

游戏编程大师系列

J2ME MIDP

手机 游戏程序设计



本书内附
范例光盘

米川英树 著 博硕文化 译

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

J2ME MIDP

手机

游戏
程序

设计

米川英树 著
博硕文化 译

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

北京市版权局著作权合同登记号: 01-2003-5907 号

版 权 声 明

本书中文简体字版经 Japan UNI Agency, Inc., 由株式会社 技术评论社授权中国铁道出版社出版(2004), 任何单位或个人未经出版者书面允许不得以任何手段复制或抄袭本书内容。

Java2ME MIDP ゲームクリエイターズガイド-J-PHONE KDDI Kanzen Taio

Copyright©2002 by 米川英树

Simplified Chinese translation rights arranged with Gijutsu-Hyoron Co.,Ltd., Tokyo
through Japan UNI Agency, Inc., Tokyo

图书在版编目(CIP)数据

J2ME MIDP 手机游戏程序设计 / (日) 米川英树著; 博硕文化译. —北京: 中国铁道出版社, 2004.12

(游戏编程大师系列)

ISBN 7-113-06319-5

I. J... II. ①米...②博... III. ①JAVA 语言-程序设计②移动通信-携带电话机-游戏-应用程序-程序设计 IV. ①TP312②G899

中国版本图书馆CIP数据核字(2004)第139371号

书 名: J2ME MIDP 手机游戏程序设计

作 者: 米川英树

出版发行: 中国铁道出版社(100054, 北京市宣武区右安门西街8号)

策划编辑: 严晓舟 郭毅鹏

责任编辑: 苏 茜 翟玉峰 王占清

封面设计: 清风书坊 江 涛

印 刷: 北京市兴顺印刷厂

开 本: 787×1092 1/18 印张: 23.5 字数: 559 千

版 本: 2005年1月第1版 2005年1月第1次印刷

印 数: 1~5000 册

书 号: ISBN 7-113-06319-5/TP·1397

定 价: 42.00 元

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

出版说明

本书完整介绍 J2ME 游戏程序开发的要诀，内容包括 J2ME 基础知识、手机程序设计方式、游戏设计技巧和应用实例分析。全书详细地介绍了 J2ME 程序设计原理和方法，并通过详实的实例及“重点”的提示，为读者讲解游戏开发所需要注意的地方，相信读者只要有基本的 Java 知识，就能掌握书中所讲述的内容。

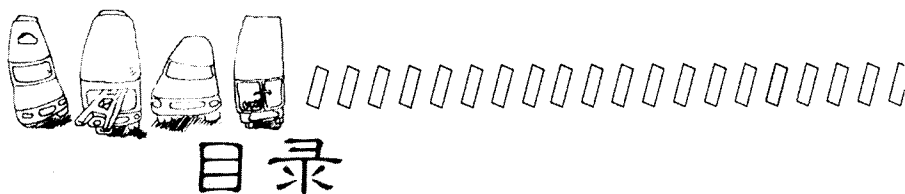
随书所附光盘中 CH-sample 文件夹的内容为书中所涉及的简体中文版的相关范例。

本书由技术评论社通过 Japan UNI Agency, Inc. 提供版权，并由博硕文化翻译，由中国铁道出版社计算机图书中心审选。宋美娜、张益、房雅丁、刘鑫、宋融冰等同志完成本书的整稿工作；在后续的工作中，林胜利同志给予我们很大的帮助，在此一并表示感谢。

2004 年 12 月

目 录

Chapter 1 Java 的下一个领域为移动电话	1
1-1 用移动电话来执行 Java	2
1-1-1 将Java放到移动电话上	2
1-1-2 什么是移动电话的Java	2
1-1-3 什么是J2ME	3
1-1-4 J2SE与CLDC的差异	5
1-2 日本国内的两种移动 Java	8
1-2-1 Profile的种类	8
1-2-2 什么是MIDP	9
1-2-3 什么是DoJa	10
1-2-4 MIDP与DoJa的差异	11
1-3 扩展 API	12
1-3-1 扩展API是指什么	12
1-3-2 J-PHONE的扩展性能	12
1-3-3 KDDI的扩展性能	13
1-3-4 两家电信商扩展API的比较	14
1-4 移动 Java 的优缺点与今后的动向	14
Chapter 2 MIDP 开发的基础与生命周期	17
2-1 MIDlet 开发环境的准备	18
2-1-1 开发环境的准备	18
2-1-2 JDK与J2MEWTK的安装	18
2-1-3 文档的准备	23
2-2 MIDlet 的基础与其生命周期	23
2-2-1 MIDlet的构成	23
2-2-2 MIDlet的规则与抽象方法	24
2-2-3 MIDlet的生命周期	25
2-2-4 MIDlet的运行流程	26
2-3 MIDlet 属性	27

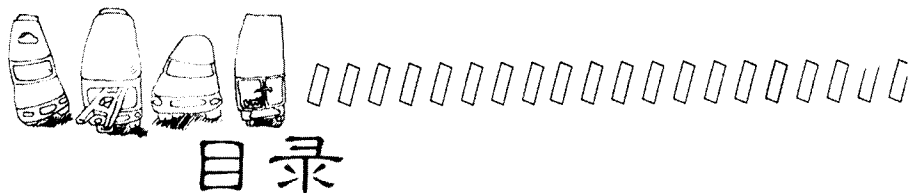


● 2-3-1 编写在JAD文件与列表文件中的属性	27
● 2-3-2 JAD文件编写内容的获取	31
2-4 实际的开发	31
● 2-4-1 开发的流程	31
● 2-4-2 开发时的注意点	32
● 2-4-3 项目的创建	33
● 2-4-4 编写程序代码	37
● 2-4-5 编译	38
● 2-4-6 运行前验证处理	39
● 2-4-7 打包与JAD文件创建	40
● 2-4-8 运行	42
● 2-4-9 调试与Web服务器的设置	43
2-5 MIDlet 的结束与暂停	
● 2-5-1 MIDlet的结束	44
● 2-5-2 MIDlet的强制停止	45
● 2-5-3 MIDlet的选择画面	46
Chapter 3 深入了解 MIDP 规范	47
3-1 了解 MIDlet 的画面显示	48
3-2 了解低级绘图	49
● 3-2-1 试着使用低级绘图方法	49
● 3-2-2 锚点	54
● 3-2-3 颜色的指定	56
● 3-2-4 字体的指定	56
● 3-2-5 图像的获取	60
3-3 了解 UI 组件	62
● 3-3-1 UI组件的使用	62
● 3-3-2 Item类的子类	65
3-4 了解 MIDlet 的按键事件	68
● 3-4-1 按键的种类	68
● 3-4-2 软件按键与命令的事件处理	69

● 3-4-3 数字按键与符号按键的事件处理	73
● 3-4-4 十字按键与确定按键的事件处理	76
● 3-4-5 指针事件	79
3-5 了解简单的声音功能	80
3-6 了解数据存储功能	81
● 3-6-1 RMS (Record Management System)	81
● 3-6-2 记录的添加、删除和更改	82
● 3-6-3 记录的ID	83
● 3-6-4 使用了RecordStore的范例	85
3-7 了解网络功能	90
● 3-7-1 Generic Connection架构	90
● 3-7-2 connector类	91
● 3-7-3 Connection接口	92

Chapter 4 制作动画 95

4-1 了解动画与线程	96
● 4-1-1 动画的基础	96
● 4-1-2 线程的使用	96
● 4-1-3 线程的启动与停止	97
● 4-1-4 线程指定时间的等待	99
4-2 动画的创建	99
● 4-2-1 类结构	99
● 4-2-2 项目的创建与图像的准备	101
● 4-2-3 AnimeMIDlet类的设置	101
● 4-2-4 AnimeCanvas类的设置	102
● 4-2-5 动画的运行	105
4-3 了解 DoubleBuffer	105
● 4-3-1 什么是DoubleBuffer	105
● 4-3-2 DoubleBuffer的使用	106
4-4 了解 Canvas 类的运行特性	107



目录

Chapter 5 制作拼图板益智游戏 109

5-1 游戏的了解与开发的准备	110
● 5-1-1 拼图板益智游戏的规则与了解	110
● 5-1-2 游戏的流程与类结构	111
● 5-1-3 考虑默认画面大小	113
● 5-1-4 开发的准备	113
5-2 实现类	114
● 5-2-1 NumberSliderPuzzle类的实现	114
● 5-2-2 Timer类与TimerTask类	116
● 5-2-3 GameTimer类的实现	118
● 5-2-4 GameCanvas类的实现	119
● 5-2-5 游戏的状态	120
● 5-2-6 Constructor	121
● 5-2-7 准备各关卡的谜题	122
● 5-2-8 初始化处理	123
● 5-2-9 命令的种类	125
● 5-2-10 命令处理	125
● 5-2-11 符合MIDlet的LiveCycle的处理	127
● 5-2-12 计时器处理	129
● 5-2-13 处理的中心部分	131
● 5-2-14 按键事件的处理	133
● 5-2-15 过关检查	137
● 5-2-16 暂停与重新启动处理	138
● 5-2-17 描绘处理	138
5-3 本章总结	145

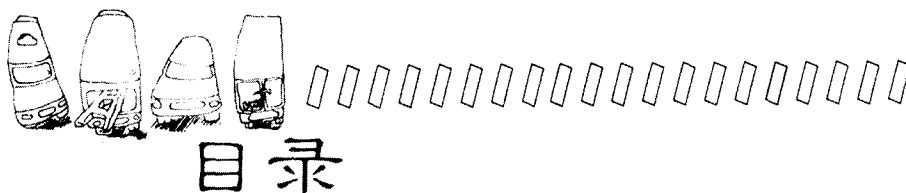
Chapter 6 制作俄罗斯方块益智游戏 147

6-1 游戏的了解与开发的准备	148
● 6-1-1 俄罗斯方块益智游戏的规则与了解	148
● 6-1-2 游戏的流程与类结构	149

6-2 实现类	151
● 6-2-1 Cube类的实现	151
● 6-2-2 Piece类的实现	152
● 6-2-3 GameCanvas类的实现	155
● 6-2-4 游戏的状态	156
● 6-2-5 Constructor	156
● 6-2-6 初始化处理	157
● 6-2-7 命令的种类与命令处理	159
● 6-2-8 配合MIDlet的LiveCycle的处理	161
● 6-2-9 方块显示区域	163
● 6-2-10 落下方块组	164
● 6-2-11 制作下一个要落下的方块组	166
● 6-2-12 按键事件处理	169
● 6-2-13 处理的核心部分	179
● 6-2-14 方块的落下速度的调整	191
● 6-2-15 暂停处理与重新启动处理	193
● 6-2-16 描绘处理	194
6-3 本章总结	199

Chapter 7 制作卡片游戏..... 201

7-1 游戏的了解与开发的准备	202
● 7-1-1 了解扑克牌游戏机的规则	202
● 7-1-2 游戏的流程	203
● 7-1-3 类结构	204
● 7-1-4 开发的准备	205
7-2 实现类	206
● 7-2-1 PokerMachine类的实现	206
● 7-2-2 Card类的实现	208
● 7-2-3 扑克牌的种类	209
● 7-2-4 扑克牌的状态	211
● 7-2-5 扑克牌的描绘	212



● 7-2-6 GameCanvas类的实现	216
● 7-2-7 游戏的状态	217
● 7-2-8 Constructor	218
● 7-2-9 初始化处理	219
● 7-2-10 命令的种类与命令处理	221
● 7-2-11 配合MIDlet的LiveCycle处理	222
● 7-2-12 处理的核心部分	223
● 7-2-13 OK按钮的处理	226
● 7-2-14 按键事件处理	227
● 7-2-15 下注处理	229
● 7-2-16 分配新扑克牌的处理	231
● 7-2-17 扑克牌选择处理	232
● 7-2-18 扑克牌交换处理	233
● 7-2-19 结果判定处理	235
● 7-2-20 描绘处理	245
7-3 本章总结	254

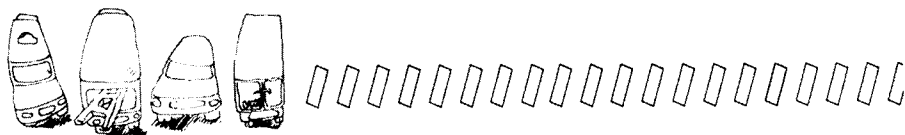
Chapter 8 制作射击游戏..... 257

8-1 游戏的了解与开发的准备	258
● 8-1-1 射击游戏的规则与了解	258
● 8-1-2 游戏的流程与类结构	258
● 8-1-3 开发的准备	260
8-2 实现类	262
● 8-2-1 SpaceShooter类的实现	262
● 8-2-2 了解Sprite功能	263
● 8-2-3 Sprite类的实现	263
● 8-2-4 MyShip类的实现	267
● 8-2-5 MyMissile类的实现	269
● 8-2-6 UFO类的实现	270
● 8-2-7 UFOMissile类的实现	275
● 8-2-8 GameCanvas类的实现	276

● 8-2-9 游戏状态	277
● 8-2-10 Constructor	278
● 8-2-11 与Sprite有关的实例变量	279
● 8-2-12 初始化处理	281
● 8-2-13 命令的种类与命令的处理	282
● 8-2-14 配合MIDlet的LiveCycle处理	284
● 8-2-15 处理的核心部分	286
● 8-2-16 UFO的出现处理	293
● 8-2-17 按钮按键处理	295
● 8-2-18 暂停与重新启动处理	298
● 8-2-19 描绘处理	298
8-3 本章总结	303

Chapter 9 制作格斗游戏 305

9-1 游戏的了解与开发的准备	306
● 9-1-1 拳击游戏规则的了解	306
● 9-1-2 游戏的流程与类结构	307
● 9-1-3 开发的准备	308
9-2 实现类	310
● 9-2-1 PoketBoxing类的实现	310
● 9-2-2 Boxer类的实现	311
● 9-2-3 拳击手的状态	313
● 9-2-4 拳击手的初始化处理	315
● 9-2-5 拳击处理与生命点数处理	315
● 9-2-6 描绘处理	317
● 9-2-7 Enemy类的实现	319
● 9-2-8 GameCanvas类的实现	322
● 9-2-9 游戏的状态	323
● 9-2-10 Constructor	324
● 9-2-11 初始化处理	325
● 9-2-12 命令的种类与命令处理	327



目录

● 9-2-13 配合MIDlet的LiveCycle处理	329
● 9-2-14 处理的核心部分	330
● 9-2-15 更改对手拳击手的状态	334
● 9-2-16 生命点数的检查与回合结束处理	342
● 9-2-17 按键事件的处理	346
● 9-2-18 暂停与重新启动处理	349
● 9-2-19 描绘处理	349
9-3 本章总结	354
Chapter 10 开发秘诀	355
10-1 游戏开发的秘诀	356
10-2 将文件缩小	358
● 10-2-1 混淆器	359
● 10-2-2 使用了RetroGuard的文件缩小化	360

J2ME MIDP

手机

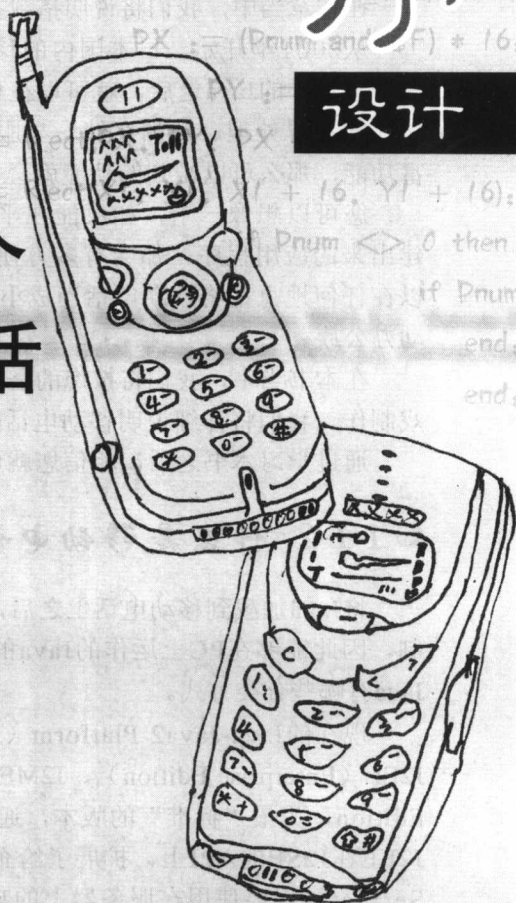
游戏程序

CHAPTER

1

设计

Java 的下一个 领域为移动电话



```
as.CopyMode := cmSrcCopy;
```

```
ap.Canvas.CopyRect(Rect_D, Load_Bmap.Canvas, Rect_L);
```



1-1 用移动电话来执行Java

● 1-1-1 将Java放到移动电话上

在本章当中，我们将说明搭载在移动电话上的Java的基础知识。

从2001年开始，日本国内的移动电话开始搭载了Java。使用了本来就是程序语言一种的Java之后，就可以在移动电话上做到以往静态的HTML内容所无法达成的计算处理、数据存储、与服务器的通信等，如果能利用Java上的其他丰富功能，那么可以做到的功能就会更多了。

您可以想像一下，如果能在平常都会随身携带的移动电话上执行自己所创建出来的应用程序，那会有多方便呢？是不是让您很心动呢？这样一来您就可以在任何地点都将移动电话当成小型PC来使用，早一步领先别人成为真正的移动办公族。

在本书当中，我们将按您的需求来扩展移动电话上的Java，学习如何进行游戏制作。我们将详细说明移动电话的程序设计与执行。

通过学习本书之后，相信您就可以自己亲手来制作出独特风格的游戏了。

● 1-1-2 什么是移动电话的Java

将Java加载到移动电话上之后，由于在移动电话上会有存储器等硬件上的限制，因此能够在PC上运作的Java的规范将无法直接使用。在这里我们先来看看Java有哪些分类方式。

现在的Java-Java2 Platform (J2) 大致可分为J2SE (Standard Edition)、J2EE (Enterprise Edition)、J2ME (Micro Edition) 三类。J2SE其名为Standard Edition，为最“标准”的版本，通常在PC上所使用的Java，就是指这个J2SE。J2EE在J2SE的API上，扩展了给企业使用的EJB (Enterprise Java Beans) 与Servlet等主要使用在服务器上的功能。J2ME则是为了对应家电或嵌入式设备等，而从J2SE中去掉一些功能的版本。而搭载在移动电话上的，就是这个J2ME。



■ Java的分类（见图1-1）

* J2SE（Standard Edition）

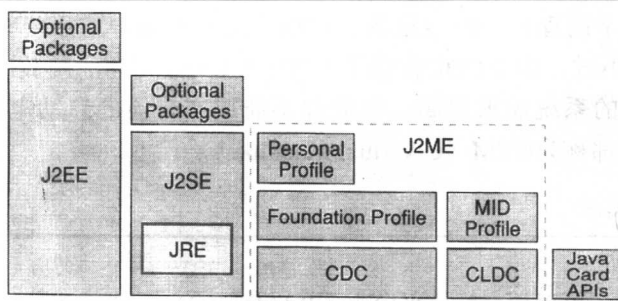
J2的标准版，通常使用在PC上。

* J2EE Enterprise Edition

扩展了给企业使用的EJB与Servlet等功能。

* J2ME Micro Edition

从J2SE中取出适用于家电或嵌入式设备功能的版本，为在移动电话上所使用的Java版本。



【图1-1】 Java的版本

● 1-1-3 什么是J2ME

那么，我们现在就来看看搭载在移动电话上的J2ME吧。

J2ME的目标是家电与嵌入式设备。这类设备有许多种类，对应这些设备的J2ME当中，被定义了CDC（Connected Device Configuration）、CLDC（Connected Limited Device Configuration）这两个Configuration。

Configuration所提供的类函数库，只有java.lang、java.io、java.util包等J2SE核心类的Subset（子集）。有关应用程序有效期间的管理（应用程序的安装、启动以及删除）、用户界面的灵活性、事件处理、高级应用程序模块等，则是以在配置固有的Profile中追加函数库的方式来提供。也就是说，Configuration只提供了基本的核心函数库与Java VM，然后再依据定义的设备Profile在其上附加GUI与事件处理、描述等来让Java运行。



■ Configuration与Profile

* Configuration

提供了基本的核心函数库与JavaVM部分。要作为应用程序来运作时还需要能在Configuration上运作的Profile。

* Profile

定义了各设备的GUI或事件处理、描述等功能，为在Configuration运行的部分。使用了Profile之后就可以当做应用程序来运行。

Configuration其中之一的CDC，是以能用在个人网关、下一代移动电话、PDA（个人数字助理）、家电设备、POS终端、车辆导航系统等上运行为前提设计出来的。不过，由于CDC提供了与JDK1.1大致相同的函数库，而导致存储器与GUI部分的系统需求很高，因此并不能用在目前的移动电话上。CDC的JavaVM，一般都称为CVM（C Virtual Machine）。

■ CDC系统需求

* 32位处理器

* 有2MB以上、可以用在Java平台上的RAM或ROM

* 可以充分运行Java 2 Virtual Machine功能的设备

* 可以不连续地以9600bit/s以下的带宽来连接网路

* 先进的用户界面

以JDK1.1为基础，也以同样的设备为目标的PersonalJava，则是以可以在CDC运作的Personal Profile方式来提供，因而以J2为基础。另外，CDC则是以下所要介绍的CLDC的Super Set（超级集合）。

CLDC，是以能使用在移动电话、双向对讲机、PDA（个人数字助理）、家电、POS终端等上面为前提设计出来的。比起CDC来其存储器等系统需求低得多。搭载在移动电话上的Java，就是以CLDC为基础。因此，本书中所要讲述的Configuration就是CLDC。

CLDC的JavaVM，因为是Kbyte的JavaVM，所以被称为KVM（K Virtual Machine），是在1999年的JavaOne（JavaOne Developer Conference）上所发布的。事实上，在JavaOne上与KVM同时还有Palm OS用的Java环境的KJava被发布。KJava也是在CDC上放置Palm OS专用的Profile方式来运行。该专用Profile可以对Palm OS的触摸屏、MIDI演奏、红外线通信，并让您进行控制，其概要的架构如图1-2所示。

■ CLDC系统需求

- * 可以用在Java平台上总计160KB~512KB的存储器
- * 低电力消费，用电池就可以运行
- * 对特定网络的接续性，无线、不持续的连线、带宽的限制（大部分的场合在9600bit/s以下）

下面我们整理一下CDC与CLDC之间的关系：

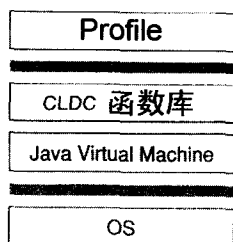
■ CDC与CLDC的关系

* CDC (Connected Device Configuration)

是以能用在个人网关、下一代智能电话、双向对讲机、PDA（个人数字助理）、家电设备、POS终端、车辆导航系统等上运行为前提设计出来的。其JavaVM，一般都称为CVM（C Virtual Machine）。不在本书的解说范围之内。

* CLDC (Connected Limited Device Configuration)

是以能使用在移动电话、PDA（个人数字助理）、家电设备、POS终端等上面为前提设计出来的。其JavaVM，被称为KVM（K Virtual Machine），是本书要说明的Configuration。



【图1-2】概要的架构

● 1-1-4 J2SE与CLDC的差异

搭载在移动电话上Java的CLDC所提供的类函数库，为J2SE核心函数库的子集合。JavaVM与KVM的J2SE的JavaVM是不一样的。在这里，我们将看看J2SE与CLDC的不同之处。