

流行软件指南丛书

Visual J++ 6.0

开发指南

陈 功 筹 编著



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



流行软件指南丛书

Visual J++ 6.0 开发指南

陈功等 编著

清华大学出版社

(京)新登字 158 号

JS196/28
内 容 简 介

本书全面介绍了微软公司最新的 Java 编程工具 Visual J++ 6.0, 以及如何使用它提供的功能强大的工具快速开发 Java 程序。书中通过大量的程序实例介绍了怎样使用 Visual J++ 编写、调试、运行 Java 程序。本书主要内容包括 Java 纵览、Java 语言简介、Visual J++ 6.0 集成开发环境、建立 Web Server、文本域及按钮控件、Java 的布局设计、文本区域和面板控件、复选框和单选按钮控件、滚动条控件、下拉列表框和滚动表控件、窗口和菜单设计、对话框控件、图形处理技术、鼠标与图形设计、图像处理、多线程程序设计、线程与动画设计、异常处理技术、Java + ActiveX 技术、CABINET 文件技术、安全技术、应用实例等。

本书适合于计算机专业开发人员、大专院校师生, 以及对 Visual J++ 和 Java 语言感兴趣的读者阅读。由于 Java 语言已经成为一种事实上的程序设计标准, 现有的应用程序 90% 以上将会要求用 Java 语言重新编写, 因此计算机专业开发人员都应掌握 Java 语言。

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。

图书在版编目(CIP)数据

Visual J++ 6.0 开发指南/陈功编著. —北京: 清华大学出版社, 1998.12

(流行软件指南丛书)

ISBN 7-302-03242-4

I. V… II. 陈… III. Java 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (98) 第 36158 号

出版者: 清华大学出版社(北京清华大学校内, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印刷者: 国防工业出版社印刷厂

发行者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 28.5 字数: 676 千字

版 次: 1999 年 1 月第 1 版 1999 年 1 月第 1 次印刷

书 号: ISBN 7-302-03242-4 / TP·1737

印 数: 0001 ~ 5000

定 价: 36.00 元

《流行软件指南丛书》编辑委员会

主 编	沈金发				
副主编	熊桂喜				
委 员	李幼哲	郑南京	李振格	王 敏	赵仲明
	王 国 印	郑金战	谷彦国	薛继峰	

《流行软件指南丛书》序

近年来,计算机以前所未有的速度发展和普及,一些优秀的计算机软件广泛流行。人们已离不开这些流行软件。学习这些软件,掌握与使用这些软件已是广大计算机用户的迫切愿望。

围绕着流行软件,国内许多出版社已翻译出版了不少国外图书,这无疑对广大用户用好这些软件起到了积极作用。但是,由于许多软件需要或已经汉化,国内也不断自行开发出一些优秀软件,加之文化背景的不同,仅靠翻译书远不能满足广大用户的需要。国内众多出版社虽然也组织编写了不少有关计算机软件的图书,但是无论从选题规模、组织方式还是人力投入上都不能和国外知名的软件丛书相比。针对这种状况,我们在清华大学出版社的大力支持下,组织国内一些长期从事计算机教学、科研与应用的专家和技术人员,精选那些为我国广大计算机用户普遍认同的优秀流行软件,精心策划了这套《流行软件指南丛书》。

《流行软件指南丛书》面向初级和中级用户,每一本介绍一种流行软件,内容基本覆盖该软件的全部功能;全书以功能为主线,组织方式与人们学习和使用软件的方式相一致;采用通俗易懂、循序渐进的方式讲解与阐述软件的各种使用方法和所涉及背景知识;根据每种软件的特点,配有针对性的图示和实例。总之,力求使本丛书中的每一本书既是学习软件的好教材,又是可以方便查阅的好参考书。

多年来,清华大学出版社在广大读者的关心和支持下,出版了不少深受欢迎的计算机图书。现在,本丛书编委会将以极大的热情、严谨的态度来组织编写这套面向广大计算机用户的普及读物,希望通过我们的努力,使这套丛书能真正成为大家工作与学习的好帮手。

最后,衷心地希望广大读者对本丛书的不足和缺点提出批评,对本丛书今后的发展提出宝贵意见。

丛书编委会

1995年6月

前 言

今天,计算技术的发展已经进入了网络时代。随着 Internet 的发展,万维网(World Wide Web)的普及,人们的生活和社会的各种活动,将越来越离不开信息网络的支持。Java是由 Sun 公司的 JavaSoft 分部开发的一种激动人心的新型编程语言。它专门设计在像因特网(Internet)这样的广域网上运行,所以说它是一种最适合于 Internet 应用开发的编程语言。它较好地解决了 Internet 上的异质性、代码交换和网络程序安全性等问题。与此同时,它也给万维网带来了新的生机和活力,使用 Java 后,原来只有静态图像和文本的网页就可以变成栩栩如生的动画。不仅如此,Java 的出现,对于计算机和信息技术应用方式 and 应用范围的影响也是广泛而深远的。可以说,Java 正是适应网络时代的需要而产生的。

Java 作为面向 Internet 的编程语言,一推出就倍受青睐。但是,作为一种编程语言,需要有功能强大的集成开发环境,才能快速、方便地开发出应用程序。Microsoft Visual J++ 6.0 正是这样的集成开发环境,它将用与 Visual C++ 完全相同的环境 Microsoft Developer Studio,这使得许多用户学习起来感觉非常亲切。

由于 Java 语言的优秀特性,不夸张地说,Java 语言将成为计算机专业开发人员的上岗证,希望读者能早日掌握这门令人兴奋的程序设计语言。

本书第一章到第三章由陈功、许广晗、刘力合作编写,第四章由张勤编写,第五章到第十二章由陈功、陈蕾、李强、刘闯、李维、时向宁、洪继群、刘石华合作编写,第十三章到第十五章由廖继、胡蓓、宋伟编写,第十六章到第二十一章由高山华、李立军、王家丰、周晨东、廖理纯合作编写,第二十二章由陈功编写。本书策划由吴达夫负责,统稿由王军负责,图片制作由张群负责,文字录入由陈清负责。

编者

1998.8

目 录

第一章 Java 纵览	1
1.1 Java 入门	2
1.2 浅谈 Java 技术的机理	7
1.3 Java 的安全性	9
第二章 Java 语言	13
2.1 Java 语法简介	13
2.2 Java 变量简介	16
2.3 Java 操作符简介	20
2.4 流向控制语句	23
2.5 面向对象的程序设计技术	26
第三章 Visual J++ 6.0 集成开发环境	38
3.1 Visual J++ 6.0 窗口界面	38
3.2 创建一个 Java 小程序	39
3.3 分析小程序	42
3.4 用模板创建 Java 小程序	44
3.5 Applet1.java 源程序	46
3.6 HTML 简介	53
3.7 解决方案和项目	56
第四章 建立 Web Server	60
4.1 Internet Information Server 程序介绍	60
4.2 建立自己的 Web 站点	61
4.3 使用自己的目录	64
4.4 设置域名服务	64
第五章 文本域及按钮控件	68
5.1 文本域控件	68
5.2 按钮控件	76
5.3 Java 事件	81

第六章 Java 的布局设计	94
6.1 BorderLayout 布局	94
6.2 CardLayout 布局	99
6.3 GridLayout 布局	104
6.4 GridBagLayout 布局	110
第七章 文本区域和面板控件	118
7.1 文本区域控件	118
7.2 面板控件	125
7.3 创建面板类	131
第八章 复选框和单选按钮控件	140
8.1 复选框控件	140
8.2 复选框事件	146
8.3 单选按钮控件	155
第九章 滚动条控件	167
9.1 创建滚动条	167
9.2 调整事件	173
第十章 下拉列表框和滚动表控件	184
10.1 创建下拉列表框	184
10.2 下拉列表框事件	190
10.3 创建滚动表	196
10.4 滚动表事件	203
第十一章 窗口和菜单设计	212
11.1 创建弹出式窗口	212
11.2 增加菜单	222
11.3 菜单项事件处理	232
第十二章 对话框控件	239
12.1 创建对话框	239
12.2 对话框事件处理技术	249
第十三章 Visual J++ 的图形处理技术	257
13.1 图形坐标系统简介	257
13.2 管理颜色和字体的类	258
13.3 常用的绘图方法	264
13.4 载入并显示图形	272
第十四章 鼠标与图形设计	277
14.1 熟悉鼠标事件	277
14.2 画图小程序	288
第十五章 图像处理	308
15.1 显示图像	308

15.2	缩放图像	315
15.3	使用 MediaTracker 处理图像	323
第十六章	Visual J++ 的多线程程序设计	329
16.1	多线程简介	329
16.2	分析一段多线程的实例	330
16.3	创建多线程的两种方法	331
16.4	多线程应用中的资源共享	337
第十七章	线程与动画设计	341
17.1	多线程小程序	341
17.2	实现动画	345
17.3	run()方法	346
17.4	如何消除闪烁	360
第十八章	Visual J++ 的异常处理技术	367
18.1	了解异常	367
18.2	Java 定义的标准异常类	369
18.3	异常的处理技术	370
18.4	创建一个自己的异常	372
第十九章	Visual J++ 的 Java + Active X 技术	374
19.1	通过 VBScript 语言访问 Java Applet 程序	374
19.2	使用 ActiveX 构件	378
第二十章	Cabinet 文件技术	389
20.1	Cabinet 文件的概念	389
20.2	创建 Cabinet 文件	390
20.3	使用 Cabinet 文件	393
第二十一章	Visual J++ 的安全技术	395
21.1	预备知识	395
21.2	实现数字签名	397
21.3	一个数字签名的实例	400
第二十二章	Visual J++ 的应用实例	405
22.1	应用实例 1	405
22.2	应用实例 2	408
22.3	应用实例 3	415
22.4	应用实例 4	422
22.5	应用实例 5	424
22.6	应用实例 6	427
22.7	应用实例 7	434

第一章 Java 纵览

Java 是一种面向 Internet/Intranet 计算的编程语言,随着“地球村”梦幻般世界的来临,Java 语言及其变体(如 JavaScript)已经广泛渗透到信息高速公路上的各种应用中。对于飞速发展的、进入到 Internet 时代的计算机技术,掌握 Java 程序设计的软件开发者无疑将成为新一轮技术更新与竞争的宠儿。

Java 是一种与平台无关的编程语言,可以开发在 Internet 上散布的应用程序。它是由 Sun 公司的 JavaSoft 分部开发的一种划时代的新型编程语言。它是一种网络语言,适合编写在因特网(Internet)这样的广域网上运行的各种应用程序,原先静止不动的网页在 Java 中因拥有了动画、行为而变得灵活、安全,且充满生气。

Java 的用途并不仅此。Java 的独特功能使得向数百万的网上用户发送面向对象的应用程序成为可能。随着计算机产业网络时代的到来,Java 已经宣告了基于网络计算的新纪元的到来。与其他任何一种语言开发工具不同,Java 是唯一适合在互联网络上分布执行的语言。人们可以从万维网 WWW 上的服务器动态下载 Java 程序,并且既可以在支持 Java 的浏览器上执行,也可以单独执行。

Java 作为面向 Internet 的编程语言,一推出就倍受青睐。但是,作为一种编程语言,需要有功能强大的集成开发环境,才能快速、方便地开发出应用程序。Microsoft Visual J++ 6.0 正是这样的集成开发环境之一,它使用与 Visual C++ 完全相同的环境 Microsoft Developer Studio,使得许多用户学起来比较方便。

Java 是一种新的计算机语言,它最显著的特点就是能够用来编写可在 Internet 网上广泛发行的独立于平台的应用程序。实际上,Java 不仅仅是一种计算机语言,它是一个由许多相关技术组成的系统,这个系统包括 Java 语言本身、安全模型、Java 实时系统以及其他一些元素。

在介绍 Visual J++ 之前,我们必须先了解一些关于 Java 的知识,本章和下一章就是供不熟悉 Java 的读者参考的,有 Java 编程经验的读者可以略过这两章,直接进入 Visual J++ 世界。

在本章,我们将简要介绍 Java 系统的各个元素,主要内容包括怎样用 Java 开发万维网(World Wide Web)应用程序、Java 小程序(Applet)和 Java 独立程序(Application)的区别、Java 虚拟机(Java Virtual Machine)和即时编译器(just-in-time compiler)的概念、怎样将

一个 Java 小程序加入到网页中、Java 安全模型和 COM 对象的关系,另外,我们还将讨论与 Visual Basic 和 Visual C++ 相比 Visual J++ 在开发网络应用程序方面的优点。

1.1 Java 入门

前面提到,Java 不仅仅是一种计算机语言。从根本上说,Java 是一个可进行分布计算 (distributed computing) 的平台和支持万维网的实时环境。下面列出 Java 语言的主要特点:

(1) Java 是面向对象 (object oriented) 的语言。

(2) Java 程序是独立于平台 (platform independent) 的语言。同样的一个 Java 程序既可以运行于 PC 机,也可以运行于 Macintosh 机、UNIX 机以及其他所有能够提供 Java 翻译器 (Java interpreter) 的操作平台上。对程序开发者来说,这意味着时间和金钱上的巨大节约。

(3) Java 具有内置安全性。这种安全性防止了网际应用程序对用户计算机的恶意侵犯,如读和写用户机上的文件。这就使网络用户可以放心地从网络上下载代码。

正是以上的特点使得 Java 成为目前最受欢迎的网络语言。下面我们简要介绍 Java 的特点。

1. Java 的由来

Java 起源于 1991 年,同年万维网也出现了。有趣的是,Java 的设计者 James Gosling 的最初目的并不是为了开发网络程序。他只不过是想要为一些消费用的电子器件 (如交互电视的顶端盒) 设计一个通用环境。当时,他的设计目标是发明一种可以方便地对这些消费器件进行编程的新语言,而这种新语言能够方便地从一个器件移植到另一个器件上。他的这种想法可以说是市场驱动的,因为消费者都希望这些电子器件能够比传统的计算机更好用和更简单。

Java 开发小组最初用 C++ 作为他们的模型。但很快他们就意识到需要一种比 C++ 安全性和可移植性更好的语言。于是,他们先开发出一种 OAK 语言,后来这种语言被称为 Java。这种语言的语法和功能与 C++ 差不多,但相比之下更为简单和平台中性化 (Platform-neutral)。

到 1992 年秋天,开发小组开发出一种具有小型可视界面的手持遥控设备。这个设备被称为 * 7, * 7 中有一个叫 Duke 的动画人物,这个人物引导人们操作图形界面,后来 Duke 成为 Java 的吉祥物。

1994 年中期,万维网的成功引起了 Sun 公司的注意。OAK 小组注意到了网页的一个根本问题:它们都是单调乏味的。无论一个网页上有多少嵌入的图形、声音和图像,网络浏览者无法和这个网页交互。虽然开发者可以通过 HTML 将数据从网页上发送到一个基于服务器的程序,但这也是他所能做的全部。网页需要交互性,正是这个灵感使 OAK 小组致力于使 Java 语言向 Internet 和万维网方向发展。

1994 年秋天,Sun 公司发行了 WebRunner 和 HotJava,它们都是用 Java 开发的网页浏览器。网络先锋 Netscape 公司注意到了这种新技术,他们宣布他们流行的浏览器 Netscape Navigator 将支持 Java。

从今天的标准来看,最早的 Java 开发工具是非常原始的,比如 Sun 公司的 Java Developers Kit(JDK)1.0。这类开发环境不仅没有交互功能,而且调试功能也很有限。至 1996 年初,几个公司都发行了交互开发环境。1996 年中期,微软公司发行了它自己的 Java 交互开发环境:Microsoft Visual J++ 1.0。

Microsoft Visual J++ 完全符合 Sun 公司的 Java 语言规范。它是一个开发 Java 小程序和独立程序的编程平台,包括 Java 编译器、编辑器、调试器和大量在线文件。

近年来,随着网络的蓬勃发展,Java 语言受到越来越多的网络程序设计人员的青睐。人们称其为 21 世纪的语言,这恐怕是 Java 的开发者们所始料不及的。

虽然 Java 是以 C++ 为模型生成的,但它的开发者们赋予它许多新特点:更简单的语法;更好的可靠性和安全性;在不同平台间的可移植性更强。因此,尽管 Java 保留了许多 C++ 的风格,实际上它是一种很不一样的语言了。Java 和 C++ 之间的主要不同点如下:

(1) 由于 Java 是为开发网络应用程序而设计的,它能够支持高效的通讯和发布;而 C++ 不具备这些功能。

(2) Java 编译器生成的是独立于平台的虚拟代码(bytecode)文件,这种文件的执行需要 Java 虚拟机;而 C++ 编译器生成的是可执行机器代码文件。

(3) Java 既可以生成与传统可执行文件差不多的独立应用程序,也可以生成在网上实时运行的小程序;而 C++ 只能生成独立应用程序或组件。

(4) 在运行时,Java 程序动态地链接所有需要的类;而 C++ 程序只能在编译时静态地链接所需代码。

(5) Java 只支持对内存的引用;而 C++ 不仅支持引用,还支持指针和直接内存访问(DMA)。

(6) Java 支持自动收集垃圾的动态内存分配;C++ 也支持动态内存分配,但开发者必须自己为每一个类分配内存。

由于有以上这些不同,在创建高级应用程序时,Java 可以部分代替 C++。但是,C++ 的灵活性、高效率 and 成熟使其仍然是一种常用的程序开发语言。

2. Java 小程序和独立程序

Java 程序可以是小程序也可以是独立的应用程序,两类 Java 程序都具有以下特点:

(1) 两类程序都由一个或多个以 .CLASS 为后缀的文件组成。

(2) 两类程序都需要用户系统安装 Java 虚拟机(JVM)。Java 虚拟机能够载入并翻译 Java 程序,并且可以提供 Java 内核包的实现。

当然,这两类程序也有不同点,它们的主要不同在于:

(1) Java 小程序可以被嵌入 HTML 网页内,从而可以在网络上发布,当浏览网页时它们可以在浏览器中运行;而 Java 应用程序却不支持网页嵌入和下载。

(2) Java 小程序只能在与 Java 兼容的容器中运行,例如现代的网页浏览器;而 Java 应用程序却没有这个限制。

(3) Java 小程序的运行受到严格的安全限制,例如它不能访问用户计算机上的文件和系统服务;而 Java 应用程序没有固有的安全限制。关于安全性问题,稍后会详细讨论。

实际上,我们很容易生成既可以是 Java 小程序又可以是 Java 独立应用程序的混合 Java 程序,下一章介绍 Visual J++ 的 Applet Wizard 时再讨论具体的实现方法。下面,我们将详细说明 Java 小程序和 Java 独立应用程序之间的异同点。

从用户的角度来说,Java 小程序和 Java 独立程序在运行时有很明显的差别。图 1.1 是在网页浏览器上下载一个 Java 小程序运行的情况,而图 1.2 是在本地机上运行 Java 独立程序时的情况。它们之间的差异是一目了然的。



图 1.1 小程序的运行结果

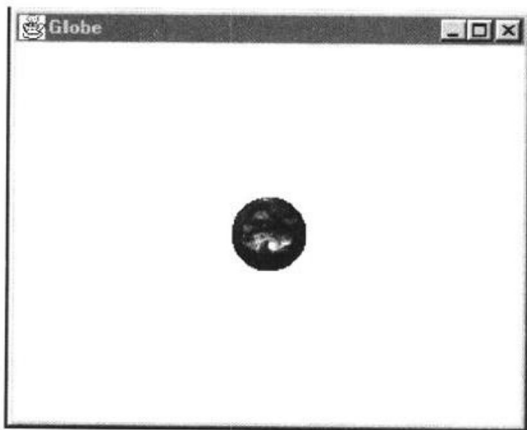


图 1.2 独立程序的运行结果

Java 小程序和 Java 独立程序在代码的编写上也有很大差别。前面提到过:Java 是面向对象的程序设计语言。这意味着它的主要组成模块是类或者说是用户自定义的数据类型。在 Java 代码中实现这些类的方式就决定了开发者设计的是 Java 小程序还是 Java 独立程序。对源代码来说,Java 小程序和独立程序之间的基本区别如下:

(1) 一个 Java 小程序必须定义一个 Applet 类的子类,一个独立程序也可以定义一个 Applet 类的子类,但这不是必需的。下面的代码演示了如何扩展 Applet 类:

```
//从 java.applet 类库中引入类
import java.applet.*;
//从 Applet 类中扩展一个名为 FirstApplet 的子类
public class FirstApplet extends Applet {
    //Java 源程序
}
```

(2) 一个 Java 独立程序必须在一个类中定义一个 main 方法,该方法代表了该独立程

程序的入口点;而一个 Java 小程序并不定义 main 方法,它的执行是由 Applet 类定义的多个方法控制的。下面的代码演示了怎样在一个 Java 独立程序中定义 main 方法:

```
//定义一个叫作 FirstApplication 的类
public class FirstApplication {
//定义 main 方法
public static void main(String Args[]) {
    // Java 源程序
}
```

3. Java 程序的开发和发行机制

Java 开发平台及相关支持技术的更新速度是非常快的。商业 Java 程序在 Java 语言中的开发和部署很好地说明了 Java 的快速成熟。这主要是由以下因素造成的:

(1) 降低了开发时间和花费

Java 语言一开始就是面向对象的,这样,开发者一旦掌握了它面向对象的特点,开发工作能够比常规的循序渐进方法进行得更有效。而且,Java 语言在快速生成高级和强壮的软件方面比 C++ 有更大的优越性。

(2) 商业类库

Java 语言有大量的内置类,这些类是开发者编写 Java 程序的基础。渐渐地,商业类库的出现使开发者的工作有一个很高的起点。例如,如果开发者能够使用提供用户界面功能(如各种窗口控制)的商业类库,他编写 Java 小程序的工作将简单得多。

(3) 基于组件的应用程序

很多软件商认为 Java 是新一代可繁殖程序的基础,这些可繁殖程序是基于对象或组件编程模型的。在这个模型中,应用程序由一些可下载的组件组成,而不是单一的闭门造车的产物。

在使用一个基于组件的应用程序时,用户并不需要将整个程序安装在他的本地系统上。相反,他可以只从浏览器上下载所需要的那部分功能。例如,在创建一个字处理程序时,用户从第一个小程序中得到主要的字处理功能,而图形功能可以在需要时才从网上的第二个小程序中下载。

这种程序开发的组件模型和传统的软件开发方式有很大的不同,现在还只处于初级阶段。如果程序开发人员采用这种开发模型,他将面对很大的挑战。

Java 语言和 Java 实时系统允许开发者在万维网上发行 Java 小程序。小程序一般都很小,在浏览器提出请求时可以被下载并执行。Java 小程序的加入给万维网增色不少。例如,Java 小程序可以提供高级控制并对用户产生的事件作出反应。Java 语言提供丰富的事件模型,它比 HTML 具有更强大的功能。

Java 还允许开发者编写可以脱离网页浏览器运行的有特色的 Java 独立程序。在这方面,Java 有可能代替 C 语言或其他的语言,这是因为 Java 具有以下特点:

(1) 使用 Java 强大的网络功能可以改善许多本地基于网络的客户/服务器程序的

性能。

(2) Java 独立程序可以在一个组织的本地网上开发、安装和运行,而 Java 小程序却有那么多的安全限制。另外,由于 Java 有许多内置的安全特性,它可以为用户系统提供额外的保护。

4. Java 虚拟机概述

前面说过,要运行 Java 程序必须在用户系统上安装 Java 虚拟机。Java 虚拟机实际上是一个应用程序,该程序包括一个 Java 虚拟码的翻译器,它可以执行虚拟码指令。Java 虚拟机把虚拟码翻成本地代码(也叫机器码)。可以看出,Java 程序的执行速度肯定没有本地应用程序快,这是目前 Java 面临的问题之一。可以说,这是为实现系统可移植性而付出的代价。打个比方说明 Java 虚拟机和虚拟码之间的关系:如果说虚拟码是汇编代码的话,那么 Java 虚拟机就相当于中央处理器(CPU)。

还有一个问题就是:如何安装 Java 虚拟机。事实上,大部分开发者不需要考虑这个问题,因为和 Java 兼容的网络浏览器都内置 Java 虚拟机,只要安装一个和 Java 兼容的浏览器就自动安装了 Java 虚拟机。例如,只要安装了微软的 Internet Explorer 就自动安装了 Java 虚拟机。我们可以从下面的网址免费得到微软的 Internet Explorer:

<http://www.microsoft.com/ie>

从长远的角度看,最新推出的操作系统中都会把 Java 虚拟机集成到系统中去,所以我们根本不用为安装 Java 虚拟机担心。

5. Java 小程序与 HTML

当我们在网上运行 Java 小程序时,实际上是在网页的 HTML 代码中得到 Java 小程序。HTML 允许向 Java 小程序传送参数,例如,可以通过参数规定一个动画的移动速度。

在浏览器中调用 Java 小程序的传统方法是使用 HTML 的 APPLET 标志。下面的例子演示了如何使用该标志调用 Java 小程序:

```
< APPLET
  CODE = MyApplet.class
  ID = FirstApplication
  WIDTH = 320
  HEIGHT = 240 >
</APPLET>
```

其中参数的意义如下:

- (1) CODE 指定要下载的 Java 虚拟码文件(.class)。
- (2) CODEBASE 用 URL 指定要下载的 Java 文件所在的目录。
- (3) ID 指定小程序的名字,在 HTML 文档中该参数可以指代 Java 小程序。
- (4) WIDTH 指定小程序窗口的宽度。
- (5) HEIGHT 指定小程序窗口的高度。

第二个参数 CODEBASE 是可以缺省的,如果 Java 类文件和 HTML 文件没有放在同一

个目录下,就必须用该参数指定相应的目录。下面的例子演示了如何用 APPLET 标志调用一个不在同一目录下的 Java 小程序:

```
< APPLET
    CODE = MyApplet.class
    CODEBASE = "http://www.acme.com/JavaClasses"
    ID = FirstApplication
    WIDTH = 320
    HEIGHT = 240 >
</APPLET>
```

前面提到可以在 HTML 中将参数传递给 Java 小程序,可以用两种方法实现这项功能:一种是用 PARAM 标志静态地传递这些参数。在下面的例子中,传递的参数是 Message,它的值被设定为“Hello!”。

```
< APPLET
    CODE = FirstApplet.class
    ID = FirstApplet
    WIDTH = 320
    HEIGHT = 240 >
< PARAM name = Message value = "Hello!" >
</APPLET>
```

另一种方法是用脚本文件调用 Java 小程序,第九章将详细介绍这种方法。

1.2 浅谈 Java 技术的机理

Java 语言及其实时环境是为方便网络计算而设计的。最初的 Java 开发小组致力于使 Java 程序具有很大的灵活性和很高的可靠性,从而可以通过网络发行并在任一个操作系统上运行。这样,Java 的机理必须有自己的特点。本节的内容就是介绍 Java 语言的机理和 Java 程序在网络上运行的方式。

1. Java 机理简介

Java 语言的设计目的就是使其程序能在各种各样的计算机上运行,因此,Java 的机理和大部分其他的现代语言不同。

Java 实时系统先将源文件以 ASCII 码文件格式存储,然后这些源文件编译成 Java 虚拟码(bytecode)文件。Java 虚拟码是一种标准化的、与机器无关的低级语言。最后,用户系统上的 Java 虚拟机将这些虚拟码文件翻译成在本地机上可执行的机器码。举一个例子,当我们从网络上下载一个 Java 小程序时,网页浏览器中的 Java 虚拟机就将 Java 虚拟码翻译成可执行的机器码。由此可见,在网上发行的 Java 程序是与机器无关的虚拟码,这就是 Java 实现独立于平台的方法。前面已经说过,这样做的代价是 Java 程序的执行速度变慢了,因为比其他现代语言程序的执行过程多了一个翻译虚拟码的中间过程,而且翻

译器的翻译速度比本地机器码的执行速度要慢。

为了提高 Java 程序特别是大型 Java 程序的执行速度而不影响它的可移植性,许多 Java 零售商在他们的浏览器中或编译器中安装了即时编译器(just-in-time compiler)。这个可选器件可加入 Java 机理中代替或补充 Java 翻译器的作用。因为 Java 翻译器是 Java 虚拟机的一部分,那么 Java 即时编译器也可以看做是 Java 虚拟机的一部分。Java 即时编译器的工作方式可以从它的名字中推断出来,它不是一行一行地翻译虚拟码指令,而是一次将整个 Java 方法翻译成相应的本地码;另外,这种翻译工作不是在编译时完成的而是边运行边进行,这一点与 C 和 C++ 语言差不多。当然,这种方法的代价是本地代码的执行时间加长,因此,对一些 Java 小程序来说,使用 Java 即时编译器反而会降低整体性能。一般来说,如果 Java 程序中的重复操作很多或计算比重很大,使用 Java 即时编译器能够大幅度地提高 Java 程序的执行速度。

2. Java 实时系统

通过上面的介绍,我们知道了 Java 实时系统主要是由以下几个组件构成的:

(1) 将 ASCII 码文件转换为 Java 虚拟码的 Java 源代码编译器

如果要编写一个 Java 小程序或 Java 独立程序,必须先编写 Java 源程序。Java 源程序通常都是以 .java 为后缀的 ASCII 文件。一般的,都要使用 Visual J++ 类库或其他商业类库来完成 Java 源代码的编写工作。一个或多个 Java 源代码就可以组成一个 Java 项目。

当我们完成了一个 Java 项目后,必须运行一个 Java 源代码编译器,例如 Visual J++。

Java 源代码编译器为每一个 Java 源文件生成一个 Java 虚拟码文件,这种文件都是以 .class 为后缀的。在 Java 虚拟码文件中就含有可以被下载并在本地机上执行的指令。

在完成虚拟码文件的转换过程之后,我们既可以将它们放在一个网络服务器的目录中,又可以把它们当作独立程序在自己的系统上运行。但是,无论把它们作为网上的 Java 小程序还是 Java 独立程序来运行,它们都是被下载到一个 Java 虚拟机中并翻译执行的。

(2) 将 Java 虚拟码转换为本地机器码的 Java 虚拟机

关于 Java 虚拟机我们已作了很多的介绍,现在只是简单介绍一下它具体的工作情况。在 Java 虚拟机转换 Java 虚拟码时,一般要做以下工作:

① 检查 Java 虚拟码文件的合法性,包括检查它们的格式和安全性。这项功能也被称作虚拟码识别器(bytecode verifier)。

② 为 Java 虚拟码文件分配内存。这项功能也被称作类载入器(class loader)。

③ 翻译 Java 虚拟码指令。

④ 完成有关标准类库的调用,这些类库经常是作为 Java 虚拟机的一部分安装的。

(3) Java 即时编译器

我们只能在支持 Java 即时编译器的浏览器(或者 Java 虚拟机)中运行 Java 即时编译器,比如微软公司的 Internet Explorer 就是这样的—一个网络浏览器。可以按以下的步骤使 Internet Explorer 支持 Java 即时编译器:

① 在 View 菜单中单击 Options 菜单项。

② 在 Options 对话框中单击 Advanced 按钮。