

科技新贵



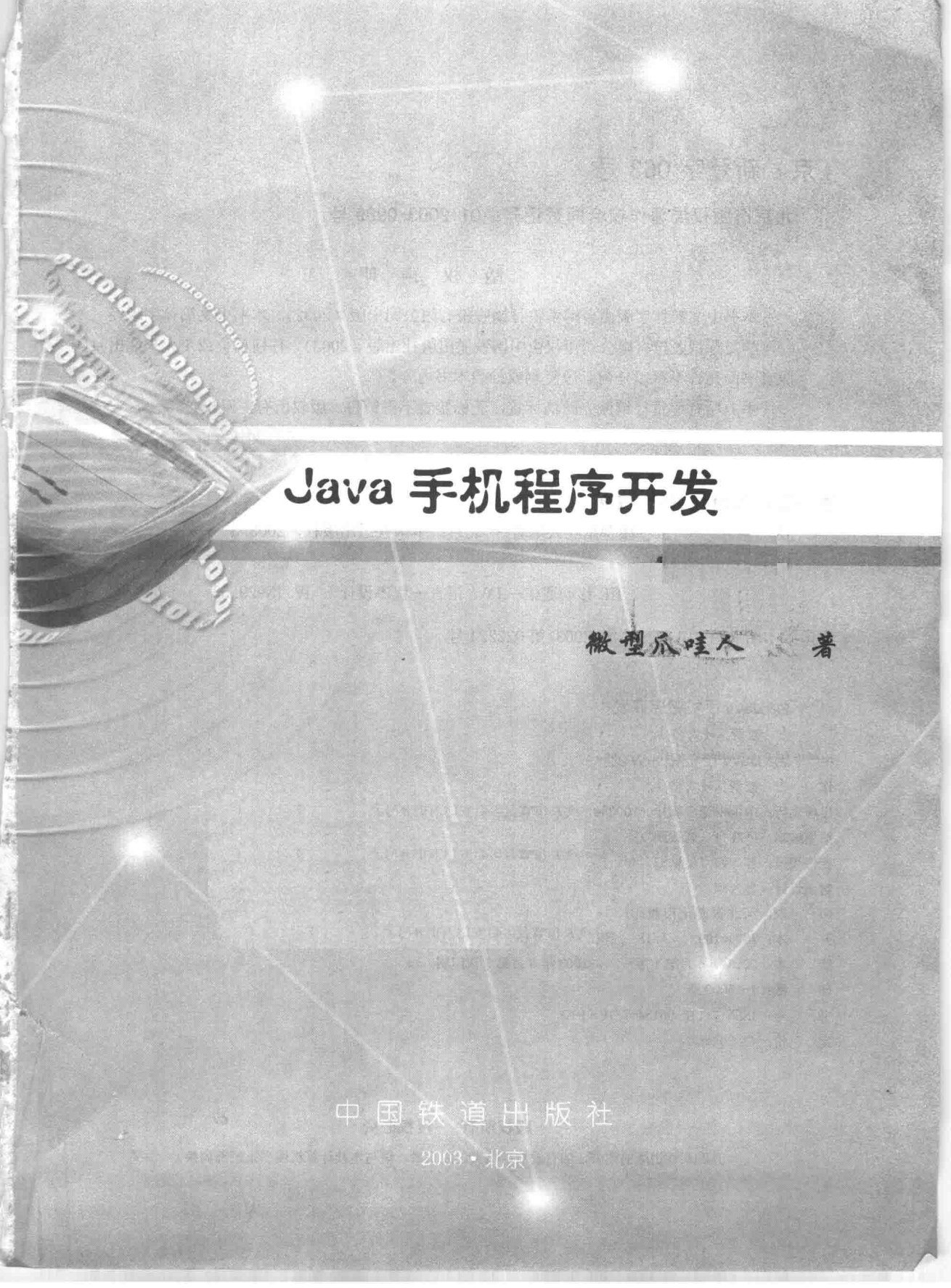
Java 手机程序开发

J2ME—CLDC/MIDP

微型爪哇人 著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



Java 手机程序开发

微型爪哇人 著

中国铁道出版社

（北京） 2003 · 北京

(京)新登字 063 号

北京市版权局著作权合同登记号: 01-2003-0989 号

版 权 声 明

本书中文繁体字版由台湾学贯行销股份有限公司出版(2002)。本书中文简体字版经台湾学贯行销股份有限公司授权由中国铁道出版社出版(2003)。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。

本书贴有学贯行销激光防伪标签,无标签者不得销售。版权所有,侵权必究。

图书在版编目(CIP)数据

Java 手机程序开发/微型爪哇人编著. —北京:中国铁道出版社, 2003. 3

ISBN 7-113-05184-7

I. J… II. 微… III. 移动通信—JAVA 语言—程序设计 IV. TN929.53

中国版本图书馆 CIP 数据核字(2003)第 022224 号

书 名: Java 手机程序开发

作 者: 微型爪哇人

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 吴秋淑

封面设计: 孙天昭

印 刷: 河北省遵化市胶印厂

开 本: 787×1092 1/18 印张: 24.25 字数: 481 千

版 本: 2003 年 5 月第 1 版 2003 年 5 月第 1 次印刷

印 数: 1~5000 册

书 号: ISBN 7-113-05184-7/TP·910

定 价: 43.00 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

在无线通讯的推波助澜下，移动通信装置即将成为个人的信息平台，包括交易、娱乐、媒体等都可以在这个平台上进行，瞬间改变了人类的生活形态，而如何快速地满足各种不同需求将是决定其价值的关键，J2ME 帮助我们找到了这些问题的答案，跨平台、安全性、动态下载与即时互动性彻底地满足了移动通信装置的需求，现在 J2ME 已经成为了手机应用开发的标准，未来所有的手机都将支持 J2ME，欢迎您一起加入手机应用程序的开发行列。

本书由台湾学贯行销股份有限公司提供版权，经中国铁道出版社计算机图书中心审选，彭木根、张卫、王键、李晓、梁明、程瑞芬、陈兰芳、崔仙翠等同志完成了本书的整稿及编排工作。

由于时间仓促，作者水平有限，错误在所难免，敬请读者批评指正，我们也会在适当的时候进行修订和补充，并发布在天勤网站：<http://www.tqbooks.net> “图书修订”栏目中。

中国铁道出版社

2003. 4

目 录

第 1 章 Java 在手机上的应用	1
1-1 Java 语言的历史	2
1-2 Java 平台的应用范围与相互关系	2
1-3 移动设备的应用开发平台——J2ME	4
1-4 Personal Java 与 Embedded Java	9
1-5 JavaPhone	10
1-6 目前 Java 在手机上的应用现况	11
1-7 使用 J2ME 平台开发手机程序的理由	14
第 2 章 CLDC / MIDP 介绍	17
2-1 CLDC 介绍	18
2-2 Java 语言的差异性	19
2-3 虚拟机的差异性	19
2-4 CLDC 定义的核心类库	20
2-5 CLDC 的专属类	22
2-6 CLDC 的安全性	23
2-7 CLDC 的系统属性	24
2-8 MIDP 介绍	24
2-9 MIDP 增加的核心类库	26
2-10 MIDP 的专属包	26
2-11 MIDP 的系统属性	26
2-12 CLDC 1.1	27
2-13 MIDP 2.0	27
2-14 K Virtual Machine	28
2-15 其他的虚拟机	28
2-16 MIDlet suite 与 JAM	29
2-17 OTA (Over The Air) 下载	30

第 3 章 开发工具与发展流程	33
3-1 准备发展工具	34
3-2 开发流程	36
3-3 MIDP 仿真器的操作	45
3-4 MIDP 仿真器的设定	46
3-5 MIDP 仿真器的其他功能	47
3-6 MIDP 仿真器的 JAM 功能	47
3-7 Wireless Toolkit	50
3-8 Wireless Toolkit 的其他设定	63
3-9 其他开发工具	65
第 4 章 应用程序模型	79
4-1 CLDC 的应用程序模型	80
4-2 MIDlet 的生命周期	81
4-3 JAM 与 MIDlet 的沟通	83
4-4 取得 MIDlet suite 内的资源	92
第 5 章 高级用户接口设计 1	97
5-1 Java 的 AWT 和 SWING	98
5-2 Display 显示管理员与 Displayable 可视组件类	99
5-3 Command 与 CommandListener	100
5-4 高级用户接口: Screen 类	106
5-5 文字跑马灯: Ticker 类	106
5-6 列表: List 类	107
5-7 文字编辑器: TextBox 类	112
5-8 信息窗口: Alert 类	120
第 6 章 高级用户接口设计 2	129
6-1 窗体: Form 类	130
6-2 Item 与 ItemListener	131
6-3 ChoiceGroup 类	134
6-4 DateField 类	138
6-5 Gauge 类	142

6-6	TextField 类	145
6-7	StringItem 类	150
6-8	Image 类与 ImageItem 类	150
第 7 章 工作调度		157
7-1	线程 (Thread)	158
7-2	Timer 与 TimerTask 类	169
第 8 章 初级用户接口设计 1		179
8-1	画板: Canvas 类与 Graphics 类	180
8-2	色彩模型	181
8-3	基本绘图方法	188
8-4	文字绘制与 Font 类	199
8-5	图片绘制与 Image 类	210
8-6	绘图辅助工具	215
8-7	事件处理机制	219
第 9 章 初级用户接口设计 2		231
9-1	仿真 Key Repeat 事件	232
9-2	图形滚动条	237
9-3	动画制作	242
9-4	地图配置	252
9-5	碰撞监测	259
9-6	Splash 画面	271
9-7	其他效果	274
第 10 章 记录系统		275
10-1	RMS 机制介绍	276
10-2	RecordStore 的管理	276
10-3	取得 RecordStore 的相关信息	277
10-4	Record 的操作	282
10-5	监测 record 的变化	287
10-6	多重字段的读取及写入	291
10-7	同步化的处理	298

Java 手机程序开发

10-8	Record 的访问	298
10-9	Record 的筛选	305
10-10	Record 的排序	311
10-11	异常处理	317
第 11 章 网络程序设计		319
11-1	Generic Network Framework	320
11-2	HTTP 通讯协议	328
11-3	Session 管理	339
11-4	Socket 网络连接	344
11-5	Datagram 网络连接	349
11-6	其他的网络连接	356
第 12 章 实用开发技巧		357
12-1	内存使用的最佳化	358
12-2	提高程序的执行效能	362
12-3	仿真浮点数运算	364
附录 A		367
附录 B		373

Chapter

1

Java 在手机上的应用



1-1 Java 语言的历史

Java 语言是由美国升阳公司 (Sun Microsystem) 的 James Gosling、Pratrick Naughton 及 Mike Sheridan 等人于 1991 年精心设计出来的计算机编程语言, 当时这个称为绿色计划的构想, 是创立一个软件架构, 使同一种编程语言所写出来的程序可以在不同的平台上运作。这种架构对于消费性电子产品而言是具有革命性的, 因为在消费性电子产品中的处理器及操作系统各不相同, 要在这些平台上开发程序必须要了解各个平台的特性, 以及其提供的 API 使用方式, 分别开发不同的版本才能执行, 影响应用程序的普及性。原来他们使用的语言为 C++, 但是基于上述需求, 他们创立了一个全新的编程语言, 这种语言称为 OAK, 也就是 Java 的前身, 后来因为名称已经被注册才改名为今日的 Java。可惜的是, 这种构想领先于当时的环境许多, 一度被升阳公司考虑结束这个计划, 好在全球信息网 (World Wide Web, 简称 WWW) 的迅速兴起, 给了 Java 另一个大放异彩的机会。

当时的 WWW 尚未如今日普及, 使用的技术也较少, 主要的呈现方式是以静态的文字及图片为主, 而 Java 的特性——平台独立、动态下载、实时互动以及安全机制非常受欢迎, 因此 Java 又成为了 WWW 上的热门编程语言。另一方面, 由于互联网的发展, 服务器端的应用也日渐受到重视, 这回大家又发现 Java 语言在服务器端应用的优异特性, 时至今日, 除了微软之外, 几乎所有的服务器端应用都以 Java 为标准。

然而最近几年, 由于微处理器性能的大幅提升, 加上无线通讯技术的突破, 手持式移动设备开始受到大众的青睐, Java 语言原先的开发目的与市场的需求不谋而合, 此时 Java 的应用层面又进一步延伸到手持式移动设备上, 并且也已经成为手机应用程序开发的标准, 国际大公司如 Nokia、Motorola 及 Siemens 等所设计的手机内都加入 Java 程序的支持, 可见在不久的将来, Java 将会成为移动设备上最热门的编程语言。

1-2 Java 平台的应用范围与相互关系

Java 平台演进到了 Java 2 之后, Java 平台分别针对不同领域的需求被分成四个版本, 亦即 J2EE、J2SE、J2ME 以及 Java Card。各个版本的意义及其应用的领域如下:

- ◆ J2EE (Java 2 Enterprise Edition): 定位在服务器端的应用。Enterprise JavaBeans 是一种用来代表企业逻辑的软件组件; JDBC 则是让程序人员能直接在 Java 内使用 SQL 的语法来访问数据库内的数据; Servlet 能够延伸服务器的功能, 通过请求-响应的模式来处理客户端的请求; JSP 是

一种可以将 Java 程序代码内嵌在网页内的技术；另外还有包括 XML、email 及 transaction 等各种应用于服务器端的 API。

- ◆ **J2SE(Java 2 Standard Edition)**: 定位在个人计算机上的应用。这个版本是 Java 平台的核心,它提供了非常丰富的 API 来开发一般个人计算机上的应用程序,包括用户接口 AWT 及 Swing,网络功能与国际化、影像处理能力以及输入输出支持等。在互联网上大放异彩的 Applet 也属于这个版本。
- ◆ **J2ME(Java 2 Micro Edition)**: 定位在消费性电子产品的应用上。这个版本针对资源有限的电子消费产品的需求精简核心类库,并提供了模块化的架构让不同类型产品能够随时增加支持的能力。这个版本的应用层面相当广泛,会是未来 Java 平台发展的重点项目。
- ◆ **Java Card**: 定位在智能卡的应用上。早期的智能卡只具备储存与简单的运算功能,Java Card 平台将精简型的 virtual machine 嵌入卡片内,使 Java 程序能通过读卡设备下载至卡片内执行,如此一来就能延伸智能卡的功能。如今 Java Card 技术已成为全球智能卡的标准,业者可以在 Java Card 上提供安全、可下载的增值业务。内含 Java Card 技术的智能卡,可以在整合的微处理器芯片上,安全存储大量个人数据与多个应用程序。目前我们即将使用的健保 IC 卡,亦使用 Java Card 平台,而美国运通银行与 Visa 组织,也已选择 Java Card 技术作为智能卡应用的开发平台。

J2EE、J2SE、J2ME 与 Java Card 这四个平台的关系如图 1-1 所示。

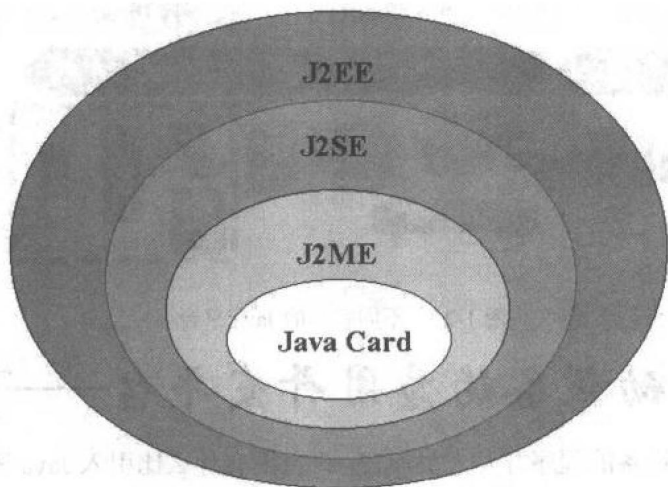


图 1-1 各种 Java 平台核心类库关系

在中间的是 J2SE, J2SE 内定义了 Java 语言标准的核心类库 (Core Class

Library), 越往外延伸表示支持的 API 数目越多, 例如最外层的 J2EE 除了完全支持此核心类库之外, 还定义了一些额外的类来开发服务器端的应用。而在 J2SE 里面的 J2ME 与 Java Card, 则只支持部分的核心类库。在这样的架构下, 这四个版本的程序是可以相互使用的, 只要在程序所使用的类与方法在各个版本所支持的核心类库范围之内即可。

对于各个不同版本的 Java 平台所设计的 virtual machine 也不相同, J2EE 与 J2SE 所使用的是完全支持的 HotSpot, J2ME 使用的是 CVM 与 KVM, 而 Java Card 所使用的则是 Card VM。

图 1-2 显示了这四个版本的 Java 平台, 而本书所要谈的手机应用程序, 则是属于红色虚线的范围:

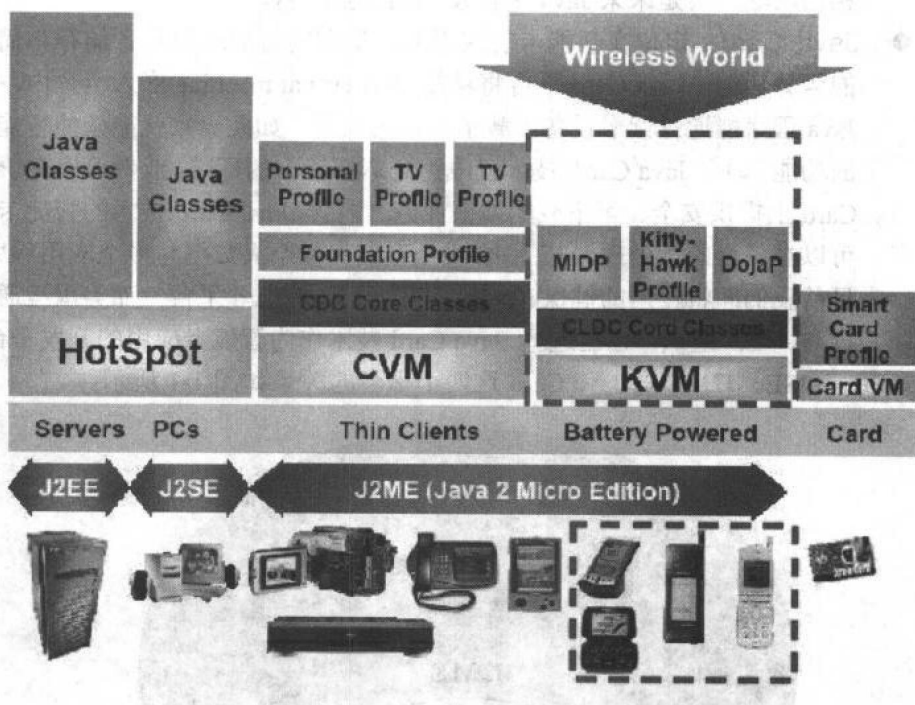


图 1-2 不同版本的 Java 平台

1-3 移动设备的应用开发平台——J2ME

对于移动设备的程序开发人员而言, 再也没有什么比引入 Java 语言更令人兴奋的事了吧。J2ME 提供了一个共通的平台, 让程序开发人员能够专注于程序的本身, 而不用花时间和各种截然不同的目标平台打交道, 真正实现“Write once, run anywhere”的理想。J2ME 平台的架构如图 1-3 所示。

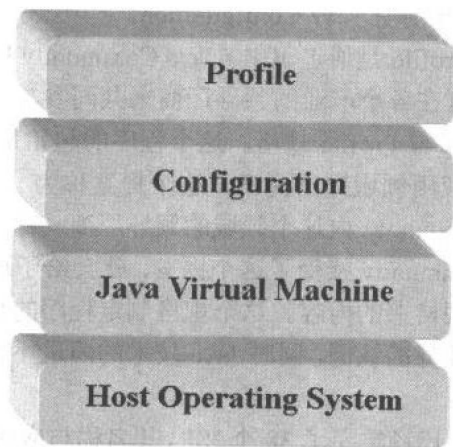


图 1-3 J2ME 的架构

在 Java virtual machine 之上, Sun 在 J2ME 中引入了两个全新的概念, 分别是 Configuration 及 Profile。由于消费性电子产品的差异性极大, Sun 认为对于不同硬件特性的电子产品, 应该有不同的设计规格, 因此 Configuration 的概念就是针对硬件的数据处理能力、存储容量、网络连接能力与用户输入输出设备来将这些电子产品区分为不同的类。目前已经定义出来的 Configuration 有两种, 分别是 CDC (Connected Device Configuration) 与 CLDC (Connected, Limited Device Configuration), 前者包含的设备对硬件资源的支持较大, 例如: 网络电话、Set-top Box、车用计算机等, 后者则可以涵盖一般的个人移动设备, 如手机及 PDA 等, 本书所著内容是针对手机的应用, 因此我们将会使用到的 Configuration 即是 CLDC。

Configuration 除了定义硬件支持程度之外, 还定义了对下面这三个特征支持的程度:

- ◆ 对 Java 语言所支持的程度
- ◆ 对 Java virtual machine 所支持的程度
- ◆ 对核心类库所支持的程度

CDC 对上述这三项的支持度较 CLDC 来得完整, 所以 CLDC 可以视为 CDC 的子集合。

另一个新的概念是 Profile。Profile 是一组延伸的 API, 不同的 Profile 针对不同类型的应用提供对应的 API, 例如对手机的应用, J2ME 定义了一个专门的 Profile-MIDP (Mobile Information Device Profile), 这个 Profile 内的 API 将在本书后面的章节中陆续介绍, 而对于 PDA 的应用开发, J2ME 也有对应的 PDAP (PDA Profile)。在 Configuration 上可以同时拥有一个以上的 Profile, 但是对于 virtual

machine 而言在其上只能存在一种 Configuration。

Configuration 及 Profile 是通过 JCP (Java Community Process) 的程序来定义的, 这个程序是由一群在各个产业的领导厂商来共同参与, 经过开会与讨论制定一个新的 Configuration 与 Profile 规格。这个程序的起始是由 Community 中的成员针对某一个重要的应用领域提出提案, 这个提案称为 JSR (Java Specification Request), 并且由 Community 内这个领域的领导厂商 (称为 Expert Group) 来评估其可行性, 一旦 Community 接受了这个提案, 就会给予这个提案一个 JSR 的编号, 并且开始制定此规格书的内容。这个规格制定程序的目的在于使所制定的规格的完整性能受到产业界的认同, 同时 Java 技术的演进不会由某一个特定公司所控制。

目前 J2ME 平台上已经定义了 38 个 JSR, 其名称与应用的范围如表 1-1 所示。

表 1-1

1: Real-time Specification for Java
提供实时系统应用程序的支持
7: Industrial Automation Extension
Java 和 Jini 技术成为工业自动化的标准
30: J2ME Connected, Limited Device Configuration
定义硬件资源有限的移动式设备的 Configuration
36: J2ME Connected Device Configuration
定义硬件资源较充足的嵌入式设备的 Configuration
37: Mobile Information Device Profile for the J2ME Platform
用于移动通讯设备上的 Profile, 目标平台为 CLDC
46: J2ME Foundation Profile
定义具有网络功能的小型设备所需要的 API, 目标平台为 CDC
50: Distributed Real-Time Specification
RMI 机制在分布式实时系统的应用
62: Personal Profile Specification
提供高度网络连接设备所使用的 API, 目标平台为 CDC

(续上表)

66: RMI Optional Package Specification Version 1.0

在 J2ME 平台上提供 RMI 机制的支持, 目标平台为 CDC

68: J2ME Platform Specification

定义 J2ME 平台的规格

75: PDA Profile for the J2ME Platform

使用于 PDA 的 Profile, 目标平台为 CLDC

80: Java USB API

定义一组可以访问 USB 接口功能的 API

82: Java APIs for Bluetooth

定义一组可以访问 Bluetooth 网络功能的 API, 目标平台为 CLDC

113: Java Speech API 2.0

定义一组可以利用手机的语音辨识功能的 API

118: Mobile Information Device Profile 2.0

MIDP 的下一代规格

120: Wireless Messaging API

提供一个访问无线网络资源的标准方式, 如 SMS、CBS、USSD

129: Personal Basis Profile Specification

为网络连接设备提供一组基本图形显示 API, 目标平台为 CLDC

134: Java Game Profile

针对游戏的开发所定义的 API

135: Mobile Media API

提供与多媒体相关的 API 让 J2ME 平台使用, 目标平台为 CLDC

138: Performance Metric Instrumentation

本规格制定标准的 API 来进行 Java 程序效能的测试

(续上表)

139: Connected Limited Device Configuration 1.1

定义 CLDC 1.1 的规格

143: JavaDesk

利用 MVC 模型提供标准的桌面应用程序 API

146: WBEM Services: JMX Provider Protocol Adapter

定义 JMX Provider Protocol Adapter for WBEM 服务的内容以及如何将 JMX 对应到 CIM

164: JAIN SIMPLE Presence

定义标准的 API, 提供可移植且安全的 SIMPLE 客户端与服务器端接口的 presence 信息

165: JAIN SIMPLE Instant Messaging

定义标准的 API, 提供可移植且安全的 SIMPLE 客户端与服务器端接口的实时信息

169: JDBC Optional Package for CDC/Foundation Profile

提供 JDBC 的 API 让 J2ME 也能使用 JDBC 的功能

171: Java Community Process (JCP) Program, version 2.5

定义新版的 JCP 程序

172: J2ME Web Services Specification

定义 J2ME 平台访问 Web 服务的内容

177: Security and Trust Services API for J2ME

定义 J2ME 平台的安全机制与信赖服务

178: Mobile Game API

定义在手机上开发游戏使用的 API, 不过这个 JSR 已经被 JCP 所停止

179: Location API for J2ME

提供关于定位服务的 API, 目标平台为 CLDC

180: SIP API for J2ME

提供与 SIP 应用相关的 API, 使 CLDC 平台能提供 SIP 服务

(续上表)

184: Mobile 3D Graphics API for J2ME

提供 3D 图形处理的 API, 目标平台为 CLDC

185: Java Technology for the Wireless Industry

制定一具有完整性的架构与 TCK 及 RI

186: JAIN Presence

定义标准的 API, 提供可移植且安全的 Presence 客户端与服务端接口的 Presence 信息

187: JAIN Instant Messaging

定义标准的 API 提供可移植且安全的 Presence 客户端与服务端接口的实时信息

190: Event Tracking API for J2ME

提供标准的 API 来追踪与记录事件

若要得到关于 JPC 及各个 JSR 的详细信息, 可以到下列的网址查找

<http://jcp.org/>。

除了 Configuration 与 Profile 之外, 为了增加平台的扩充性, J2ME 允许 third party 厂商在 Configuration 之上增加扩充的 API (extension), 提供某些特殊平台额外的支持能力, 例如 Siemens、Motorola 等手机大厂, 都有针对自家的手机特性推出扩充的 API, 这些 API 通常提供的是这只手机特有的功能, 例如控制背光或打开震动功能等, 有些还可以让您能够在 MIDlet 内直接发送 SMS。

虽然这种 API 可以让 Java 程序能够处理特定手机的功能, 但是这样也让 J2ME 应用程序的跨平台性受到影响, 如果您使用这些特别的扩充 API 来开发应用程序, 就无法在不支持这些扩充 API 的手机上执行。

1-4 Personal Java 与 Embedded Java

在 Sun 推出 J2ME 平台之前, 还曾经推行过两个针对嵌入式设备而设计的 Java 平台, 一个是 PersonalJava, 另一个是 EmbeddedJava。PersonalJava 算是一个过渡性的规格, 用来发展个人化的消费性设备, 包括完整的 JVM 以及 JDK1.1 API 的子集合, 目前已经被整合到 Personal Profile/Foundation Profile 与 CDC 里面, 不过由于使用的产品很多, 所以短时间内还会有 PersonalJava 存在。

Personal Java 与 J2ME 的关系如图 1-4 所示。