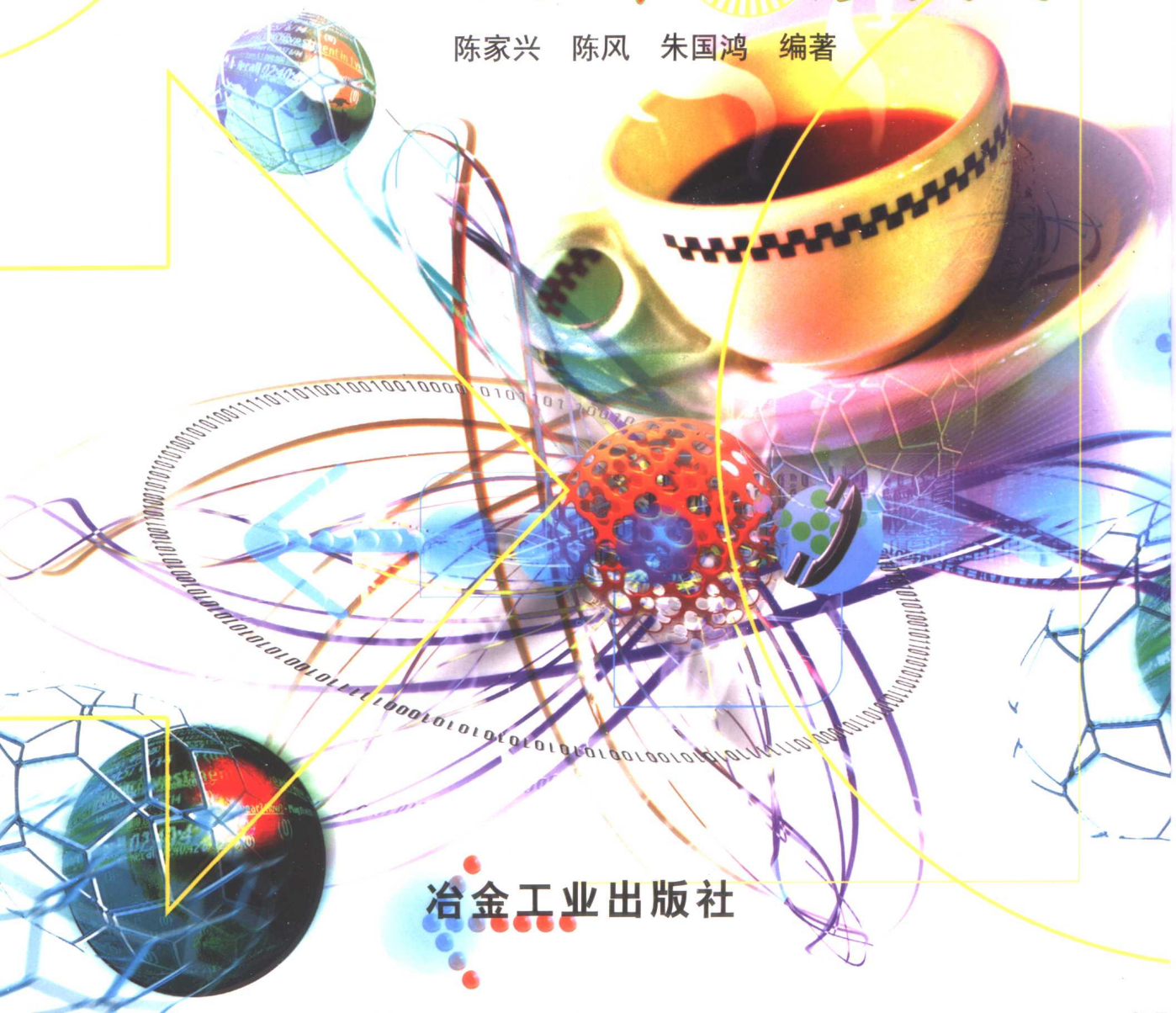


JAVA

图形用户界面

设计与实例

陈家兴 陈风 朱国鸿 编著



冶金工业出版社

JAVA 图形用户界面 设计与实例

陈家兴 陈 风 朱国鸿 编著

北 京

冶 金 工 业 出 版 社

2004

内 容 简 介

Java 是 Sun 公司开发的新一代编程语言, 它正逐步成为 Internet 应用的主要开发语言。本书主要介绍了 Java 语言基础、基本的用户界面构件及设计、事件处理与窗口构造构件、JFC/Swing 基本知识及 Swing 系列组件, 通过简单文本编辑器实例、简单浏览器实例、围棋程序实例、Java 开发时钟实例、俄罗斯方块实例及字母打字练习实例具体介绍了如何利用 Java 语言建立图形用户界面程序的方法。

本书语言通俗易懂、内容丰富、实用性强, 既可作为各大中专院校相关专业和 Java 培训班的学习和参考用书, 也可作为相关技术人员和自学者的学习和参考用书。

图书在版编目 (CIP) 数据

JAVA 图形用户界面设计与实例 / 陈家兴等编著. —北京: 冶金工业出版社, 2004.1
ISBN 7-5024-3446-1

I. J... II. 陈... III. JAVA 语言—程序设计
IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 120112 号

出版人 曹胜利 (北京沙滩嵩祝院北巷 39 号, 邮编 100009)

责任编辑 戈兰

湛江蓝星南华印务公司印刷; 冶金工业出版社发行; 各地新华书店经销

2004 年 1 月第 1 版, 2004 年 1 月第 1 次印刷

787mm×1092mm 1/16; 27.25 印张; 634 千字; 428 页; 1-3500 册
49.00 元

冶金工业出版社发行部 电话: (010) 64044283 传真: (010) 64027893

冶金书店 地址: 北京东四西大街 46 号 (100711) 电话: (010) 65289081

(本社图书如有印装质量问题, 本社发行部负责退换)

前 言

一、本书背景

Java 是 Sun 公司推出的新一代面向对象程序设计语言，它来自于 Sun 公司的 Green 项目，特别适合于 Internet 应用程序开发，其平台无关性直接威胁到 Wintel 的垄断地位。Java 已经得到工业界的认可，计算机产业的许多大公司都购买了 Java 的许可证，包括 IBM、Apple、DEC、Adobe、HP、Oracel 以及 Microsoft。众多的软件开发商开始支持 Java 的软件产品。在以网络为中心的计算时代，不支持 HTML 和 Java，则应用程序的应用范围只能限于同质的环境（即相同的硬件平台）。Intranet 正在成为企业信息系统最佳的解决方案，而其中 Java 将发挥不可替代的作用。Java 语言正在不断发展和完善，Sun 公司是其主要的发展推动者，较通用的编译环境有 JDK（Java Development Kits）和 JWS（Java Work Shop）。同时，还有很多其他公司正在开发 Java 语言的编译器与集成环境，Java 语言的正确性与效率都将会会有很大的提高，用户用 Java 编程将更方便。

Java 是一种简单的、面向对象的、分布式的、解释的、健壮的、安全的、结构中立的、可移植的、性能很优异的、多线程的、动态的语言。它十分适合在 Internet 环境上开发应用系统。目前，很多用户都希望能掌握 Java 语言，并利用其进行程序设计，本书正是基于这种需求而编写的。

二、本书内容结构

本书共 15 章，可分为三大部分，各部分主要内容安排如下：

第一部分：第 1~3 章。主要介绍了 Java 语言基础、基本的用户界面构件及设计以及事件处理与窗口构造构件。

第二部分：第 4~9 章。主要介绍了 Swing 的基础知识及其系列组件，包括图标、标签与边框，按钮、单选按钮和复选框，列表和组合框、文本编辑组件以及菜单和工具条等相关知识。

第三部分：第 10~15 章。主要介绍了简单文本编辑器实例、简单浏览器实例、围棋程序实例、Java 开发时钟实例、俄罗斯方块实例及字母打字练习实例。

三、本书特点

本书内容丰富、结构合理且注重理论与实践相结合，力求帮助读者由浅入深地学习和掌握与 Java 图形用户界面设计相关的原理及其技巧，针对性强，具有较强的实用性。

四、适用对象

本书适用范围广，既可作为各大中专院校相关专业和 Java 培训班的学习和参考用书，也可作为相关技术人员和自学者的学习和参考用书。

读者如有好的建议或意见，可发 E-mail 至：**service@cnbook.net**，也可到相关网站进行探讨，网址：**http://www.cnbook.net**（本书源代码也可从该网站免费下载）。

由于作者水平有限，编写时间仓促，书中错漏之处在所难免，敬请广大读者批评指正。

编 者

2003 年 11 月

目 录

第 1 章 Java 语言基础1	2.12 FlowLayout 类..... 39
1.1 Java 语言简介.....1	2.13 GridLayout 类..... 41
1.1.1 Java 出现的背景..... 1	2.14 GridBagLayout 类..... 42
1.1.2 Java 语言对软件开发技术的影响..... 2	2.15 面板..... 45
1.1.3 工业界对 Java 语言的评价..... 2	2.16 用户登陆系统的设计..... 46
1.1.4 Java 语言的应用前景..... 3	小结..... 49
1.2 Java 语言的特点.....3	第 3 章 事件处理与窗口构造构件 50
1.2.1 Java 语言的自身特性..... 3	3.1 基于继承的事件处理..... 50
1.2.2 Java Applet..... 5	3.1.1 旧 AWT 事件模式..... 50
1.2.3 Java 与 C 及 C++ 的比较..... 5	3.1.2 action 的工作模式..... 51
1.3 获取 Java 资源的手段.....7	3.1.3 事件处理器..... 52
1.4 Java 语言的新进展.....7	3.2 授权事件模式..... 54
1.5 Java 简单程序.....8	3.2.1 事件和接收者类型..... 54
1.5.1 Application 应用程序..... 8	3.2.2 ActionEvent 类..... 56
1.5.2 Applet 程序..... 9	3.2.3 用接收器适配器简化操作..... 58
1.6 Java 语言的开发工具..... 11	3.3 接收鼠标、键盘的输入..... 60
1.6.1 JDK 的组成..... 11	3.3.1 事件组件方法和调用时机..... 61
1.6.2 JDK 的下载..... 11	3.3.2 鼠标的按键..... 61
1.6.3 JDK 参考文件的下载与安装..... 15	3.3.3 鼠标的移动..... 63
1.6.4 JCreator 的使用..... 17	3.3.4 键盘的输入..... 64
小结..... 23	3.4 窗口构造构件简介..... 65
第 2 章 基本的用户界面构件及设计24	3.4.1 List 类..... 65
2.1 AWT 简介.....24	3.4.2 Choice 类..... 67
2.2 框架.....25	3.4.3 菜单栏、菜单和菜单项..... 69
2.3 标签.....27	3.4.4 对话框..... 72
2.4 按钮.....28	3.4.5 文件对话框..... 74
2.5 复选框.....29	小结..... 76
2.6 单选按钮.....31	第 4 章 JFC/Swing 简介 77
2.7 文本框.....33	4.1 Swing 与 AWT..... 77
2.8 文本区.....34	4.2 轻量组件与重量组件..... 78
2.9 布局管理器简介.....36	4.3 插入式界面模式简介..... 78
2.10 BorderLayout 类.....36	4.4 对等组件与插入式界面样式..... 79
2.11 CardLayout 类.....38	4.5 Swing 组件..... 79

4.6 Swing 包概览.....	80	6.3 切换按钮和按钮组	119
4.7 Swing 组件的预览.....	82	6.4 单选按钮	120
4.8 编写 Swing 小应用程序.....	85	6.4.1 单选按钮的构造函数	120
4.8.1 根板、层板和内容板.....	85	6.4.2 单选按钮的示例	121
4.8.2 创建 Swing applet.....	86	6.5 复选框	124
4.8.3 创建 Swing 应用程序.....	88	6.5.1 JCheckBox 的构造函数.....	124
小结	91	6.5.2 复选框示例	125
第 5 章 图标、标签与边框	92	小结	128
5.1 JComponent 类.....	92	第 7 章 列表和组合框	129
5.1.1 JComponent 类的层次结构.....	92	7.1 列表	129
5.1.2 JComponent 类的扩展功能.....	92	7.1.1 JList 类的构造函数和常用方法....	129
5.1.3 JComponent 的常用方法.....	93	7.1.2 使用列表	130
5.2 Swing 中的图标.....	96	7.1.3 列表示例程序	134
5.2.1 图标接口.....	96	7.2 组合框	137
5.2.2 创建图标.....	97	7.2.1 JComboBox 类的构造函数	
5.3 图像图标.....	98	和常用方法	138
5.3.1 ImageIcon 类.....	98	7.2.2 使用组合框	138
5.3.2 ImageIcon 类构造函数		7.2.3 不可编辑组合框示例	140
和常用方法	99	7.2.4 可编辑组合框示例	142
5.3.3 创建 ImageIcon 类对象.....	99	小结	145
5.4 标签.....	100	第 8 章 文本编辑组件	146
5.4.1 JLabel 类.....	101	8.1 文本编辑组件简介	146
5.4.2 JLabel 类的构造函数	101	8.2 文本编辑组件 API.....	147
5.4.3 创建标签对象示例.....	101	8.3 使用文本组件	151
5.5 边框.....	104	8.3.1 使用 Text Field.....	151
5.5.1 边框应用程序接口	104	8.3.2 使用密码框	151
5.5.2 使用 BorderFactory 类.....	105	8.3.3 使用文本区域	152
5.5.3 一个完整的创建和		8.3.4 使用编辑窗格来显示 URL 文本... 153	
设置边框的例子	106	8.3.5 使用文本窗格	153
小结	112	8.4 文本编辑的示例程序.....	154
第 6 章 按钮、单选按钮和复选框	113	小结	161
6.1 抽象按钮类.....	113	第 9 章 菜单和工具条	162
6.2 JButton 类	113	9.1 菜单及其使用	162
6.2.1 JButton 类的构造函数		9.1.1 Swing 菜单组件的层次	162
和常用方法	114	9.1.2 Swing 菜单的 API.....	162
6.2.2 一个按钮的演示程序.....	116		

9.1.3 建立菜单.....	164	11.4.1 Show.java.....	262
9.1.4 为菜单项添加事件处理.....	166	11.4.2 BrowserFrame.java.....	262
9.1.5 添加键盘操作.....	166	11.4.3 ToolBar.java.....	265
9.1.6 创建弹出式菜单.....	167	11.4.4 MenuBar.java.....	266
9.2 工具条及其使用.....	182	11.4.5 WebNode.java.....	268
9.2.1 工具条类的 API.....	182	11.4.6 PopupDialog.java.....	269
9.2.2 ToolBarDemo 示例程序.....	183	11.4.7 List.java.....	270
9.2.3 拥有不同组件的工具条示例.....	186	11.4.8 WebWindow.java.....	271
小结.....	190	11.4.9 HistoryList.java.....	272
第 10 章 简单文本编辑器实例.....	191	11.4.10 BookmarkList.java.....	273
10.1 实例说明.....	191	11.5 运行结果.....	276
10.2 开发思路.....	191	小结.....	277
10.3 重点代码与分析.....	192	第 12 章 围棋程序实例.....	278
10.3.1 简单框架的建立.....	192	12.1 实例说明.....	278
10.3.2 添加编辑区.....	194	12.2 开发思路.....	278
10.3.3 添加菜单.....	195	12.3 重点代码与分析.....	279
10.3.4 文本编辑器中的对话框.....	204	12.3.1 运行框架的建立.....	279
10.3.5 完善的文本编辑器菜单系统.....	208	12.3.2 棋盘格子的绘制.....	283
10.3.6 弹出式菜单.....	211	12.3.3 程序的完成.....	284
10.3.7 添加工具条.....	214	12.4 完整程序代码.....	293
10.4 完整程序代码.....	216	12.4.1 Weiqi.java.....	293
10.4.1 MyTextEditor.java.....	217	12.4.2 Qizi.java.....	294
10.4.2 Finder.java.....	237	12.4.3 Qipan.java.....	294
10.4.3 About.java.....	240	12.5 运行结果.....	305
10.5 运行结果.....	242	小结.....	305
小结.....	243	第 13 章 Java 开发时钟实例.....	306
第 11 章 简单浏览器实例.....	244	13.1 实例说明.....	306
11.1 实例说明.....	244	13.2 开发思路.....	306
11.2 开发思路.....	244	13.3 重点代码与分析.....	307
11.3 重点代码与分析.....	246	13.3.1 html 文件部分.....	308
11.3.1 运行框架的建立.....	247	13.3.2 基本画面描绘的算法.....	310
11.3.2 添加菜单栏.....	248	13.4 完整程序代码.....	318
11.3.3 添加工具条.....	250	13.4.1 JavaClock.html.....	318
11.3.4 添加主窗口.....	251	13.4.2 JavaClock.java.....	319
11.3.5 添加动作.....	252	13.5 运行结果.....	329
11.4 完整程序代码.....	262	小结.....	329

第 14 章 俄罗斯方块实例.....	330	14.4.10 Seven.java.....	394
14.1 实例说明.....	330	14.4.11 Eight.java.....	403
14.2 开发思路.....	330	14.5 运行结果.....	406
14.3 重点代码与分析.....	331	小结.....	406
14.3.1 数学基础.....	331	第 15 章 字母打字练习实例.....	407
14.3.2 工作平台的建立.....	333	15.1 实例说明.....	407
14.3.3 方块文件的建立.....	336	15.2 编程思路.....	407
14.3.4 程序的运行过程.....	340	15.3 重点代码与分析.....	408
14.4 完整程序代码.....	343	15.3.1 运行框架的建立.....	408
14.4.1 Root.java.....	343	15.3.2 主体部分的构建.....	409
14.4.2 GameTable.java.....	344	15.4 完整程序代码.....	418
14.4.3 Game.java.....	345	15.4.1 Applet1.html.....	418
14.4.4 One.java.....	353	15.4.2 Applet1.java.....	418
14.4.5 Two.java.....	360	15.4.3 MyPanel.java.....	419
14.4.6 Three.java.....	363	15.5 运行结果.....	427
14.4.7 Four.java.....	369	小结.....	427
14.4.8 Five.java.....	375	参考文献.....	428
14.4.9 Six.java.....	384		

第 1 章 Java 语言基础

最近一年多来,在 Internet 上出现的特别吸引人的就是 Java 语言和用 Java 编写的浏览器 HotJava。据说这是当初在设计 Java 时,程序设计师们在印尼的爪哇岛度假,喝 Java 咖啡时得到的灵感,因此得名。

本章将向读者介绍 Java 语言的一些基础知识,其主要内容如下:

- (1) Java 语言简介。
- (2) Java 语言的特点。
- (3) 获取 Java 资源的手段。
- (4) Java 语言的新进展。
- (5) Java 语言简单程序。
- (6) Java 语言的开发工具。

1.1 Java 语言简介

Java 语言是 SUN 公司开发的一种编程语言。它具有平台独立性、安全性、面向对象、动态性、分布性等卓越的特性,具有强大的网络功能。广义的 Java 是一个普遍适用的软件平台,可用于 Internet、Intranet 网及各种设备、系统、计算平台。它是一种基本的、结构紧凑的先进技术,一经产生就引起了广泛的关注,并在很短的时间内蓬勃发展起来。

1.1.1 Java 出现的背景

1991 年, SUN MicroSystem 公司的 James Gosling、Bill Joe 等人为在电视、控制烤面包箱等家用消费类电子产品上进行交互式操作而开发了一个名为 Oak 的语言(即一种橡树的名字),但当时并没有引起人们的注意。直到 1994 年下半年,Internet 的迅猛发展及环球信息网 WWW 的快速增长,促进了 Java 语言研制的进展,使得它逐渐成为 Internet 上受欢迎的开发与编程语言。一些著名的计算机公司纷纷购买了 Java 语言的使用权,如 MicroSoft、IBM、Netscape、Novell、Apple、DEC、SGI 等。因此,Java 语言被美国的著名杂志 PC Magazine 评为 1995 年十大优秀科技产品,随之出现了大量用 Java 编写的软件产品,受到工业界的重视与好评,认为“Java 是八十年代以来计算机界的一件大事”。微软总裁比尔·盖茨在悄悄地观察了一段时间后,不无感慨地说:“Java 是长时间以来最卓越的程序设计语言”,并确定微软整个软件开发的战略由 PC 单机时代向以网络为中心的计算机时代转移,而购买 Java 则是他重大战略决策的实施部署。

Java 的诞生必将对整个计算机产业发生深远的影响,对传统的计算模型提出了新的挑战。SUN MicroSystem 公司的总裁 Scott McNealy 认为,Java 为 Internet 和 WWW 开辟了一个崭新的时代。环球信息网 WWW 的创始人 Berners-Lee 说:“计算机事业发展的下一个浪潮就是 Java,并且将很快会发生的”。

MicroSoft 和 IBM 两大公司都计划在 Internet 上销售用 Java 编写的软件。

Apple、HP、IBM、MicroSoft、Novell、SGI、SCO、Tandem 等公司均计划将 Java 并

入各自开发的操作系统,而负责开发并推广 Java 技术的 SunSoft 公司(这是 SUN 下属的一个子公司),将通过颁发许可证的办法来允许各家公司把 Java 虚拟机和 Java 的 Applets 类库嵌入其开发的操作系统,这样各类开发人员就能更容易地选择多种平台来使用 Java 语言编程,不同的用户也就可以脱离 Web 浏览器来运行 Java 应用程序,这无疑是很受广大用户欢迎的,也为 Java 语言的应用开拓了极为广阔的前景(当然,各类 JavaOS 之间的兼容性必须得到重视,好在 JavaSoft 已保证将监督这种兼容性)。

另外,由 JavaSoft 推出的完全用 Java 编写的 Internet 上新型浏览器 HotJava,比以前的 Alpha 版更为实用,不仅能编制动态的应用软件,而且能编制完整的成套桌面应用软件,将来还会提供更多的能帮助编制动态应用软件的模块,显然,这也是为 Java 的应用提供了有力的例证。

2003 年 6 月 7 日,由 SUN 公司和台湾经济事务部信息发展局、台湾信息技术研究所等单位牵头,成立了一个“Java 联盟”,参加该联盟的有 22 个在台湾相当著名的计算机公司,并在台北建立一个“Java 开发中心”,在新竹建立一个“Java 语言实验室”,以掀起台湾开发与应用 Java 语言的热潮。香港则在 2003 年 4 月就举行了全岛的 Java 杯比赛,在计算机界掀起了学习 Java 的热潮(尤其是在大学生中,出现了一批 Java 迷)。

有人预言:Java 将是网络上的“世界语”,今后所有的用其他语言编写的软件统统都要用 Java 语言来改写。

1.1.2 Java 语言对软件开发技术的影响

工业界不少人预言:“Java 语言的出现,将会引起一场软件革命”,这是因为传统的软件往往都是与具体的实现环境有关,换了一个环境就需要对代码进行一番改动,耗时费力,而 Java 语言能在执行代码(二进制码)上兼容,这样以前所开发的软件就能运行在不同的机器上,只要所用的机器能提供 Java 语言解释器即可。

Java 语言将对未来软件的开发产生影响,可从如下几个方面考虑:

(1) 软件的需求分析:可将用户的需求进行动态的、可视化描述,以满足设计者更加直观的要求。而用户的需求是各色各样的,但不受地区、行业、部门、爱好的影响,都可以用 Java 语言描述清楚。

(2) 软件的开发方法:由于 Java 语言具有面向目标的特性,所以完全可以用 O-O 的技术与方法来进行开发,这是符合最新的软件开发规范要求的。

(3) Java 语言的动画效果比 GUI 技术更加逼真,尤其是利用 WWW 提供的巨大动画资源空间,可以共享全世界的动态画面的资源。

(4) 软件最终产品:用 Java 语言开发的软件可以具有可视化、可听化、可操作化的交互、动画与动作,要它停就停,要它继续就继续,而这是在电影与电视播放过程中难以做到的。

(5) 其他方面:使用 Java 语言对开发效益、开发价值都有比较明显的影响。

1.1.3 工业界对 Java 语言的评价

1996 年 5 月 29 ~ 31 日,在美国旧金山召开了一个全世界 Java 语言开发者大会(Java One Developer Conference),出席会议的多达 6500 多人,来自工业界的超过一半,有人评价说:

“这是近年来计算机界最光辉的一次盛会”，一些工业界的老总们相当看好 Java 语言，认为它的使用将会引起一场软件革命。从软件的设计风格、设计方法、设计目标到设计过程，都会产生彻底的变革，有人说“甚至会改变整个星球的生活方式”。

在这次会议上，Java 的创始人之一 James Gosling 说：“Java 不仅仅只是 applets，它能做任何事情”；Dta 咨询公司的高级软件工程师 Rich Kadel 说：“Java 不仅仅是一种程序设计语言，更是现代化软件再实现的基础；Java 还是未来新型 OS 的核心；将会出现 Java 芯片；将构成各种应用软件的开发平台与实现环境，是人们必不可少的开发工具……”。

由于各界都看好它，因此，各大公司都纷纷表示支持 Java，Intel、Xerox 公司声言将把 Java 嵌入到他们的产品中去。就连华尔街金融界也在投入资金人力用 Java 开发电子贸易、金融软件。

所以有人说：“现在第三方的开发商都团结在 Java 大旗周围了！”，纷纷推出用 Java 开发的各种软件产品，以期尽快地占领市场。

1.1.4 Java 语言的应用前景

Java 语言有着广泛的应用前景，大体上可以从以下几个方面来考虑其应用：

- (1) 所有面向对象的应用开发，包括面向对象的事件描述、处理、综合等。
- (2) 计算过程的可视化、可操作化的软件的开发。
- (3) 动态画面的设计，包括图形图像的调用。
- (4) 交互操作的设计（选择交互、定向交互、控制流程等）。
- (5) Internet 的系统管理功能模块的设计，包括 Web 页面的动态设计、管理和交互操作设计等。
- (6) Intranet（企业内部网）上的软件开发（直接面向企业内部用户的软件）。
- (7) 与各类数据库连接查询的 SQL 语句实现。
- (8) 其他应用类型的程序。

1.2 Java 语言的特点

Java 是一个广泛使用的网络编程语言，它是一种新的计算概念。

首先，作为一种程序设计语言，它简单、面向对象、不依赖于机器的结构、具有可移植性、鲁棒性、安全性、并且提供了并发的机制、具有很高的性能。其次，它最大限度地利用了网络，Java 的小应用程序（applet）可在网络上传输而不受 CPU 和环境的限制。另外，Java 还提供了丰富的类库，使程序设计者可以很方便地建立自己的系统。

下面分别从这三个方面来讨论 Java 的特点，然后通过与 C、C++ 相比较进一步指出它所具有的优点。

1.2.1 Java 语言的自身特性

Java 语言有下面一些特点：简单性、面向对象、分布式、鲁棒性、安全性、体系结构中立、可移植性、解释执行、高性能、多线程以及动态性。

1. 简单性

Java 语言是一种面向对象的语言，它通过提供最基本的方法来完成指定的任务，只需

理解一些基本的概念,就可以用它编写出适合于各种情况的应用程序。Java 省略了运算符重载、多重继承等模糊的概念,并且通过实现自动垃圾收集大大简化了程序设计者的内存管理工作。另外,Java 也适合于在小型机上运行,它的基本解释器及类的支持只有 40KB 左右,加上标准类库和线程的支持只有 215KB 左右。库和线程的支持也只有 215KB 左右。

2. 面向对象

Java 语言的设计集中于对象及其接口,它提供了简单的类机制以及动态的接口模型。对象中封装了它的状态变量以及相应的方法,实现了模块化和信息隐藏;而类则提供了一类对象的原型,并且通过继承机制,子类可以使用父类所提供的方法,实现了代码的复用。

3. 分布式

Java 是面向网络的语言。通过它提供的类库可以处理 TCP/IP 协议,用户可以通过 URL 地址在网络上很方便地访问其他对象。

4. 鲁棒性

Java 在编译和运行程序时,都要对可能出现的问题进行检查,以避免错误的产生。它提供自动垃圾收集来进行内存管理,防止程序员在管理内存时容易产生的错误。通过集成的面向对象的例外处理机制,在编译时,Java 提示出可能出现但未被处理的例外,帮助程序员正确地进行选择以防止系统的崩溃。另外,Java 在编译时还可捕获类型声明中的许多常见错误,防止动态运行时不匹配问题的出现。

5. 安全性

用于网络、分布环境下的 Java 必须要防止病毒的入侵。Java 不支持指针,一切对内存的访问都必须通过对象的实例变量来实现,这样就防止程序员使用“特洛伊”木马等欺骗手段访问对象的私有成员,同时也避免了指针操作中容易产生的错误。

6. 体系结构中立

Java 解释器生成与体系结构无关的字节码指令,只要安装了 Java 运行时系统,Java 程序就可在任意的处理器上运行。这些字节码指令对应于 Java 虚拟机中的表示,Java 解释器得到字节码后,对它进行转换,使之能够在不同的平台运行。

7. 可移植性

与平台无关的特性使 Java 程序可以方便地被移植到网络上的不同机器。同时,Java 的类库中也实现了与不同平台的接口,使这些类库可以移植。另外,Java 编译器是由 Java 语言实现的,Java 运行时系统由标准 C 实现,这使得 Java 系统本身也具有可移植性。

8. 解释执行

Java 解释器直接对 Java 字节码进行解释执行。字节码本身携带了许多编译时的信息,使连接过程更加简单。

9. 高性能

Java 语言和其他解释执行的语言如 BASIC、TCL 不同,Java 字节码的设计使之能很容易地直接转换成对应于特定 CPU 的机器码,从而获得较高的性能。

10. 多线程

多线程机制使应用程序能够并行执行,而且同步机制保证了对共享数据的正确操作。通过使用多线程,程序设计者可以分别用不同的线程完成特定的行为,而不需要采用全局

的事件循环机制，这样就很容易地实现网络上的实时交互行为。

11. 动态性

Java 的设计使它适合于一个不断发展的环境。在类库中可以自由地加入新的方法和实例变量而不会影响用户程序的执行。并且 Java 通过接口来支持多重继承，使之比严格的类继承具有更灵活的方式和扩展性。

12 丰富的类库

Java 提供了大量的类以满足网络化、多线程、面向对象系统的需要。

(1) 语言包提供的支持包括字符串处理、多线程处理、异常处理、数学函数处理等，可以用它简单地实现 Java 程序的运行平台。

(2) 实用程序包提供的支持包括哈希表、堆栈、可变数组、时间和日期等。

(3) 输入输出包用统一的“流”模型来实现所有格式的 I/O，包括文件系统、网络、输入/输出。

(4) 低级网络包用于实现 Socket 编程。

(5) 抽象图形用户接口包实现了不同平台的计算机的图形用户接口部件，包括窗口、菜单、滚动条、对话框等，使得 Java 可以移植到不同平台的机器。

(6) 网络包支持 Internet 的 TCP/IP 协议，提供了与 Internet 的接口。它支持 URL 连接、WWW 的即时访问，并且简化了用户/服务器模型的程序设计。

1.2.2 Java Applet

Java 语言的特性使它可以最大限度地利用网络。Applet 是 Java 的小应用程序，它是动态、安全、跨平台的网络应用程序。Java Applet 嵌入 HTML 语言，通过主页发布到 Internet。网络用户访问服务器的 Applet 时，这些 Applet 从网络上进行传输，然后在支持 Java 的浏览器中运行。

由于 Java 语言的安全机制，用户一旦载入 Applet，就可以放心地来生成多媒体的用户界面或完成复杂的计算而不必担心病毒的入侵。

虽然 Applet 可以和图像、声音、动画等一样从网络上下载，但它并不同于这些多媒体的文件格式，它可以接收用户的输入，动态地进行改变，而不仅仅是动画的显示和声音的播放。

1.2.3 Java 与 C 及 C++ 的比较

对于变量声明、参数传递、操作符、流控制等，Java 使用了和 C、C++ 相同的传统，使熟悉 C、C++ 的程序员能很方便地进行编程。同时，Java 为了实现其简单、鲁棒、安全等特性，也摒弃了 C 和 C++ 中许多不合理的内容。

1. 全局变量

Java 程序中，不能在所有类之外定义全局变量，只能通过在一个类中定义公用、静态的变量来实现一个全局变量。例如：

```
Class GlobalVar
{
    public static global_var;
}
```

在类 GlobalVar 中定义变量 global_var 为 public static, 使其他类可以访问和修改该变量。

Java 对全局变量进行了更好的封装。而在 C 和 C++ 中, 依赖于不加封装的全局变量常常造成系统的崩溃。

2. Goto

Java 不支持 C、C++ 中的 goto 语句, 而是通过异常处理语句 try, catch, finally 等来代替 C、C++ 中用 goto 来处理遇到错误时跳转的情况, 使程序更可读且更结构化。

3. 指针

指针是 C、C++ 中最灵活, 也是最容易产生错误的数据类型。由指针所进行的内存地址操作常会造成不可预知的错误, 同时通过指针对某个内存地址进行显式类型转换后, 可以访问一个 C++ 中的私有成员, 从而破坏安全性, 造成系统的崩溃。而 Java 对指针进行完全的控制, 程序员不能直接进行任何指针操作, 例如把整数转化为指针, 或者通过指针释放某一内存地址等。同时, 数组作为类在 Java 中实现, 良好地解决了数组访问越界这一 C、C++ 中不作检查的错误。

4. 内存管理

在 C 中, 程序员通过库函数 malloc () 和 free () 来分配和释放内存, C++ 中则通过运算符 new 和 delete 来分配和释放内存。再次释放已释放的内存块或未被分配的内存块, 会造成系统的崩溃; 同样, 忘记释放不再使用的内存块也会逐渐耗尽系统资源。而在 Java 中, 所有的数据结构都是对象, 通过运算符 new 为它们分配内存堆。通过 new 得到对象的处理权, 而实际分配给对象的内存可能随程序运行而改变, Java 对此自动地进行管理并且进行垃圾收集, 有效防止了由于程序员的误操作而导致的错误, 并且更好地利用了系统资源。

5. 数据类型的支持

在 C、C++ 中, 对于不同的平台, 编译器对于简单数据类型如 int、float 等分别分配不同长度的字节数, 例如: int 在 IBM PC 中为 16 位; 在 VAX-11 中为 32 位, 这导致了代码的不可移植性。但在 Java 中, 对于这些数据类型总是分配固定长度的位数, 如对 int 型, 它总占 32 位, 这就保证了 Java 的平台无关性。

6. 类型转换

在 C、C++ 中, 可以通过指针进行任意的类型转换, 常常带来不安全性。而 Java 中, 运行时系统对对象的处理要进行类型相容性检查, 以防止不安全的转换。

7. 头文件

C、C++ 中用头文件来声明类的原型以及全局变量、库函数等, 在大的系统中, 维护这些头文件是很困难的。而 Java 不支持头文件, 类成员的类型和访问权限都封装在一个类中, 运行时系统对访问进行控制, 防止对私有成员的操作。同时, Java 中用 import 语句来与其他类进行通讯, 以便使用它们的方法。

8. 结构和联合

C、C++ 中的结构和联合中所有成员均为公有, 这就带来了安全性问题。Java 中不包含结构和联合, 所有的内容都封装在类中。

9. 预处理

C、C++中用宏定义来实现的代码给程序的可读性带来了困难。在 Java 中，不支持宏，它通过关键字 `final` 来声明一个常量，以实现宏定义中广泛使用的常量定义。

1.3 获取 Java 资源的手段

Java 在互联网上可以说无处不在。从网上得到 Java 的信息是最为方便快捷的，而且往往可以得到最新信息。下面列出几个可供参考的网点：

(1) SUN 公司简介。

<http://www.sun.com>

(2) Java 简介。

<http://java.sun.com> 或 <http://www.javasoft.com>

(3) Java 开发者阵营。

<http://www.sun.com/sunsoft/camps>

(4) SUN 公司网上杂志。

<http://www.sun.com/sunworldonline>

(5) Gamelan。

<http://www.gamelan.com> (列出包含 Java 内容的网址及与网上 Java 资源的链接)

以上都是国外的站点。在中国教育科研网范围内要得到 Java 信息是非常方便的，许多高校的文件服务器上都有 Java 的软件和信息，而且一般比国外发布时间滞后不很多，这里就不一一叙述了。

1.4 Java 语言的新进展

Java 语言发展很快，除了前面提到的之外，在此再介绍如下几个方面：

(1) Just-in-Time 编译器：它能将 Java 字节代码编译成与本机结构相适应的代码，以提高执行速度。

(2) Joe: Joe 是 Java 向客户机/服务器模式的扩展。它是第一个完全用 Java 语言编写的商用对象请求代理软件 (ORB)，遵循对象管理组 (OMG) 的“通用对象请求代理体系结构” (CORBA)。其测试版可以由 <http://www.sun.com/sunsoft/neo> 免费下载。

(3) 类库的发展：SUN 公司将 Java 类库划分为核心 (CORE) 和扩展 (Standard Extension) 两类，前者是可以保证适用于所有 Java 平台的。SUN 与许多第三方软件供应商都在开发新的类库。扩展类库如果发展较完善，就可以被定为核心类库甚至打包到 JDK (Java 开发工具包) 之中。目前正在发展和测试过程中的类库有很多。例如，Java Enterprise API 为 Intranet 的开发而设计；Java Commerce API 用于开发商务应用；Java Management API 用于解决网络管理问题；Java Media API 包含对二、三维图形、图像和语音、动画的处理；Java Embedded API 用于嵌入式设备 (如电话) 的软件开发等。

(4) 开发工具：除了 SUN 公司的 JDK 和 Java WorkShop 之外，还有许多公司研制出了一些 Java 的开发环境，如 Symantec Cafe、Visual Café、Visual J++ 等。甚至有的数据库厂商也推出了自己的 Java 开发环境，如 Informix 的 Jwork 和 Sybase 的 Power J。在这些工

具中, JDK 反映了 Java 的最新进展。虽然它的界面不引起可视化工具友好, 但却是各种其他开发环境的基础。

1.5 Java 简单程序

以上介绍了关于 Java 的历史、影响和发展, 下面介绍 Java 的程序。Java 的程序分为两类:

- (1) 可独立解释执行的应用程序 (Application)。
- (2) 需放入 HTML 文件中依靠浏览器显示结果的 Applet。

在 Java 的开发工具包 JDK (Java Developer's Kit) 环境中能方便地编译、运行和调试这两种程序。这一章将给出应用程序 Application 和 Applet 最简单的程序示例, 同时对 JDK 及其使用进行介绍。

下面先介绍两个简单的 Java 程序, 并对其进行分析。

1.5.1 Application 应用程序

和其他语言一样, Java 可用于创建独立的应用程序, 称为 Application。所谓独立, 是相对于 Applet 对浏览器的依赖而言。实际上它的执行离不开 JDK 中的编译器 Javac 和解释器 java。

下面通过若干完整而简单的 Java 应用程序来了解 Java 应用程序的结构和特点。

```
1  //an application
2  public class HelloWorldApp
3  {
4      public static void main (String args[ ]){
5          System.out.println("Hello World!");
6      }
7  }
```

本程序的作用是输出下面一行信息:

Hello World!

其中:

- (1) //后面为程序的注释部分。
- (2) 用保留字 class 来声明一个新的类, 其类名为 HelloWorldApp, 它是一个公共类 (public)。
- (3) 类 HelloWorldApp 的类定义, 整个类定义由大括号 {} 括起来, 包括变量和方法声明。

public 表明这个方法是可被公开访问的。

void 表明该方法无返回值。

static 表明这是一个类方法。

main 是这个方法的名称。

main 的组成方法的具体语句也必须由花括号将它们与其他方法和变量分隔开。main 方法是一个特殊的方法, 它是每一个应用程序所必需的, 是应用程序解释执行的入口, 必须声明为类方法, 必须带有字符串数组类型的参数, 但参数名可以是任意的。main () 调