2 完成小应用程序	134

10.3.3 将程序片段结合起来	135
10.4 小 结	137
第 11 章 建立用户界面	138
11.1 AWT 介绍	138
11.1.1 一个简单的例子	138
11.1.2 不要担心: 使用 AWT 的一些实际建议	140
11.2 事件处理	140
11.2.1 增加基本事件处理	141
11.2.2 使用事件处理方法	143
11.3 控 件	145
11.3.1 界面控件	145
11.3.2 所有控件的通用方法	147
11.3.3 容 器	149
11.3.4 页面布局	150
11.4 设计用户界面	151
11.5 小 结	154
第 12 章 自动显示带小应用程序	155
12.1 设计要点	155
12.1.1 指定消息	155
12.1.2 指定字体	156
12.1.3 改变动画速度	156
12.2 开始程序的第一步	156
12.2.1 文本书写的精细控制	158
12.2.2 增加字体控制	158
12.3 精确地定位文本	159
12.4 文本的动画	160
12.5 增加参数	166
12.6 小 结	169
第 13 章 幻灯片制作和播放小应用程序	170
13.1 设计要点	170
13.1.1 文件支持	170
13.1.2 用户参数	171
13.1.3 用户界面	172
13.1.4 设计小应用程序的启动方式	172
13.2 编辑小应用程序	173
13.2.1 分析参数	174
13.2.2 载入图像和媒体跟踪器	175
13.2.3 产生用户界面	176
13.2.4 画 图	178
13.2.5 自动播放线程	179

13.2.6 start()方法与 stop()方法	180
13.3 小 结	187
附录 A 万维网上的 Java 资源	189
A.1 Sun 微系统公司的 Java 站点——Javasoft	189
A.2 Gamelan 公司站点	189
A.3 JARS 公司站点——Java 小应用程序的分类搜寻服务中心	190
A.4 查找引擎	190
A.5 文件里的资源	190
附录 B Javascript 和 Java 语言参考	192
B.1 保留字	192
B.2 注 释	192
B.3 常 量	193
B.4 变量声明	193
B.5 变量赋值	194
B.6 运算符	194
B.7 对 象	195
B.8 数 组	196
B.9 Java 中的条件和循环	196
B.10 类的定义	196
B.11 方法和构造函数的定义	197
B.12 包和接口	197
B.13 出错与异常情况处理	198
附录 C 50 个实用的 Java 小应用程序	199
本书术语	204

第1章 什么是 Java

自万维网(WWW)诞生以来, Java 可能是在 Internet 技术中讨论得最多的取得重大进展的事物之一了。计算机界的各行各业的人们都在讨论 Java 以及它将如何改变计算机世界的面貌。毫无疑问, Java 将改变人们使用 Internet 和网络应用软件的方法。Java 给万维网领域带来了许许多多新的思想, 这些思想预示着重大的 Internet 和计算机行业将发生最激动人心的变化。

由于对 Java 的过分渲染, 使一些人产生了迷惑。我们将在本章精确地解释什么是 Java, 什么不是 Java。我们将讨论 Java 的一些基础知识, 有了这些基础知识, 你就能对运行在 Internet 上的程序有比较清晰的了解, 并能够运用 Java 来开发这些程序。由于 Java 的许多功能是一系列有趣的开发过程的结果, 所以, 本章也介绍了 Java 的开发史, 你将了解 Java 的过去、现在和将来。

我们讨论 Java 话题的中心内容是: Java 是一种程序设计语言。Java 是一种面向对象的程序设计(OOP)语言, 它使用了其他许多面向对象程序设计语言的一些通用元素, 比如 C++, 并且它可以使程序设计变得更加容易。像其他程序设计语言一样, Java 有自己独特的语法和程序结构, 它也支持可执行的应用程序。

JDK(Java 语言开发工具箱)包括所有的开发应用程序所必需的工具, 这些工具是使用 Java 来编程的, 它们包括:

- Javac: Java 编译器
- Jdb: Java 调试器
- Javadoc: Java 文件程序
- Java: Java 虚拟机
- Appletviewer: Java 小应用程序观察器

其中的一些 Java 控件可能不太直观, 如 Java 虚拟机, 在 Java 应用程序中, 我们不能直接访问 Java 虚拟机。然而, 一些可运行 Java 应用程序的商业性站点浏览器可以使用 Java 虚拟机, 这个使用过程同应用程序观察器的使用过程是一样的。这些元素互相关联(如图 1.1 所示), 它们都围绕着 Java, 我们可以在 WWW 上使用这些元素。

1.1 Java 简史

Java 的前身是一种名为 Oak 的语言, Oak 作为一种新语言诞生于 1990 年。当时, Sun 微系统公司成立了一个代号为“绿色”(Green)的项目开发小组来开发新产品, 以拓宽 Sun 微系统公司的市场, 他们的第一个目标是试图开发一个面向家用电器市场的软件产品。Oak 开始是为一个名叫“*7”的个人数字助理软件而设计的, Sun 微系统公司用该软件来开拓无缝图形用户界面市场。

然而，“*7”没有市场，最后，Sun 微系统公司成立了一个名为“FirstPerson”的公司，其目的是把“*7”开发成为交互电视行业的首选软件。由于诸多因素的影响，交互电视的前景很快变得暗淡，Oak 失去了其市场。就在 FirstPerson 公司和 Oak 语言失败的同时，万维网正在迅猛发展，尤其是 Netscape 公司制造的网景导航员（Navigator）成为 Internet 发展的强大动力之一。Sun 微系统公司认识到，Oak 在 Web 站点上的发展机会更大。很快，Oak 就以一个新的名字——Java 在 Internet 上发布了。

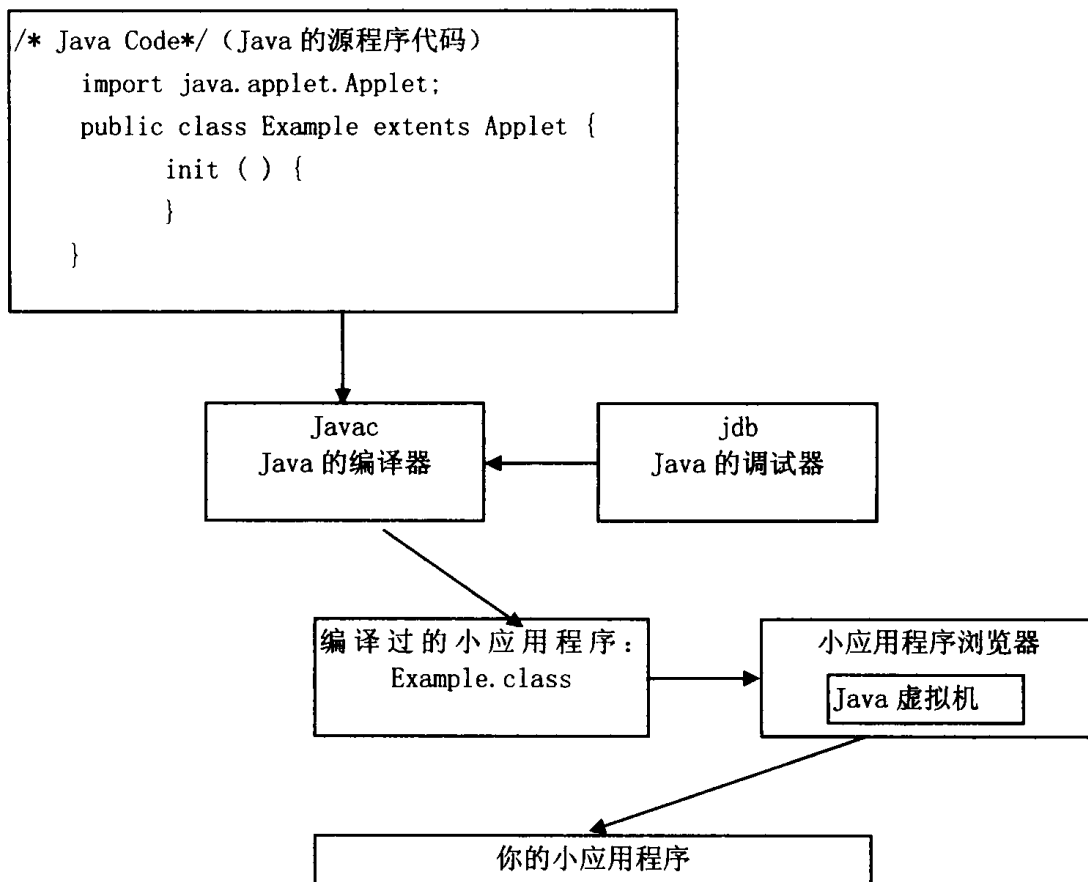


图 1.1 Java 的各种组件之间的相互关系

一开始，Java 是以命令行开发环境的形式发布的，但它已经开始影响 Internet 和计算机业的发展方向了。Java 程序语言在 Internet 上是免费发布的(如图 1.2)。为了尽快建立 Web 站点的程序设计标准，Sun 微系统公司把 Java 的全部功能和组件授权给大批的 Internet 软件销售商。

为了更好地理解什么是 Java，我们有必要进一步了解一些有关编程环境和操作的通用组件。由于了解一些面向对象程序设计的基础知识及这些知识如何影响 Java 的产生是很重要的，所以，本书不打算在一开始就介绍如何编写 Java 程序代码。下列部分概述了一些程序设计的要点，这样你就可以为理解后面的章节打好基础。

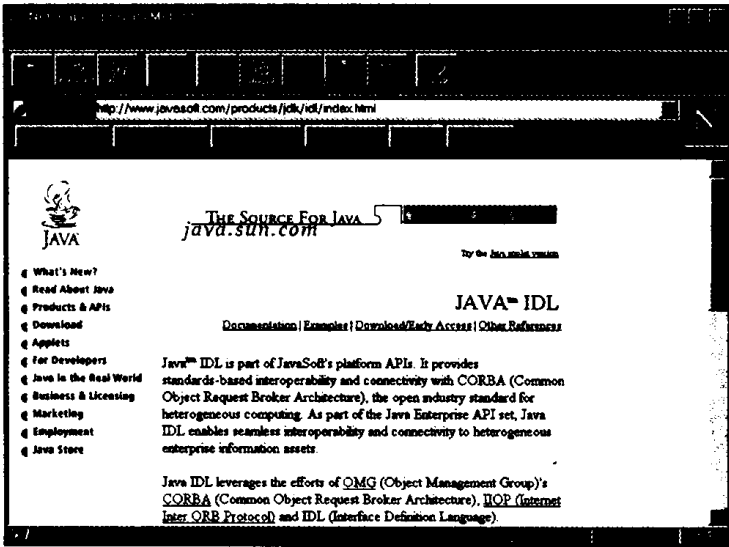


图 1.2 Sun 微系统公司的 Java 站点
(网络地址: <http://java.sun.com>)

1.1.1 分布式计算

Java 给计算机界带来了很多方面的革新，其中最重要的是 Java 改变了我们使用计算机的方式。Java 是围绕着网络和连接的概念设计的。在当今计算机世界中，我们都在硬盘中运行程序，你需要不断地安装软件、使用软件，在软件过期时将其升级。如果你需要把你的计算机同别人的计算机相连，就必须装载一个应用程序，如 Web 浏览器或 e-mail 软件，这些应用程序可以和网络连接。但试想一下，如果我们以另一种方式来运行我们的程序：你的应用程序存储在网络上，每次当你使用该程序时就从网上下载它。在这种模式下，你的软件总是最新的，你的计算机不断地同网络保持连接(如图 1.3)。分布式计算就是通过网络来共享应用程序资源。

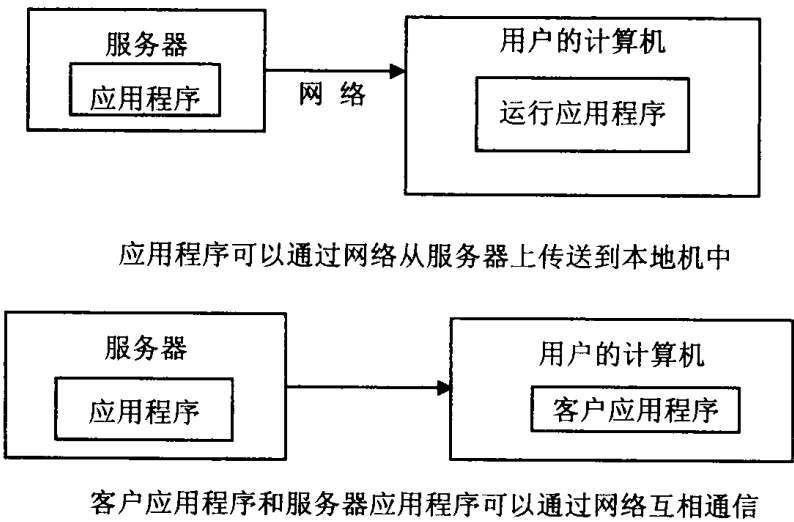


图 1.3 计算机应用程序传播模型

Java 的最终目标是能够创建随处可用的网络应用程序。由于当前数据传输的带宽和其他网络条件的限制,这个目标还很遥远。但是,Java 正在为网络应用程序(如 Web 浏览器)增添新的功能并使它更富有趣味性。我们可以用 Java 编写小应用程序并把它们合并到 Web 站点上,以运行主页。比如,图 1.4 所示的字符自动显示带(tickertape)小应用程序可以在一个 Web 主页上滚动字符。

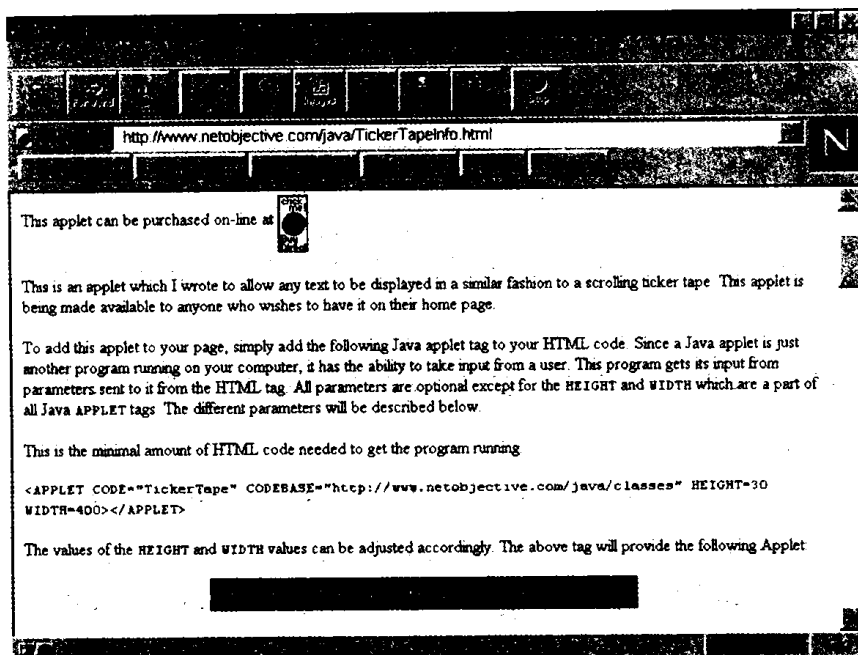


图 1.4 文本滚动条小应用程序能够以一种醒目的方式表达信息

Java 的安全性问题

Java 包含安全功能以增强它在 Internet 上的应用。安全性问题会潜在地影响 Java 小应用程序。Java 使用代码校验和有限制地访问文件系统的方法来保证代码在任何时候都不会破坏本地机器系统。

当一个小应用程序下载到本地机器上时,系统将检查该代码以保证该小应用程序仅仅使用有效的系统调用和方法。如果该小应用程序通过了校验,它将被允许运行,但它可能没有访问本机文件系统的权利。有限制地访问文件系统使小应用程序有了一定的局限性,但也防止了小应用程序意外地删除本地机上的文件、滋生病毒或造成其他严重的破坏。

分布式应用程序和小应用程序的核心是网络通信。Java 是一种与 Internet 进行集成的语言。作为一种程序设计语言,Java 已经被设计为在 Internet 或其他大规模网络上使用,所以,它包括大量的网络库以支持 TCP/IP 网络协议和许多其他网上通用的协议,如 HTTP 或 FTP。这一功能意味着你可以很容易地将网络功能集成到你自己的应用程序中去。其最终的结果是:网络的功能变得更广泛,应用程序可以更多地使用网络。

小应用程序的思路对分布式软件模式非常适合。一个小应用程序实质上是一个可分离

的应用程序。可执行的应用程序可以非常大，通常情况下，应用程序拥有一系列的支持文件和配置文件，它们需要有序地运行。所有这些文件导致了許多非常庞大而复杂的程序，它们在网络上传播很困难。小应用程序的文件很小，移植方便，可以下载到主页上，可以增强主页的功能，具有交互性，并可增加有趣的图形和声音，使 Web 站点更吸引人。由于小应用程序通常很小，它们不会占用很多的主页下载时间。而且，由于小应用程序实际上是程序，它们比传统的 HTML（超级文本标记语言）和 CGI（通用网关接口）脚本语言更加灵活。

1.1.2 面向大众的编程语言

Java 主要是建立在 C++ 基础上的。C++ 是一种适用于多平台的最通用的编程语言，是一种高级的、面向对象的编程语言。C++ 编程的难度很大，主要应用于高级商业应用，所以，C++ 语言的功能极为强大。为了优化程序的性能，C++ 能让编程人员控制计算机的最低层的部分。但 C++ 所提供的控制功能通常对中等大小的应用程序是不必要的，且对于没有经验的编程人员来说，C++ 是很难掌握的。

比如，C++ 使编程人员能直接控制内存的操作，这意味着，编程人员必须非常小心地给应用程序分配内存——如何使用某段内存，以及当应用过程结束时把它释放掉。内存分配和“块回收”经常在 C++ 程序中导致一些非常复杂和耗时的错误。幸运的是，Java 代替你处理了这些内存处理的细节，这就可以为你节省许多调试程序的时间，而调试应用程序对程序设计人员来说，是一件非常繁琐的事。

自动内存处理对于编程人员来说是一件大喜事。如果你问任何一个用 C++ 编程的软件人员：C++ 编程最困难的部分是什么？他们会回答是“指针”。指针是 C++ 用来跟踪内存的定位和存储内容的方法。想象一下，Java 必须跟踪数据在内存中的位置并且决定如何去存取它们，这听起来真令人感到不可思议，但 Java 的确做到了这一点。Java 是一种不使用指针的计算机程序设计语言。

1.1.3 对 象

近年来，面向对象的程序设计语言在程序语言领域中不断发展壮大。掌握对象这种概念并不难。对象是一些小的源代码，用来完成某种特定的功能。软件人员可将它们组合起来成为一个完整的程序。

比如，一个小程序可能包含输入对象、处理对象和输出对象。对于一个计算器的小应用程序来说，输入控制键可看作输入对象，它可以接收用户的输入，并把它传给处理器对象，由处理器对象完成计算任务，接着，处理器对象将结果传给输出对象，输出对象可以向用户显示结果（如图 1.5）。对象可使程序设计更加灵活、方便。

虽然面向对象程序设计的基础概念很简单，但如何操作还是一项很复杂的任务。编程人员需要花大量的时间来规划开发小应用程序的过程，保证每个单元组件使用方便以及它们相互之间工作协调。设计良好的对象可使程序在长时间的使用过程中更加轻松、容易，所以在开始的开发计划中需要对它们进行周密的规划。面向对象程序设计方法给 Java 的发展带来了希望。

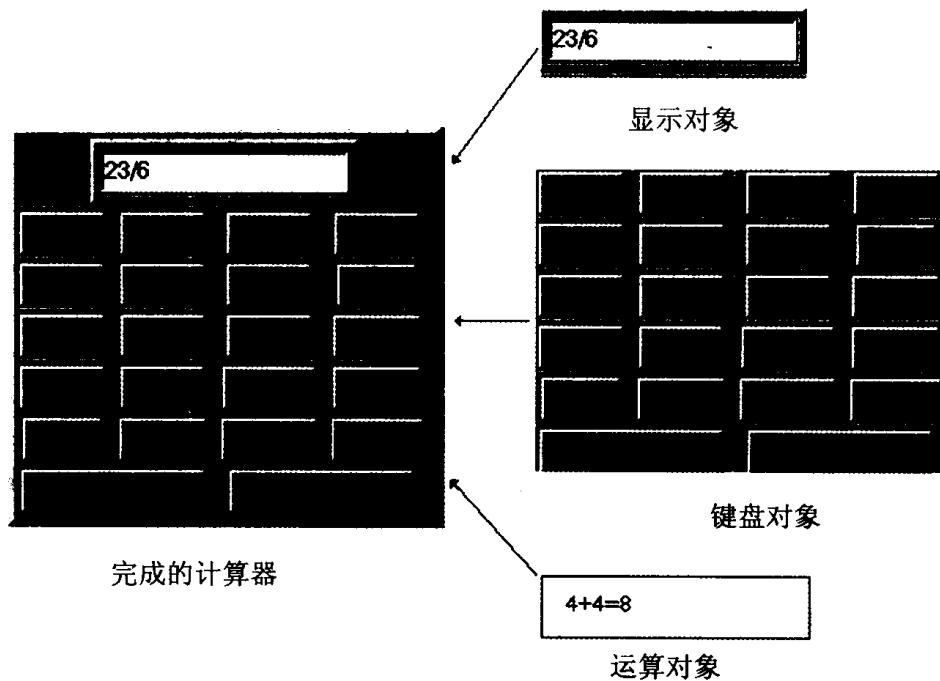


图 1.5 把一个计算器小应用程序分解成对象控件

当 Java 逐渐普及时，许多对象可以组成一个统一软件，它们可以在你自己的应用程序中得到使用。比如，某些人可能已经成功地开发了一个刻度盘组件，它可以像刻度旋钮那样让你用刻度盘设置值，这个刻度盘将把刻度的位置转换成某个程序中所用到的值。如果这个对象设计良好，并有很好的使用说明，你就可以把它集成到你自己的小应用程序中，根本不用费力从头开发它，你所要做的只是从刻度盘对象中取值。这种与程序无关的功能就是面向对象的程序设计的最终目的和最大的优点。

我们已经知道，用 Java 是可以进行编程工作的。就像学习其他编程语言一样，我们可以参照一定的资料，花一定的时间来学习 Java。幸运的是，Java 拥有高级程序语言的功能，但却没有 C++ 编程语言那样难以理解和掌握。请记住：Java 是一种建立在某些更为流行的程序设计语言基础上的程序设计语言，比如 C++ 和 C 语言。Java 保留了这些语言的功能，并尽可能地提高它们的可操作性。

Java 对象的设计也使得程序设计更加容易。Java 的一切是建立在对象模型基础上的，甚至那些可运行的小应用程序也是对象。Java 特别保留了这些功能以使用户在长期的应用开发过程中加快 Java 的开发速度。

上面的计算器小应用程序例子中，有一个数字控制输入键对象。假设你现在需要编制一个电话小应用程序。由于电话控制键同计算器的控制键相似，因此，你可以把计算器的控制键对象取出来放到电话小应用程序中去。只需作一些小的修改，你便可以重新排列这些键，使它们看起来像一部电话。这种方法使你能利用一个已开发的对象，将它修改以满足自己编程的需要。对象可替你节省大量的时间，通过重新使用组件，你可以更快地创建小应用程序。

1.1.4 Java 适合你

所有这些 Java 功能上的改进使它成为一种能够满足你工作需要的编程语言，但这些改进也带来了一些不足：

(1) Java 的内存管理使小应用程序的运行速度降低。但是，即使 Java 不能保持 C++ 的原始速度，也并不意味着它是一种速度很慢的语言。虽然一些高层次的应用，像 3-D 或 CAD 软件包的开发更加倾向于 C++，我们也没有必要因为 Java 的速度有些慢而不采用它。

(2) 面向对象的程序设计模式使软件设计规划更加复杂。但是，如果在开始编码之前你花一些时间规划你的小应用程序，面向对象将给你的工作带来方便。当你开始开发更多的小应用程序并重新使用某些对象时，你将发现这种规划非常值得，你在今后的长期应用开发中将节省许多时间。

即使存在这些相对的不足，Java 仍然是一种优秀的工具。记住，Java 是从最底层开发的用来解决高级程序设计语言问题的程序设计语言。Java 的开发者使它尽可能生动有趣并易于大家使用和学习。让 Java 来改变你设计程序的方法吧，你将发现 Java 是如此有趣，如此简单和方便！

1.2 与平台无关的方案

自 20 世纪 90 年代初 Java 诞生以来，它一直运用于各种不同的项目中。一开始，它作为个人数字助理软件专用的编程语言，后来成为交互性电视系统的首选语言，最后又定形为一种 Web 程序设计语言。这些转变表明：Java 有着很强的灵活性和适应性，它与机器的类型无关并适用于各种不同的操作系统中。

当你编写一个 Java 小应用程序时，该程序实际上是在一种非常特别的机器——Java 虚拟机上运行。Java 虚拟机是与平台无关方案的第一步。当你用 C++ 等语言开发软件时，你通常是在专用平台上编程，它们是运行 Windows 的计算机或苹果机。这些编程语言使用一系列的系统调用功能，而这些功能是和你的编程机器所使用的处理器有关的，是专用功能。由于这些语言使用专用的机器指令，当你在一种新的处理器上运行程序时，你必须修改它们。这些工作浪费了很多时间和资源。

Java 编码不是为任何一种物理机编写的，它是为虚拟机编写的，这种虚拟机实际上是另一种软件。虚拟机可以解释并执行编程者原来编写的程序，虚拟机与特定的机器有关，如 Window95 虚拟机，Sun 工作台虚拟机等等。甚至在 Netscape 中也内联了一种虚拟机，以使得浏览器能运行 Java 程序。

通过虚拟机在不同平台上的移植，Java 程序可用在任何运行虚拟机的计算机上。这种特性说明了为什么 Java 小应用程序在 Unix 工作平台上的运行和在 Windows 平台上的运行是一样的。Sun 微系统公司花了很大的精力将虚拟机移植到市面上几乎所有的主流机器中。这样，Sun 微系统公司就保证了 Java 的适用性，这种适用性使你一旦编写了 Java 程序，就可以在许多计算机上使用它。

1.2.1 适应一切的二进制编码

虚拟机需要一种特别的二进制编码来运行 Java 程序，这种编码叫做字节码，它不包含任何专用平台指令。这意味着在 Sun 工作平台上，Java 程序所编译的字节码同运行 Window 95 的计算机产生的字节码是一样的，这种编码是与平台开发环境无关的方案的第二步。设想一下，如果软件只需编写一次，就可在所有的计算机上运行，这将节省多少时间和金钱！我们再也不需要苹果版或 Windows 版软件，也不需要雇用大批的编程人员来将苹果版软件转化为 Windows 版软件。

当你在编译一段 Java 程序时，编译器将产生独立于平台的字节码，并且在任何 Java 虚拟机上都可以运行这种字节码。我们可以在小应用程序观察器或 Netscape 上看到这种编码的运行结果，也可以在 Sun 工作平台或运行 Window NT 的平台上看到同样的结果，因为底层环境的可移植性，使得单一的二进制码可适应一切计算机。

1.2.2 与平台一致的用户界面

虚拟机使得 Java 保持了与平台一致的用户界面。当你在 Windows 平台上运行 Java 应用程序时，应用程序中的菜单和工具条的界面形式与标准的 Windows 组件是一致的；如果你在 Sun 系统上运行同样的小应用程序，按钮、菜单等控件将同 Sun 系统的操作界面保持一致。

由于虚拟机是与平台有关的，所以，小应用程序使用的所有元件都将保留和它在主机上一样的特性。如果苹果机用户使用你的小应用程序，他可以像使用其他苹果软件一样使用你的小应用程序；如果他是一个 Windows 用户，你的小应用程序对他来说同其他 Windows 应用程序没什么区别。这种特性允许用户在相似的环境下工作，而不必学习任何新的用户界面的使用方法。

1.2.3 交互式的站点主页

在 Web 的初期，它只是一种共享静态数据的简单方法。文本或图像可由浏览器下载或显示，但不存在任何交互性，你不可能改变已载入的主页，除非你重新载入新的主页。后来，由于有了表格和通用网关接口(CGI)脚本语言，增强了 Web 的交互性能力，但仍然缺乏多媒体或实时的交互性能。Java 使你能够操纵动画(如图 1.6 所示)、声音或实时输入，或者进行数据处理，甚至你还可以将小应用程序同其他应用程序相连。Java 为 Web 站点增添了新的功能。

自从 Java 问世以来，许多小应用程序展示了 Java 制作动画的能力。直至今今天，用小应用程序制作动画仍然是一种趋势(如图 1.7 所示)。由于 Java 小应用程序显示标准格式图像(如 GIF 和 JPEG 图形格式)非常容易，所以，它的广泛流行是必然的。以前，在 Web 站点中制作动画往往采用网络和服务器的方法，一次显示一幅图像或显示服务器传过来的动画，这种方法速度慢、动作不连贯或者根本看不出是动画。由于 Java 小应用程序运行在本地机上，不会给网络增加负荷，因此，动画的过程非常连贯。