

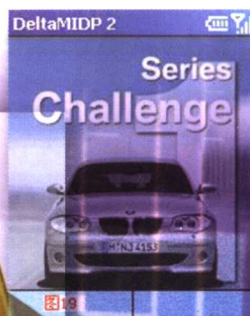


移动开发系列丛书

J2ME

无线移动游戏开发

施 铮 编著



配光盘



人民邮电出版社
POSTS & TELECOM PRESS

J2ME

无线移动游戏开发

施 铮 编著



人民邮电出版社

图书在版编目 (CIP) 数据

J2ME 无线移动游戏开发/施铮编著. —北京: 人民邮电出版社, 2006.3
ISBN 7-115-13743-9

I. J... II. 施... III. ①JAVA 语言—程序设计②移动通信—通信设备—
游戏—应用程序—程序设计 IV. ①TP312②TN929.5

中国版本图书馆 CIP 数据核字 (2006) 第 014331 号

内 容 提 要

本书全面介绍了基于 J2ME 中 MIDP 2.0 框架下移动设备游戏开发的技术、原理和方法。全书共分为 13 章, 内容包括 J2ME 概论、开发环境、游戏控制基础、交互处理、图形处理、颜色处理、图像处理、动画处理、图像层管理、字体处理、音频\视频处理、状态存储和应用程序发布。全书采取技术接口详解和示例直观化驱动方式, 结构清晰、实例丰富、易于理解, 是广大 J2ME 开发人员必备的参考书。

本书适合于具备一定 Java 语言基础的专业和准专业的 Java 开发人员阅读, 也可作为正在转向进行 Java 移动开发的各类程序员的必备 Java 参考书。

移动开发系列丛书

J2ME 无线移动游戏开发

-
- ◆ 编 著 施 铮
责任编辑 屈艳莲
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 800×1000 1/16
印张: 38.25
字数: 868 千字
印数: 1—4 000 册
 - 2006 年 3 月第 1 版
2006 年 3 月北京第 1 次印刷

ISBN 7-115-13743-9/TP · 4863

定价: 68.00 元 (附光盘)

读者服务热线: (010) 67132692 印装质量热线: (010) 67129223

Java 是现今应用最广泛的纯面向对象的编程语言，该语言易学易用、移植性强，是开发信息管理系统、Web 应用和嵌入式设备最具竞争力的程序设计语言。自从 Sun 公司于 1995 年发布了 Java 语言以来，Java 语言在功能上不断增强，应用领域也不断扩展。

本书背景

随着越来越多移动电话和个人数字助理开始融入到信息高速公路之上，从移动设备上访问 Web 站点变得越来越重要。也就是说，移动互联网时代已经到来。Java 开创了消费设备中小型的储存容量的先河，它是用于开发手机、传呼机及其他微型设备应用程序的理想语言。而 J2ME 正使这一切成为可能。

J2ME 是为了那些使用有限能源、有限网络连接（常常是无线连接）以及有限图形用户界面能力的设备而开发的一种以广泛的消费性产品为目标的高度优化的 Java 运行环境，包括寻呼机、移动电话、可视电话、数字机顶盒和汽车导航系统。J2ME 为小型设备带来了 Java 语言的跨平台功能，允许移动无线设备共享应用程序。

本书特色

目前由于 J2ME 技术自身和运行环境都在发展过程中，相关的书籍非常少，尤其像本书这样全面介绍基于 J2ME 游戏开发的书目前市场上可能没有一本，这与国内对移动设备游戏开发需求飞速发展的趋势十分不相称。本书正是为了解决 J2ME 开发人员在游戏开发时无从获得平台相关信息而编写的。本书内容翔实全面，结构清晰，示例丰富，是广大 J2ME 开发人员必备的参考书，也是热爱 Java 编程的开发者赶上主流无线设备游戏开发新技术、学习 J2ME 不可多得的一本好书。本书各章节内容的安排遵循从总体到局部，从易到难的原则，安排合理，循序渐进，叙述浅显易懂，为读者顺利掌握 J2ME 游戏开发作好准备，力求做到为读者捅透最后一层窗纸。

读者对象

本书面向的是软件开发者、内容提供商和其他希望为移动设备开发 Java 游戏、软件的专业人士。本书还面向希望将一个微缩的 Java 应用程序集成至已有信息平台的行业软件提供商。同时，本书也可供高等院校相关专业的学生和相关培训机构的学员参考。

本书目标

讲解游戏开发涉及的各个技术环节，使读者掌握游戏开发的声、色、图、动画的全面处理技能。

本书组织

根据移动游戏开发涉及的技术难易和技术之间的制约关系，全书总共分为以下 13 章。



第1章“J2ME 概论”，对 J2ME 做了整体介绍，讲解了 J2ME 的体系结构以及 J2ME 中 MIDP 应用程序 MIDlet 的概念、结构与生命周期。通过本章的学习，读者可以迅速对 J2ME 下程序的结构有个比较全面的了解，为后续几章的学习打好基础。

第2章“开发环境”介绍了 J2ME 的两个主要开发工具：WTK 和 JBuilderX。WTK 是 SUN 公司提供的无线开发工具。JBuilderX 是目前市场最为流行的 Borland 公司的集成开发可视工具。对开发工具的熟练使用是开发游戏的前提条件。

第3章“游戏控制基础”介绍了 J2ME 的线程机制。线程机制的应用是实现游戏控制的基础，游戏中对各类角色的自动控制都需要线程来实现。

第4章“交互处理”介绍了 MIDP 框架中的两种用户交互方式——高层用户界面接口和底层用户界面接口。并深入拓展研究底层事件处理机制和外设输入处理。

第5章“图形处理”介绍了 MIDP 框架的绘图机制，并分别研究绘制各类图形的方法。

第6章“颜色处理”介绍了 RGB 色彩空间的概念和彩色设备的颜色处理机制。

第7章“图像处理”介绍了 MIDP 中图像处理的机制、背景贴图的机制和静态、动态场景的处理。

第8章“动画处理”介绍了动画实现的原理，进一步深入拓展研究 MIDP1.0 和 MIDP2.0 动画处理的实现机制。并在最后学习动作、射击类游戏的核心——图像之间碰撞的检测。

第9章“图像层管理”介绍了 MIDP2.0 图像层的管理机制，学习基于图像层管理机制的层的绘制和滚屏的处理。

第10篇“字体处理”介绍了在 MIDP 中使用 Font 类来实现对字体功能的支持。开发人员可以获取字体的各种属性，通过这些属性之间的结合可以有助于显示出具有复杂表现特性的字符和文本。

第11篇“音频、视频处理”介绍了 MIDP 框架中支持移动多媒体的 Media API 接口，研究如何使用 Media API 接口实现音效的播放，最后学习视频的播放。

第12篇“状态存储”介绍了 MIDP 框架中对持久数据的管理机制，深入学习如何操作持久数据。

第13篇“应用程序发布”介绍了 MIDlet 打包机制，以及如何使用 JBuilderX 为 MIDlet 打包，最后学习使用混淆器为 MIDlet 程序加密。

本书中的源码均经过笔者上机采用 JBuilderX 调试过。

由于作者水平有限，时间紧任务重，难免存在不妥之处，敬请读者指正。本书责任编辑的电子邮箱是 quyanlian2@ptpress.com.cn。

施 铮

2006 年 2 月

本光盘是《J2ME 无线移动游戏开发》一书的配套光盘，盘中包含全书所有实例的源程序和图片、声音文件。所有程序均在 JBuilderX 环境下调试通过。

为了方便读者使用，将所有实例所在的 JBuilderX 工程文件全部提供给读者。因此，读者首先需要确认已正确安装有 JBuilderX 开发工具，然后只需要插入光盘，将光盘上工程所在的“mobile”目录全部拷贝到读者的计算机硬盘上，进入“盘符:\mobile\race”目录，将文件“MIDlet_Example.jpj”用 JBuilderX 打开，就可以使用全部实例了。

在“盘符:\mobile\race”目录下共有 5 个子目录，分别为 src、classes、bak、doc 和 jad-temp。

- src 目录下中的“ch04”~“ch12”子目录中分别存放第 4 章至第 12 章实例的源文件，这些源文件均以.java 作为后缀。
- classes 目录下存放的是经过编译生成的对应 java 源文件的可执行字节代码类文件以及游戏中使用的多媒体资源文件。类文件存放在“ch04”~“ch12”子目录中，多媒体资源文件存放在“icons”、“pic”和“res”三个子目录中。
- bak 目录由 JBuilderX 系统使用，用于存储源文件的备份文件。
- doc 目录用于存储文档文件。
- jad-temp 目录用于存储各个 MIDlet 应用程序的应用程序描述文件—jad 文件。

下面将一步一步介绍怎样使用 JBuilderX 打开、运行实例工程。具体步骤如下。

(1) 假设将实例工程文件目录拷贝至“D”盘。启动 JBuilderX 后，选择【File】→【Open Project】菜单命令，出现 Open Project 对话框，如图 1 所示。



图 1 打开工程

(2) 打开工程目录, 选中工程文件 “MIDlet_Example”。单击【OK】按钮, 整个工程文件都被装载入 JBuilderX 中, 如图 2 所示。

(3) 工程中文件的编排方式: 工程中的 “ch04” ~ “ch12” 包名分别对应正文的第 4 章~第 12 章。选择一个程序, 单击鼠标右键, 选择【Micro Run】命令, 运行对应的程序, 如图 3 所示。

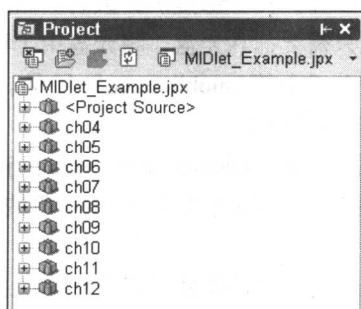


图 2 工程文件

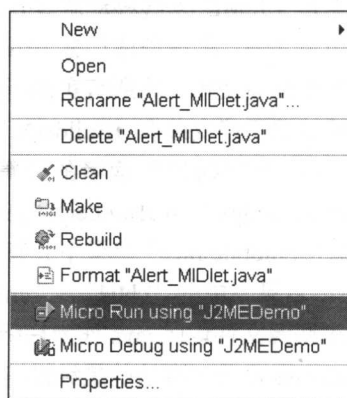


图 3 运行程序

注: 程序所附带的图片、声音文件均在工程下的 classes 目录下。

第 1 章 J2ME 概论	1
1.1 什么是 J2ME	2
1.2 J2ME 体系结构	3
1.3 MIDP 应用程序	5
第 2 章 开发环境	11
2.1 WTK 开发环境	12
2.1.1 WTK 下载和安装	12
2.1.2 WTK 环境的使用	13
2.2 JBuilderX 开发环境	15
2.2.1 JBuilderX 简介	16
2.2.2 JBuilderX 安装	16
2.2.3 JBuilderX 目录结构	18
2.2.4 配置移动开发环境	19
2.2.5 新建工程	21
2.2.6 新建应用程序	24
2.2.7 熟悉开发界面	25
2.2.8 掌握便捷工具	28
2.2.9 运行期配置	29
2.2.10 编译应用程序	31
2.2.11 运行应用程序	32
2.2.12 调试应用程序	34
第 3 章 游戏控制基础	35
3.1 Java 多线程机制	36
3.1.1 多线程与多进程	36
3.1.2 Java 的多线程	37
3.1.3 线程的状态	40
3.1.4 线程的优先级	40
3.1.5 线程的同步	41

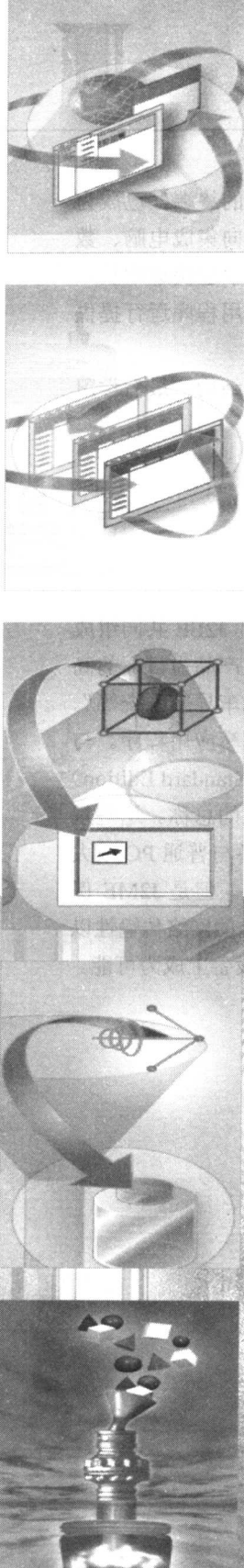


3.1.6 线程的阻塞	42
3.1.7 精灵线程	43
3.1.8 线程组	43
3.2 Timer 类和 TimerTask 类	44
第 4 章 交互处理	49
4.1 用户界面体系结构	50
4.2 高层用户界面组件	53
4.2.1 Alert 界面组件	53
4.2.2 TextBox 界面组件	57
4.2.3 List 界面组件	62
4.2.4 Form 界面组件	70
4.2.5 Gauge 界面组件	71
4.2.6 TextField 界面组件	75
4.2.7 DateField 界面组件	81
4.2.8 StringItem 界面组件	85
4.2.9 ImageItem 界面组件	88
4.2.10 ChoiceGroup 界面组件	94
4.2.11 Ticker 界面组件	100
4.2.12 Command 界面组件	102
4.3 事件处理	107
4.3.1 事件模型	108
4.3.2 事件监听器的使用	109
4.4 界面组件综合实例	119
4.5 外设输入处理	132
4.5.1 MIDP1.0 键盘处理	132
4.5.2 MIDP 2.0 键盘处理	141
第 5 章 图形处理	147
5.1 图形接口	148
5.1.1 Graphics 绘图对象	148
5.1.2 Canvas 绘图画布	149
5.1.3 坐标空间	151
5.2 基本图形绘制	151
5.2.1 绘制直线	152

5.2.2 绘制矩形	162
5.2.3 绘制弧形	173
5.2.4 绘制三角形	179
5.3 绘制风格	179
5.4 综合实例	181
第 6 章 颜色处理	191
6.1 色彩空间	192
6.1.1 基色组件构成	192
6.1.2 基色组件使用	193
6.2 颜色透明度	201
6.2.1 透明度	201
6.2.2 前景与背景的融合	210
6.3 动态调整颜色参数	217
6.4 设备颜色	222
第 7 章 图像处理	225
7.1 图像处理机制	226
7.1.1 Image 类简介	226
7.1.2 不可变图像	226
7.1.3 可变图像	230
7.1.4 图像方位变换	232
7.1.5 透明度处理	234
7.2 游戏场景处理	236
7.2.1 瓷片类 TiledLayer 简介	236
7.2.2 静态场景处理	241
7.2.3 动态场景处理	247
第 8 章 动画处理	255
8.1 动画机制	256
8.1.1 实现原理	256
8.1.2 实现方式	257
8.1.3 实例	257
8.2 MIDP 动画机制	262
8.2.1 MIDP 1.0 动画处理	262

8.2.2 MIDP 2.0 动画处理	271
8.3 碰撞控制	322
8.3.1 MIDP 1.0 碰撞处理	323
8.3.2 MIDP 2.0 碰撞处理	331
8.4 综合实例	364
第 9 章 图像层管理	377
9.1 层的概念	378
9.2 层管理机制	379
9.3 层的绘制	382
9.4 滚屏处理	384
9.4.1 基于视窗	385
9.4.2 基于场景	392
9.5 综合实例	399
第 10 章 字体处理	421
10.1 认识 Font 类	422
10.2 字体属性	423
10.2.1 字体外观	423
10.2.2 字体样式	425
10.2.3 字体尺寸	427
10.2.4 设置字体属性	428
10.3 设置字体	429
10.4 绘制字体	430
10.5 文本布局	441
10.6 自定义字体	452
第 11 章 音频、视频处理	465
11.1 认识 Media API	466
11.1.1 Media API 简介	466
11.1.2 Media API 体系结构	467
11.2 音频处理	468
11.2.1 获取音频文件	469
11.2.2 播放音频文件	474
11.2.3 控制媒体播放	476

11.2.4 实例——音频播放器	480
11.2.5 响应播放事件	483
11.3 音符处理	491
11.3.1 播放单音	491
11.3.2 播放序列音	495
11.4 视频处理	506
第 12 章 状态存储	517
12.1 RMS 概述	518
12.2 记录存储管理	520
12.2.1 打开/创建记录存储	521
12.2.2 关闭记录存储	522
12.2.3 删除记录存储	523
12.2.4 获取记录存储	524
12.2.5 获取记录存储信息	525
12.3 操作记录	527
12.3.1 添加记录	528
12.3.2 获取记录	531
12.3.3 修改记录	534
12.3.4 删除记录	535
12.3.5 遍历记录	536
12.3.6 过滤记录	544
12.3.7 比较记录	548
12.3.8 实例—积分排行榜	551
12.4 响应记录事件	580
第 13 章 发布应用程序	587
13.1 MIDlet 打包机制	588
13.1.1 清单文件	588
13.1.2 应用程序描述文件	589
13.2 打包 MIDlet	590
13.3 使用混淆器	598



Chapter 1

第 1 章 J2ME 概论

1.1 什么是J2ME

1.2 J2ME 体系结构

1.3 MIDP 应用程序

在现代信息社会的生活中，个性化的智能信息用品手机作为个人信息交互的终端，已成为人们生活、学习、工作中的必需品。如今的手机已不单单提供通话功能，而是向集成电脑、数码相机等功能的智能手机方向发展。手机用户对运行在手机上的应用程序的需求也快速增长，为满足这一需求，SUN 公司推出的 J2ME（Java 2 Micro Edition）平台为手机应用程序运行提供了一个绝佳的开发环境。

在本章中，首先介绍一下什么是 J2ME，再进一步深入学习 J2ME 的体系结构，最后学习 J2ME 中 MIDP 应用程序 MIDlet 的概念、结构与生命周期。

1.1 什么是 J2ME

J2ME 是 SUN 公司针对嵌入式消费类电子产品推出的开发平台，与 J2SE 和 J2EE 共同组成 Java 技术的 3 个重要的分支。Sun 公司将 J2ME 定义为“一种以广泛的消费性产品为目标的高度优化的 Java 运行环境，包括寻呼机、移动电话、可视电话、数字机顶盒和汽车导航系统等”。J2ME 为小型嵌入式设备带来了 Java 语言的跨平台功能，允许移动无线设备共享应用程序。与先前 SUN 公司推出的 J2EE（Java 2 Enterprise Edition）平台企业版、J2SE（Java 2 Standard Edition）平台标准版规范相比，J2ME 不是一个单独的技术规范，而是一系列技术规范的总称。这些规范定义了 Java 技术在资源限制的设备中的表现形式。由于这些小型嵌入式设备与普通 PC 或大型服务器设备相比较没有统一的硬件标准，并且外观、操作方式也是千差万别。但是 J2ME 保证了在任何时间和任何设备上运行的 Java 代码具备可移植性、部署灵活性、安全的网络传输性以及代码稳定性。使得开发人员可以让编写的程序能够运行在任意厂商或型号的设备上成为可能。

图 1-1 描述了 Java2 3 种平台支持应用程序的设备。

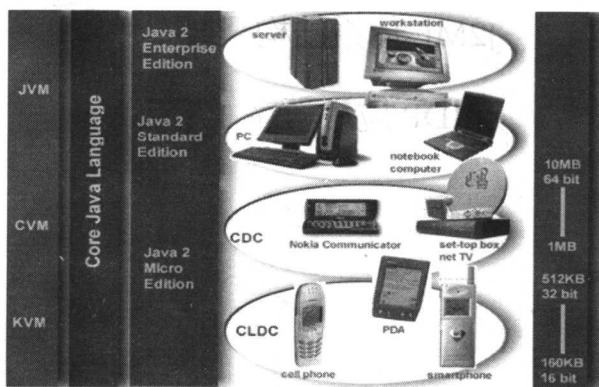


图 1-1 Java 2 三种平台比较

1.2 J2ME 体系结构

J2ME 平台是由配置 (Configuration) 和简表 (Profile) 构成的。配置将基本环境定义为一组核心类和一个运行在特定类型设备上的特定的 JVM (Java 虚拟机), 主要有 KVM 和 CVM, 配置决定了使用的 JVM。简表是针对一系列设备提供的开发包集合, 向 J2ME 配置中添加特定于域的类, 定义设备的某种作用。在 J2ME 中还有一个重要的概念是可选包 (Optional Package), 它是针对特定设备提供的类库, 比如某些设备支持的蓝牙功能。

如图 1-2 所示, J2ME 层次中 Java 虚拟机是 J2ME 技术的核心, 配置和简表用于连接虚拟机, 它们提供了用于特定 J2ME 环境的类应用程序接口。每个配置和简表处理一般或具体的消费产品, 配置和简表规范是由多种多样的设备生产商和用户共同开发并建立的。配置用于一组通用设备的最小的 Java 平台, 常常归为一种横向的设备分组。横向分组设备是那些共享相同的内存安排、通信带宽、能量需求以及用户能力的设备, 一般认为配置能够提供这众多的设备的所有需求。

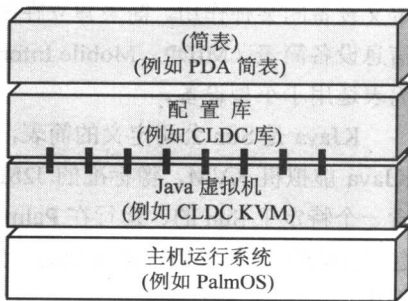


图 1-2 J2ME 体系结构

另一方面, 简表完善了配置, 为某个具体的设备家族或某个具体的工业产品应用程序提供更高的性能。换言之, 简表为具体的纵向市场的设备比如说移动电话提供更多的性能。这里的关键就是简表必须完善配置, 没有配置和虚拟机提供核心类应用程序接口和运行期环境的前提下, 简表也不可能运转。通常, 简表为一种给定的垂直分组设备提供用户界面、输入法、持久性机制。这类简表被认为是开发这些设备应用程序的完整的工具包。我们见到最多的应用程序简表的例子就是移动电话简表和个人数字助手 (PDA) 简表, 简表为范围广泛的设备提供非常特殊的功能或应用程序可移植性。

配置将基本运行环境定义为一组核心类和一个运行在特定类型设备上的特定的 JVM。将来还可能定义其他的配置, 但目前 J2ME 仅存在 CLDC 和 CDC 两种配置。

连接限制设备配置 (CLDC, Connected Limited Devices Configuration) 与 KVM 一起使用, 用于内存有限的 16bit 或 32bit 设备, 用于开发小型 J2ME 应用程序的配置。

CLDC 的硬件参数如下。

- 512 KB 以下内存。
- 有限能量供应, 通常使用电池。
- 有限或非持续网络连接。
- 简单的用户界面。



- 16bit 或者 32bit 的处理器。

连接设备配置 (CDC, Connected Devices Configuration) 与 CVM 一起使用, 用于要求内存超过 2MB 的 32bit 或 64bit 设备。互联网电视机顶盒便是一个典型的这类设备。

CDC 的硬件参数如下。

- 2MB 以上内存。
- 具有网络连接能力, 通常为无线网络。
- 需要实现 Java 虚拟机规范的全部功能。
- 32bit 或者 64bit 的处理器。

简表定义了应用程序所支持的设备类型。特别地, 它向 J2ME 配置添加了特定于域的类型来定义设备的某种作用。简表建立在配置的顶部。J2ME 中, 已经定义了两种简表: KJava 和移动信息设备简表 (MIDP, Mobile Information Device Profile), 它们也被建立在 CLDC 上。这两种简表适用于小型设备。

KJava 是 Sun 公司定义的简表, 它包含 KJava API。KJava 简表建立在 CLDC 配置的顶部。KJava 虚拟机 KVM, 像标准的 J2SE 虚拟机那样接受相同的字节代码和类文件格式。KJava 包含一个特定于 Sun 的、运行在 Palm 操作系统上的 API。这个 Kjava API 和 J2SE 抽象视窗工具包 (AWT) 有很多地方都是相同。然而, 由于它不是一个标准的 J2ME 软件包, 已逐渐从市场退出。

MIDP 简表适用于诸如蜂窝电话和寻呼机等移动设备。MIDP 和 KJava 一样, 也是建立在 CLDC 之上的, 并且提供一个标准的运行环境, 允许在终端用户设备上动态地部署新的应用程序和服务。MIDP 是一个公共的、为移动设备设计的工业标准简表, 它不依赖某个特定的商家。对于移动应用程序开发来说, 它是一个完整的、被良好支持的平台。

MIDP 简表是开发人员用来开发手机游戏的开发接口。MIDP 2.0 简表提供的开发包一共有 10 个, 分别如下。

- java.lang 包: 提供 Java 语言的基础类库。
- java.util 包: 提供一些实用处理类, 例如日期类。
- javax.microedition.io 包: 提供输入/输出流操作类以及联网类。
- javax.microedition.lcdui 包: 提供高层用户可视界面组件类以及底层响应键盘类和绘图类。
- javax.microedition.lcdui.game 包: 提供处理游戏动画、背景类以及图层管理类。
- javax.microedition.media 包: 提供音频、视频处理类。
- javax.microedition.media.control 包: 提供控制音频、视频类。
- javax.microedition.midlet 包: 提供定义 MIDP 应用程序的主类, 为 MIDP 应用程序提供访问关于其运行所在的环境信息。
- javax.microedition.pki 包: 提供控制安全连接的类。
- javax.microedition.rms 包: 提供处理数据持久机制的类。

现在 MIDP 的最新版本为 MIDP 2.0, 与 MIDP 1.0 比较最大的不同在于新增了专门用于开发游戏的 `game` 包开发接口。目前, 市场上大多数智能手机都能够支持 MIDP 1.0, 部分新出的手机支持 MIDP 2.0。

图 1-3 所示描述了不同的虚拟机、配置和简表之间的关系。它同时把 J2SE API 和它的 Java 虚拟机进行了比较。虽然 J2SE 虚拟机通常也被称为一种 JVM, 但是 J2ME 虚拟机、KVM 和 CVM 都是 JVM 的子集。KVM 和 CVM 均可被看作是一种 Java 虚拟机, 它们是 J2SE JVM 的压缩版, 并特定于 J2ME。

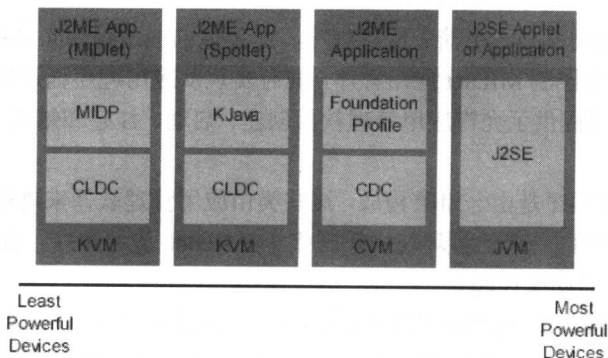


图 1-3 配置/简表比较

1.3 MIDP 应用程序

MIDP 应用程序被称为 MIDlet, MIDlet 是 MIDP 中的基本执行单元, 它不能单独运行, 必须运行在特定的运行环境中, 这个特定的运行环境被称为容器。容器可以看作是一个运行于 Java 虚拟机之上的一个大的应用程序, 容器负责运行 MIDlet, 但容器不负责完成特定应用, 因此需要开发人员在 MIDlet 中编写程序来完成具体应用。

至于 MIDlet 容器的实现机制、内部结构、外部形状对于开发者而言均不需要考虑。因为不管容器是来自于哪家厂家、何种型号的移动设备, 只要支持 J2ME 的 MIDP 简表, 符合 SUN 公司发布的 CLDC 和 MIDP 规范标准, 容器对于开发人员来说均具有一样的接口。

对于具体的 MIDP 实现来说, 负责管理 MIDlet 下载、更新、启动、暂停和删除等功能的是由设备上的应用管理软件来完成的。

为了创建一个 MIDlet 应用程序, 必须继承定义在 MIDP 中的 `javax.microedition.midlet` 包中的 MIDlet 类, 这样才能使得移动设备的 MIDlet 容器对 MIDlet 应用程序进行管理和控制。

MIDlet 类语法描述如下: