

• 游戏开发经典丛书 •

(美) David Wallace Croft 著  
彭 晖 译

# Java 游戏 高级编程

Advanced Java Game Programming



清华大学出版社

# Java 游戏高级编程

(美) David Wallace Croft 著

彭 晖 译

清华大学出版社

北 京

## 内 容 简 介

本书通过丰富的示例游戏,详细讲述了 Java 游戏高级编程方面的知识。包括部署框架、Swing 动画、动画库、高级图形技术、持久性机制、游戏体系结构以及一些通信和同步技术等。

本书针对那些具有一定编程经验、渴望将自己独特的游戏设计思想变成部署实体的 Java 高级开发人员。

EISBN: 1-59059-123-2

Advanced Java Game Programming

David Wallace Croft

Original English language edition published by Apress L. P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA. Copyright ©2004 by Apress L.P. Simplified Chinese-Language edition copyright ©2005 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Apress 出版公司授权清华大学出版社出版。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。

北京市版权局著作权合同登记号 图字: 01-2005-4356

版权所有,翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

Java 游戏高级编程/(美)克罗夫特(Croft, D.W.)著;彭晖译. —北京:清华大学出版社, 2005.8

书名原文: Advanced Java Game Programming

ISBN 7-302-11083-2

I. J… II. ①克… ②彭… III. ①JAVA 语言—程序设计 ②游戏—应用程序—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 050985 号

出版者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社总机:010-62770175 客户服务:010-62776969

组稿编辑:曹 康

文稿编辑:徐燕华

封面设计:康 博

版式设计:康 博

印刷者:北京鑫海金澳胶印有限公司

装订者:三河市化甲屯小学装订二厂

发行者:新华书店总店北京发行所

开 本:185×260 印张:27.5 字数:704 千字

版 次:2005 年 8 月第 1 版 2005 年 8 月第 1 次印刷

书 号:ISBN 7-302-11083-2/TP·7340

印 数:1~4000

定 价:58.00 元

# 前 言

*花代价所换来的一点才智，抵过别人传授的数倍不止。*

——本杰明·富兰克林

本书针对的是那些想使用最新的高性能技术，创建面向 Web 和桌面的复杂 2D 计算机游戏的 Java 程序员。本书循序渐进地介绍了一个可重用的动画库，每一章都依赖于前面的一些章节。在需要说明这些主题的地方，都详细地介绍了每一个新库类的源代码，并且给出了实际使用这些类的示例游戏。免费许可使您可以直接修改和调整库代码和示例游戏，开发自己的游戏程序。

## 目的

本书也可取名为《现代 Java 游戏编程》。在以前出版的 Java 游戏编程的书籍中介绍的很多技术和主题，已被 Java 编程语言和它的高级类库所淘汰。这包括像线程管理、事件管理、图形用户界面、网络通信、持久性和部署等方面。在本书中，我对这些变更作了适当的注释，并简要地对这些技术作了一些对比。

虽然通过将每一帧中需要重绘像素的数量最小化就可以在 Java 中创建高速的动画，但是在最近发布的 Java 编程语言的 1.4 版中引入的新类，又提供了通过可移植方式直接访问视频硬件的方法。正如在本书中介绍的一样，即使每一帧中需要更新的像素很多，也可以实现高速的帧速率。在这里介绍了一个示例程序，用来演示在真彩色高分辨率模式中，成功地将全屏动画和显示器 75Hz 的刷新率同步。

尽管本书最初的设计是面向游戏，但是读者应该知道这里介绍的技术也可以用在其他应用方面，包括广告和仿真。嵌在 Web 页面中的动画调幅广告 applet 会立即闪现在您的大脑中。调幅广告可以使用强大的通用编程语言(例如 Java)编写的这个事实，充分发掘它们超越动画的辅助功能的潜能，包括活动数据连接。在仿真领域，学生和科学家都可以从可视化上获益。当这些模型使用 Java 表示的时候，用户就可以与它们进行交互、更改参数并观察新的结果。

## 读者对象

本书专门针对具有丰富经验、正准备和渴望将他们独特的游戏设计思想转换为部署实体的 Java 开发人员。对这些读者而言，从想了解有趣的游戏特定逻辑以来，一直在学习 Java 编程语言和实现基本的基础结构代码。

同时, 这些 Java 高级开发人员希望详细地了解可能会影响游戏性能的可重用库中的优化选项。在介绍游戏库和本书示例游戏中的这些源代码时, 我假定读者已经掌握了一些基本的主题, 例如 Java 编程语言、面向对象的编程(Object Oriented Programming, OOP)技术、图形用户界面(Graphical User Interface, GUI)组件库, 以及一些设计模式等。

## 范围

本书简要(没有深入地)介绍了那些虽然必需但又不是专门用于 Java 游戏编程的技术。这包括通用的开放源代码开发工具和通用的标准以及应用编程接口(API), 这些都是很多经验丰富的 Java 游戏程序员已经使用过的技术。在书中, 还提供了在这些方面需要进一步阅读的参考书目。

本书介绍的很多可重用库都利用了的核心 Java 平台里可用的最新 API、当前的 J2SE 1.4 版。本书没有介绍那些过时的 API, 例如 AWT(Abstract Window Toolkit)。相反, 深入地研究了应用现代 API 的具体技术, 包括 Java 游戏编程用于动画设计的 Swing。

除了 Java 2D 和 Image I/O API 以外, Java Media API 中绝大部分 API 都没有在本书中介绍, 因为我考虑到它们在游戏开发中会有一些限制。在游戏中使用 Java 3D、JMF(Java Media Framework)或者 Java Speech API, 就强制玩家下载安装一个可选包和它的本地代码实现。我相信这种附加的步骤将会使很多潜在的玩家退缩, 尤其是那些遇到安装问题和下载延迟问题的玩家。尽管已经对 J2SE 内核库的一部分进行了安装, 但是 Java Sound API 仍然需要一个 soundbank 文件, 这个文件对有些操作系统而言是默认包含在 Java 运行库环境(JRE)发布中的, 而对 Windows 而言则不是。Java 共享数据工具包(Java Shared Data Toolkit, JSST)API, 尽管也可以在网络游戏中使用, 但是它们好像已经被废止了。我在游戏编程里, 还没有发现确实需要使用 Java 高级图像(Java Advanced Imaging, JAI)的地方。

本书并没有回避将 Java 与平台专用代码混合的 Java 本地接口(JNI)API 的使用, 但是也不鼓励这样使用。本书中给出的所有代码都是用纯粹、可移植的 Java 语言编写的, 它们能够在所有安装有 Java 虚拟机的平台上运行。作为一个开发人员, 我发现不使用定制的本地库也可以实现高性能。作为一个玩家, 我更加喜欢用纯 Java 编写的游戏, 因为我不必因为安全风险而烦恼, 并且我知道游戏将会运行在我喜欢的平台上, 而不管这是个什么平台。

本书没有介绍使用 Java 2 Micro Edition(J2ME)平台进行 Java 游戏开发。我希望像摩尔定律中描述的那样, 硬件性能的快速发展能使 J2ME 很快过时。如今的手持 PDA 使用的微处理器的性能已经与 3 年以前使用的最好计算机的处理器性能一样强大了。现在在 PDA 上安装 J2SE 和运行 Swing 应用程序已经成为可能。如果在近几年内看到 J2EE 运行在手表上, 我将不会感到惊奇。

在学习后面的关于多玩家联网模式的相关章节时, 确实需要读者具有一些 J2EE 的知识。然而, 这也限制使用 Servlet API 进行开发。更为复杂的 J2EE API(例如 Enterprise Java Beans(EJBs))的使用方法, 本书并没有进行介绍。所有示例的网络游戏都会运行在一个简单的 servlet 容器以内, 并且也不需要安装一个完全的应用服务器。

## 内容简介

在第 1 章“配置开发环境”中，我向读者介绍了将要在整本书中使用的代码库。描述了这个库的设计，并给出了编译示例游戏的说明。可能会在游戏开发环境中使用的一些附加代码、图形和音频文件，也在本章确定了下来，还介绍了软件开发的一些流行工具。以模板的形式给出了一个演示 Java 游戏编程基础知识的游戏示例。

在第 2 章“部署框架”中，我们关注的重点是在可以安装在多种不同类型的平台上的一个框架中部署游戏。在动画线程管理技术的相关内容中，介绍了一个被广泛使用的框架接口以及 applet 生命周期。在对适合于游戏不同发布而进行不同部署的选项进行了一个简要的回顾以后，给出了允许游戏部署在不同开发环境中而不需要任何修改的一个抽象层的源代码示例。

在第 3 章“Swing 动画”中，介绍了一个基于 Swing 的动画库，这个动画库会在整本书中使用。这一章主要考虑的问题是动画的性能和灵活性。在介绍核心动画引擎类的源代码时简单介绍了优化权衡。

在第 4 章“动画库”中，介绍了通用动画任务的一个类集合。这些类提供了很多功能，例如场景管理和 Sprite 实现。它们也用来介绍怎样开发与核心动画引擎类互操作的游戏特定代码。

在第 5 章“高级图形技术”中，介绍了高级图形技术，包括硬件加速图形、多缓冲和全屏独占模式。给出了使这些技术的使用变得更加方便的可重用类和使用这些可重用类的示例游戏。对其中的每一个技术，都介绍了关于它们使用方法的一些限制。

在第 6 章“持久数据”中，对在各种部署框架中加载和保存游戏数据的机制进行了对比。在这个过程中还介绍了可以用于大多数游戏编程的可重用数据持久性的类库。对更加高级的持久性需求的其他选择也进行了考虑。

在第 7 章“游戏体系结构”中，介绍了适合于 Java 游戏开发的面向对象的软件体系结构的优点。使用这种体系结构的一个示例游戏是作为新游戏开发模板进行介绍的。本章还简单介绍一下数据驱动的设计。

在第 8 章“A\*算法”中，介绍了现在在游戏行业中最流行和最常用的一个人工智能(AI)算法的实现。本章还介绍了使用 A\*算法绕过障碍物寻找路径的示例。

在第 9 章“HTTP 隧道技术”中，介绍一个可以在大部分 Java 游戏部署环境里常见的安全限制以内进行操作的一个联网库。本章还给出了一个示例，在这个示例中，数据在服务器和客户机之间进行传递。

在第 10 章“HTTP 轮询机制”中，将联网库扩展以支持在线多玩家游戏。使用轮询来同步客户机和服务器上的游戏状态。在由单玩家游戏向多玩家联网游戏的转换中，介绍了我所推荐的软件体系结构的灵活性。

在第 11 章“HTTP Pulling 机制”中，事件驱动的消息是作为轮询的另一个选择而推荐使用的。使用在前面两章中介绍的这些类，演示了这是怎样在无符号 applet 的安全限制以内实现的。使用这些技术进行进一步开发的一些建议已经超出了本书的范围，这里只进行简单的介绍。

## 本书合作站点

本书应该保持其“高级性”。为此，我专门为这本书建立了一个 Web 站点。从这个站点上，您能够订阅与本书相关的电子邮件列表，包括补充的指南和勘误表。您将能够测试和运行本书所介绍的示例游戏，并下载开放源代码库和用来创建这些游戏的公共领域的多媒体文件。针对老师，我已经在 Creative Commons Attribution License 下发布了我在讲授游戏开发课程中使用的课程提纲、幻灯片和布置给学生的作业。我也提供了我的联系方式，这样您可以将您的建议和意见反馈给我，以便我以后进行更正。

在开始阅读第 1 章以前，请先访问 <http://www.croftsoft.com/library/books/ajgp>，并订阅 *Advanced Java Game Programming* 通知邮件列表。

# 目 录

<b>第 1 章 配置开发环境</b> .....	1
1.1 升级到 Java 1.4 .....	1
1.2 内核类 .....	1
1.3 在线运行 Demo 版程序 .....	2
1.4 深入游戏库 .....	3
1.4.1 croftsoft 目录 .....	3
1.4.2 arc 目录 .....	4
1.4.3 bin 目录 .....	4
1.4.4 doc 目录 .....	4
1.4.5 ext 目录 .....	5
1.4.6 lib 目录 .....	5
1.4.7 lic 目录 .....	5
1.4.8 res 目录 .....	5
1.4.9 src 目录 .....	6
1.4.10 tmp 目录 .....	6
1.5 XML 简介 .....	6
1.6 使用 Ant 编译 .....	7
1.7 使用开放源代码 .....	10
1.7.1 学习版权的基本知识 .....	10
1.7.2 选择一个许可 .....	11
1.7.3 重命名修改后的代码 .....	12
1.7.4 共享源代码 .....	13
1.8 为游戏获取多媒体资源 .....	13
1.8.1 图片资源 .....	13
1.8.2 音频资源 .....	14
1.9 基本示例 .....	15
1.9.1 修改源代码 .....	15
1.9.2 修改构建文件 .....	26
1.10 小结 .....	27
1.11 参考文献 .....	27
<b>第 2 章 部署框架</b> .....	28
2.1 部署为 applet .....	28
2.1.1 实现生命周期方法 .....	28
2.1.2 管理 applet 动画线程 .....	30



2.1.3	读取 JAR 文件 .....	34
2.1.4	使用插件升级客户端 .....	36
2.1.5	了解存在的限制 .....	38
2.1.6	applet 签名 .....	38
2.1.7	缓存 applet .....	39
2.2	部署为可执行 JAR .....	39
2.2.1	生成清单文件 .....	39
2.2.2	对不安全性进行保护 .....	40
2.3	用 Java Web Start 进行部署 .....	40
2.3.1	准备发布文件 .....	41
2.3.2	访问默认浏览器 .....	43
2.3.3	使用反射进行动态链接 .....	46
2.4	将多个 applet 部署为一个 applet .....	47
2.4.1	MultiAppletStup .....	48
2.4.2	MultiAppletNews .....	50
2.4.3	Lifecycle .....	54
2.4.4	LifecycleWindowListener .....	54
2.4.5	MultiApplet .....	58
2.4.6	CroftSoftCollection .....	65
2.5	小结 .....	67
2.6	参考文献 .....	68
<b>第 3 章</b>	<b>Swing 动画 .....</b>	<b>69</b>
3.1	ComponentAnimator .....	69
3.1.1	更新和绘制阶段 .....	70
3.1.2	精灵的定义 .....	70
3.1.3	ExampleAnimator .....	71
3.2	RepaintCollector .....	73
3.2.1	Swing 串行化 .....	73
3.2.2	动画的问题 .....	73
3.2.3	RepaintCollector .....	75
3.2.4	SimpleRepaintCollector .....	75
3.2.5	BooleanRepaintCollector .....	77
3.2.6	CoalescingRepaintCollector .....	78
3.2.7	其他实现 .....	81
3.3	LoopGovernor .....	81
3.3.1	固定的延迟 .....	82
3.3.2	帧速率同步 .....	84
3.3.3	SamplerLoopGovernor .....	85
3.3.4	WindowedLoopGovernor .....	85
3.4	AnimatedComponent .....	90
3.5	小结 .....	98

3.6	参考文献 .....	98
<b>第4章</b>	<b>动画库 .....</b>	<b>99</b>
4.1	ComponentPainter 实现 .....	99
4.1.1	NullComponentPainter .....	99
4.1.2	ArrayComponentPainter .....	99
4.1.3	ColorPainter .....	101
4.1.4	SpacePainter .....	103
4.1.5	TilePainter .....	106
4.2	ComponentUpdater 实现 .....	114
4.2.1	NullComponentUpdater .....	114
4.2.2	ArrayComponentUpdater .....	115
4.2.3	EdgeScrollUpdater .....	115
4.3	ComponentAnimator 实现 .....	120
4.3.1	NullComponentAnimator .....	120
4.3.2	TileAnimator .....	120
4.3.3	FrameRateAnimator .....	122
4.3.4	CursorAnimator .....	126
4.4	Sprite 实现 .....	131
4.4.1	Sprite .....	131
4.4.2	AbstractSprite .....	132
4.4.3	IconSprite .....	135
4.4.4	BounceUpdater .....	137
4.4.5	IconSequenceUpdater .....	141
4.5	小结 .....	143
<b>第5章</b>	<b>高级图形技术 .....</b>	<b>144</b>
5.1	硬件加速图像 .....	144
5.1.1	Image .....	144
5.1.2	BufferedImage .....	145
5.1.3	VolatileImage .....	145
5.1.4	自动图像 .....	146
5.1.5	兼容图像 .....	146
5.1.6	缓存算法 .....	149
5.2	多缓冲和多线程 .....	150
5.3	全屏独占模式 .....	160
5.3.1	启用全屏模式 .....	160
5.3.2	FullScreenToggler .....	161
5.3.3	配置帧 .....	165
5.3.4	改变显示模式 .....	165
5.3.5	DisplayModeLib .....	166

5.3.6	GraphicsDeviceLib .....	166
5.3.7	消除锯齿 .....	167
5.3.8	BufferStrategyAnimatedComponent .....	168
5.3.9	FullScreenDemo .....	170
5.3.10	使用独占模式的顾虑 .....	177
5.4	小结 .....	177
5.5	参考文献 .....	177
<b>第 6 章</b>	<b>持久数据 .....</b>	<b>178</b>
6.1	数据格式 .....	178
6.1.1	对象串行化 .....	178
6.1.2	属性 .....	181
6.1.3	XML .....	182
6.1.4	瓦片地图图像 .....	190
6.1.5	随机种子 .....	193
6.2	持久性机制 .....	195
6.2.1	JAR 资源文件 .....	196
6.2.2	用户主目录文件 .....	197
6.2.3	JNLP 持久性 .....	200
6.2.4	applet 持久性 .....	203
6.2.5	稳固持久性 .....	206
6.2.6	嵌入式数据库 .....	211
6.2.7	服务器端的持久性 .....	212
6.3	数据完整性 .....	214
6.3.1	消息摘要 .....	214
6.3.2	散列缓存 .....	215
6.4	小结 .....	217
6.5	参考文献 .....	217
<b>第 7 章</b>	<b>游戏体系结构 .....</b>	<b>218</b>
7.1	接口 .....	218
7.2	继承 .....	218
7.3	目标 Mars .....	221
7.4	模型-视图-控制器 .....	221
7.4.1	模型 .....	222
7.4.2	多重接口继承 .....	224
7.4.3	视图 .....	234
7.4.4	控制器 .....	240
7.5	复合 MVC .....	243
7.5.1	复合模型 .....	243
7.5.2	复合视图 .....	254

7.5.3	复合控制器 .....	260
7.5.4	将三者进行组合 .....	260
7.6	数据驱动设计 .....	264
7.6.1	AnimationInit .....	264
7.6.2	AnimatedApplet .....	265
7.7	小结 .....	269
7.8	参考文献 .....	269
<b>第 8 章</b>	<b>A*算法 .....</b>	<b>270</b>
8.1	Cartographer .....	271
8.2	NodeInfo .....	272
8.3	AStar .....	273
8.4	AStarTest .....	279
8.5	SpaceTester .....	283
8.6	GridCartographer .....	283
8.7	GradientCartographer .....	286
8.8	TankConsole .....	289
8.9	TankOperator .....	290
8.10	StateSpaceNode .....	291
8.11	TankCartographer .....	292
8.12	DefaultTankOperator .....	293
8.13	PlayerTankOperator .....	297
8.14	小结 .....	299
8.15	参考文献 .....	300
<b>第 9 章</b>	<b>HTTP 隧道技术 .....</b>	<b>301</b>
9.1	测试示例 .....	302
9.2	可重用的客户端代码 .....	303
9.2.1	Encoder .....	304
9.2.2	Parser .....	304
9.2.3	StreamLib .....	304
9.2.4	StringCoder .....	305
9.2.5	HttpLib .....	306
9.2.6	Queue .....	308
9.2.7	ListQueue .....	309
9.2.8	Loopable .....	312
9.2.9	Looper .....	313
9.2.10	HttpMessagePusher .....	315
9.3	游戏特定的客户端代码 .....	318
9.4	可重用的服务器端代码 .....	323
9.4.1	Server .....	323

9.4.2	HttpGatewayServlet .....	324
9.5	游戏特定的服务器端代码 .....	327
9.6	打包 WAR .....	330
9.6.1	web.xml .....	330
9.6.2	build.xml .....	331
9.7	小结 .....	332
9.8	参考文献 .....	332
<b>第 10 章</b>	<b>HTTP 轮询机制 .....</b>	<b>333</b>
10.1	测试示例程序 .....	333
10.2	可重用的客户端代码 .....	334
10.2.1	SerializableCoder .....	334
10.2.2	HttpMessagePoller .....	335
10.2.3	Consumer .....	340
10.2.4	QueuePuller .....	340
10.2.5	HttpMessageClient .....	342
10.3	游戏特定的客户端代码 .....	345
10.3.1	Request .....	345
10.3.2	AbstractRequest .....	346
10.3.3	FireRequest .....	346
10.3.4	MoveRequest .....	347
10.3.5	ViewRequest .....	348
10.3.6	GameData .....	348
10.3.7	Synchronizer .....	350
10.3.8	NetController .....	354
10.3.9	NetMain .....	356
10.4	可重用的服务器端代码 .....	357
10.4.1	XmlBeanCoder .....	358
10.4.2	SerializableLib .....	359
10.5	游戏特定的服务器端代码 .....	360
10.5.1	GameInit .....	360
10.5.2	Player .....	362
10.5.3	NetGame .....	363
10.5.4	MarsServer .....	370
10.5.5	MarsServlet .....	376
10.6	小结 .....	378
10.7	参考文献 .....	378
<b>第 11 章</b>	<b>HTTP Pulling 机制 .....</b>	<b>379</b>
11.1	测试示例 .....	379
11.2	可重用的客户端代码 .....	381

11.2.1	Authentication .....	381
11.2.2	Id .....	381
11.2.3	LongId .....	382
11.2.4	ModelId .....	383
11.2.5	SeriModelId .....	383
11.3	游戏特定的客户端代码 .....	384
11.3.1	Request .....	384
11.3.2	CoalesceableRequest .....	384
11.3.3	ChatController .....	385
11.3.4	Response .....	386
11.3.5	CreateUserConsumer .....	386
11.3.6	Event .....	387
11.3.7	ChatClient .....	387
11.3.8	ChatSynchronizer .....	393
11.3.9	ChatPanel .....	395
11.3.10	ChatApplet .....	398
11.4	服务器端代码 .....	400
11.4.1	User .....	400
11.4.2	UserStore .....	401
11.4.3	PullServer .....	402
11.4.4	MoveServer .....	403
11.4.5	SeriChatGame .....	404
11.4.6	ChatServer .....	409
11.5	跟踪消息 .....	411
11.5.1	多玩家联网模式 .....	412
11.5.2	单玩家本地模式 .....	413
11.6	扩展示例 .....	414
11.7	小结 .....	414
11.8	参考文献 .....	415
附录 A	源代码索引 .....	416
附录 B	CVS 简介 .....	420
B.1	检出代码 .....	420
B.2	创建自己的项目 .....	421
B.3	并行程序设计 .....	423
B.3.1	实施代码所有权 .....	423
B.3.2	互相监视 .....	424
B.3.3	创建分支 .....	425
B.4	参考文献 .....	425

# 第 1 章 配置开发环境

身教胜于言传。

——本杰明·富兰克林

本章主要介绍怎样配置开发环境，以及怎样使用本书所介绍的游戏库和示例程序，创建自己的游戏程序。

## 1.1 升级到 Java 1.4

编译和运行本书的示例游戏，需要下载安装 Java 2 软件开发工具箱(Software Development Kit, SDK)和 1.4 以上版本的 Java 2 标准版(Java 2 Standard Edition, J2SE)。其中的 Java 2 指的是 Java 2 开发平台，而 Java 1.4 指的是 Java 2 开发平台的某一个具体版本。尽管可以去责怪造成这种混乱的 SUN Microsystems 的营销人员，但是事实确实如此。

由于本书的示例游戏代码都使用了 J2SE 1.4 版本中新增的图形加速类，所以只能在 J2SE 1.4 以上版本中运行这些示例代码。J2SE 1.4 版本中新增的那些类极大地提高了 Java 2 的动画性能。在编著本书的时候，J2SE 1.4 版已经可以在多个操作系统平台上运行了，这些操作系统包括 Linux、Mac OS X、Solaris 和 Windows。如果没有 1.4 版以上的 J2SE，可以立即到 <http://java.sun.com> 网站去下载，然后将它安装在包括 Apple 在内的那些常用开发平台上。Apple 的开发人员还可以从 <http://developer.apple.com/java/> 网站下载 Java 的最新版本。

游戏代码应该是完全不需要进行修改就可以直接在支持 J2SE 1.4 版的各种平台上运行。话虽然是这么说的，但是这里还是要提醒您，本人只在 Linux 系统和 Windows 系统上对代码进行了测试。只有根据使用 Mac OS X 和 Solaris 系统的读者对该书的反馈意见，我才能知道这些代码对这两种操作系统的适应情况。

## 1.2 内核类

J2SE 定义了一个标准的应用编程接口(API)库，这个标准库中包含数千个类，这些类又被集合到内核包中。在每一个 J2SE 的兼容系统中，这些内核类肯定都是预安装的。这样可以使开发人员可以更加轻松，因为它减少了客户机需要下载和安装的代码量。Java.lang 就是这样的一个内核包，该内核包包含了绝大部分 Java 平台所需要的基本类。

在目标平台上，非内核类一般都不是预安装的，因此，在使用这些非内核类的时候，需要进行慎重地考虑。这些非内核类一般都称为可选包(optional package)。最近也才将它们通称为标准扩展(standard extension)包。之所以说它们是“扩展”的，是由于它们不是和内核类库一起发布的；之所以说它们是“标准”的，是由于它们为与其底层实现无关的特殊功能定义了标准的接口。javax.media.j3d 就是一个可选包，它是一个专门用于渲染 3D 场景画面的类库。

由于内核包的名称都是以前缀 Java 开头，而标准扩展包的名称都是以前缀 Javax 开头，所以我们很容易区分哪一个包是内核包，哪一个包是标准扩展包。但是，这是不久以前的情况，而现在在内核类库中也包含有以非 java(例如 javax.swing 包)开头的包。我不清楚 Sun Microsystems 为什么会改变这种习惯的命名方法，但是我猜测这与最近微软的官司可能会有些关系。我知道在这种更改决定做出以后，总会不断地迷惑一些甚至是高水平的 Sun 工程师。在某一些具体的问题上，他们可能将必须检查整个 Java 库。也许他们会将这个 Java 称为 Java 3 平台的 2.0 版。如果他们要是真的这样做，我还是希望他们能够考虑将包的命名规则恢复为它原来的那种习惯约定。

大部分示例游戏代码都只依赖于内核类。这保证了它们在 J2SE 平台上不再需要进行另外的安装就能够运行。然而，也有一些游戏还要依赖于可选包。在介绍这些示例游戏的时候，我会提示您。

J2SE 用于较小平台(例如嵌入式设备)的小型版是 Java 2 Micro Edition(J2ME)。J2SE 用于那些繁重任务平台(例如应用服务器)的大型版是 Java 2 Enterprise Edition(J2EE)。本书不包括 J2ME 的内容，对 J2EE 的内容也只涉及到其冰山一角。在需要使用 J2EE 代码的地方，我事先会提示您；在需要使用可选包的时候，您只能和我采取一样的方法。

### 1.3 在线运行 Demo 版程序

本书 Web 站点的 URL 是 <http://www.croftsoft.com/library/books/ajgp/>。在该站点上，可以通过预编译和在线安装测试 Demo 版程序，以便对有些内容首先有个亲身的体验。我还会不断地更新这个 Demo。请访问本书的 Web 站点以获取其最新版本的内容，也请订阅邮件列表以获取更新的通告邮件。

这个 Demo 包含有几个 Java 动画程序，其中包括游戏程序和模拟仿真程序。可以从左到右单击屏幕顶部的标签，以浏览整个列表。其中最有趣的一个程序就是 Sprite(精灵)，这是一个“精灵”动画测试程序。如图 1-1 所示，可以更改选项和参数，观察动画执行的不同效果。



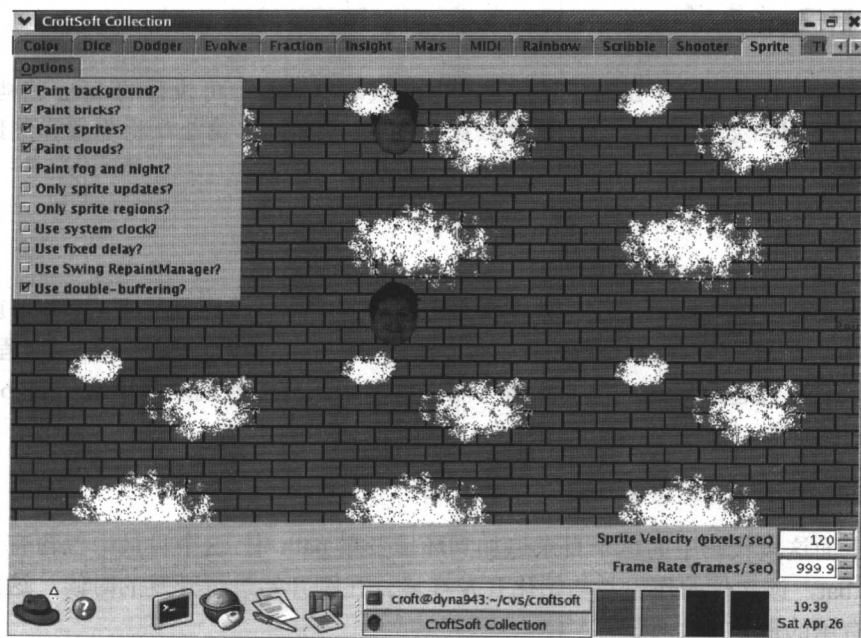


图 1-1 Sprite 程序的 Demo 版

## 1.4 深入游戏库

可以从本书的站点上以压缩归档文件的形式，为游戏库下载源代码、图形和音频文件的快照(snapshot)。该代码快照中包含本书归档的版本号。还可以通过 CVS 客户端，下载这些代码的最新版。其他一些更详细的信息，请参见附录 B。

下载并解压缩以后，就可以使用这些源代码编译和运行示例游戏。下面几节会详细介绍源代码库的目录结构。

### 1.4.1 croftsoft 目录

该归档文件内部的顶层目录是 `croftsoft/`。该目录中包含主要的构建文件(build file)、`readme.txt` 文件和游戏库的子目录。下面列表详细列举了该游戏库子目录的结构。有些子目录并没有包含在归档文件(ZIP 文档)中，但是，在编译示例游戏的时候，会自动创建它们。

- `arc/`——归档包
- `bin/`——二进制文件和实用程序
- `doc/`—— javadoc 文档
- `ext/`——扩展库
- `lib/`——编译后的类库
- `lic/`——许可
- `res/`——资源文件
- `src/`——源代码
- `tmp/`——临时目录