

J2ME

手机编程基础

Leopold Lee 编著



53



清华大学出版社

J2ME 手机编程基础

Leopold Lee 编著

清华大学出版社

内 容 简 介

Java 是一个让许多程序员为之激动的话题,自从其发布以来就以良好的跨平台特性让人们赞不绝口。除了在网络服务器端和浏览器的应用之外,Java 还提供了对于某些移动设备上应用程序开发的规范,如手机上的 MIDP 和 CLDC。该规范提供了 Java 与无线网络两种技术的整合,使得手机的功能进一步得到增强。本书主要讲述 J2ME 手机编程的基础知识及其应用,全书以通俗易懂的语言精典实例讲解 MIDP 和 CLDC 规范以及基于这两种规范使用 Java 进行手机通信程序开发的方法与技巧,以便使读者迅速掌握这门新兴而又蓬勃发展的技术。从而为在该手机无线通信编程领域大显身手奠定坚实的基础。

本书适合 Java 初学者或有一定基础的程序设计人员,同时也适合对无线网络或无线商务及娱乐有兴趣的读者,以及希望使用 J2ME 进行手机无线通信编程的人员。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

J2ME 手机编程基础/彼特·李编著. —北京:清华大学出版社,2003

ISBN 7-302-06781-3

I. J... II. 彼... 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 047340 号

出 版 者: 清华大学出版社

<http://www.tup.com.cn>

社 总 机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-62776969

责任编辑: 宋 韬

封面设计: 付剑飞

版式设计: 杨 磊

印 刷 者: 北京市清华园胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 185×230 印张: 13.5 字数: 280 千字

版 次: 2003 年 7 月第 1 版 2003 年 7 月第 1 次印刷

书 号: ISBN 7-302 06781-3/TP·5046

印 数: 1~4000

定 价: 21.00 元

前 言

如果说是网络正在改变着这个世界,相信大家都不会反对。网上聊天、游戏将人与人之间的距离拉得更近,网上纳税、网上购物也越来越为人们所接受,这些都得归功于电子商务和网络的迅速发展。

随着网速的不断提高和人们在网络上可做的事情的范围不断扩大,网线或电话线的局限性越来越成为人们的负担,发展无线网络的呼声越来越高。手机是人们日常所用的非常频繁的通信工具,虽然现在其费用比有线电话要高,但是其随时随地的沟通的优势,使人们摆脱了线路的束缚。目前中国的一些大中城市几乎人手一部或几部手机,正是由于手机的高普遍率,使得无线移动商务才得以迅速普及和发展。

在无线运营商方面,中国移动和中国联通都提供了手机无线通信方面的一些服务,中国移动开展了 GPRS 无线数据业务服务,经过不断增容和改进,现在一些主要城市通过 GPRS 无线上网的速度和稳定性已经与普通的拨号上网相差无几。联通更是看好这块蛋糕,CDMA 1X 业务的开展,在业界掀起了无线网络的热潮,该业务除了更清晰、更健康的话音服务之外,数据传输和因特网接入功能也更快、更稳定。

但是,只有网络的支持而没有用户手中的接入设备也是无法实现无线通信的。手机是最常用的无线设备,因此通过手机进行无线数据业务服务前景广阔,而且随着手机硬件的不断发展,其在无线移动商务方面的前途更是不可限量。移动商务主要是指消费者借助无线网络,通过手机或 PDA 等手持设备,进行有形货物或无形劳务的具有货币价值的交易,或执行数据采集的工作更好地推进商务发展等活动。人们使用手机查询银行账户、转账、数据采集、地理信息查询、收听音乐如今都已经成为现实。因为现有的 J2ME 技术已经做到了这一点。

Motorola、Nokia 等世界移动电话的提供商也在积极地进行数据业务的支持工作,其技术领域涵盖了网络设备和手持设备两端,市场上支持无线数据业务和 Java 技术的手机产品是越来越多,这一切都在告诉我们,无线网络的时代已经不远了。

本书主要讲述跨平台的 Java 技术(J2ME)在移动电话平台的应用程序开发,共有 14 章,主要介绍了 Sun 公司推出的 MIDP 和 CLDC1.0 规范的 API 的使用方法和技巧,包括了 Java 语言入门、界面设计、规范介绍、网络设计、手机存储管理、计时器应用和程序优化等方面,使读者能够具有自行编写手机应用程序的基础知识,并能够开发出符合自己要求的应用程序。

本书主要由 Motorola 公司的软件工程师 Leopold. Lee 编写,另外,下列人员也参加了素材及文字整理工作,他们是张勇、刘东、宋明仰、武勤、杜果、苏盼、马宏、顾信

禾、胡松、孙亮、王行、郭星、柳影、陈鹏、莫智兵、陆彤、赫聪、彭亭、金海、夏玲、张安、杨腾飞、丁扬、童园、卢晴等,在此一并表示感谢。

由于编写时间比较仓促,本书难免会有疏漏之处,恳请各位读者不吝指正。谢谢!

作 者
2003 年 5 月

目 录

第 1 章 初识 J2ME	1
1.1 Java 平台和 J2ME 技术概况	1
1.1.1 Java 语言的发展和现状	1
1.1.2 J2ME 概述	3
1.1.3 CLDC 和 MIDP	5
1.2 支持 J2ME 的手机设备及常用开发工具	7
1.2.1 支持 J2ME 的手机设备	7
1.2.2 支持 J2ME 的开发工具	14
1.3 本章小结	16
第 2 章 Java 编程基础	17
2.1 面向对象编程	17
2.1.1 对象和类	17
2.1.2 属性和行为	18
2.1.3 类的组织和类行为	19
2.2 Java 编程知识	21
2.2.1 语句和表达式	21
2.2.2 变量和数据类型	21
2.2.3 Java 程序的注释	23
2.2.4 Java 语言中的字面量	24
2.2.5 表达式和运算符	25
2.2.6 字符串运算	28
2.2.7 数组	28
2.2.8 多维数组	29
2.2.9 if 条件语句	30
2.2.10 switch 条件语句	30
2.2.11 while 和 do 循环	31
2.2.12 for 循环	32
2.2.13 循环退出语句	32
2.3 Java 语言中对象的处理	33
2.3.1 新对象的创建	33
2.3.2 类变量和实例变量的访问和设置	34
2.3.3 类方法的调用	35
2.4 Java 中的强制类型转换	35

2.4.1	原始类型的强制转换	35
2.4.2	对象之间的强制转换	36
2.4.3	原始数据类型和对象之间的强制转换	36
2.5	本章小结	37
第 3 章	第一个 MIDlet 程序	38
3.1	“你好, J2ME”实例	38
3.2	J2ME WTK	39
3.2.1	WTK 工具包的安装	39
3.2.2	启动 WTK 工具包	42
3.2.3	创建工程	43
3.2.4	书写源代码和编译建立工程	45
3.2.5	在模拟器中试运行	46
3.3	工程分析	47
3.4	程序打包	48
3.5	本章小结	49
第 4 章	连接受限设备配置表	50
4.1	CLDC 的由来	50
4.1.1	小而灵便的 Java 虚拟机	50
4.1.2	运行时库	51
4.2	CLDC 概览	51
4.2.1	设备要求	51
4.2.2	语言要求	51
4.2.3	KVM 虚拟机要求	52
4.2.4	CLDC 中使用的 J2SE 类	53
4.2.5	系统属性	56
4.3	CLDC 独有的类	56
4.4	本章小结	61
第 5 章	移动信息设备简表	62
5.1	软件需求	62
5.2	MIDlets 和 MIDlet Suites	63
5.3	用户接口	64
5.4	数据持久性	64
5.5	无线互连	65
5.6	其他的 J2SE 类	65
5.7	MIDlet 基础	65
5.7.1	MIDlet 的主类	66

5.7.2	激活 MIDlet	67
5.7.3	MIDlet Suites 详解	67
5.7.4	MIDlet 生存周期	69
5.8	本章小结	70
第 6 章	用户接口低层 API	71
6.1	MIDP 用户界面类库	72
6.2	显示	73
6.2.1	Displayables	73
6.2.2	Display 类	73
6.2.3	设置当前的 Displayable	74
6.3	命令	75
6.3.1	命令的定义	75
6.3.2	命令的创建	76
6.3.3	命令对象与显示对象的关联	77
6.3.4	对 Command 触发事件的响应	77
6.4	图像	78
6.5	Canvas 类	79
6.5.1	创建 Canvas	79
6.5.2	Canvas 中的按键事件	80
6.5.3	Canvas 中的指针事件	81
6.5.4	可视性通知	82
6.5.5	强制重画	82
6.5.6	事件序列化	82
6.5.7	绘制图形和文本	82
6.5.8	获得 Canvas 的大小参数	83
6.6	本章小结	85
第 7 章	MIDP 中的图形和图像设计	86
7.1	MIDP 中的图形绘制	86
7.1.1	Repaint 机制	86
7.1.2	图形绘制	86
7.1.3	坐标	87
7.1.4	图形剪裁和坐标变换	87
7.1.5	颜色	87
7.1.6	字体	88
7.2	Graphics 类相关的图形绘制	88
7.2.1	绘制前的初始化工作	88

7.2.2	线条的绘制	89
7.2.3	弧形的绘制和填充	89
7.2.4	绘制和填充矩形和圆角矩形	90
7.2.5	文本和图像的绘制	91
7.3	MIDP 对图像的支持	92
7.4	PNG 图像格式简介	92
7.4.1	PNG 的存储	92
7.4.2	PNG 的文件格式	93
7.5	MIDP 对 PNG 的支持	93
7.6	显示一幅有趣的图片	94
7.6.1	建立工程文件	94
7.6.2	编辑要显示的图像	94
7.6.3	编写 MIDlet 代码	97
7.6.4	运行新建工程	98
7.7	Image 和 ImageItem 类的常用方法	99
7.7.1	Image 类	99
7.7.2	ImageItem 类	100
7.8	使用 MIDP 对图像的支持实现动画效果	101
7.9	本章小结	107
第 8 章	高层用户接口 API	108
8.1	屏幕	108
8.1.1	屏幕类	108
8.1.2	屏幕的标题	109
8.1.3	广告条	109
8.2	提醒	110
8.2.1	Alert 类	110
8.2.2	Alert 的类型	111
8.2.3	Alert 类的超时	112
8.2.4	Alert 对象的显示	113
8.2.5	Alert 对象实例	113
8.3	列表与文本框	115
8.3.1	列表	116
8.3.2	文本框	118
8.4	表单	119
8.4.1	表单和表单的子项	119
8.4.2	子项的状态事件	120

8.4.3	选项组子项	121
8.4.4	日期域子项	121
8.4.5	滑杆子项	123
8.4.6	图像子项	123
8.4.7	字符串子项	124
8.4.8	文本域子项	124
8.5	本章小结	125
第9章	MIDP 记录管理系统	126
9.1	记录和记录集	126
9.1.1	数据的可持续性	127
9.1.2	记录	127
9.1.3	记录集	127
9.1.4	记录集使用中的体积限制	128
9.2	打开和关闭记录集	128
9.2.1	记录集类	128
9.2.2	打开记录集	128
9.2.3	记录集的关闭	130
9.2.4	记录集的相关信息	130
9.2.5	列出和删除记录集	130
9.3	记录集中数据的读写	131
9.3.1	添加新记录	131
9.3.2	记录的读取	131
9.3.3	记录中数据的写入	132
9.3.4	记录的删除	132
9.4	记录集数据的格式	133
9.4.1	将其他格式的数据转换为二进制	133
9.4.2	从二进制形式还原	133
9.4.3	几个关于数据转换的问题	134
9.5	面向记录的几个高级操作	135
9.5.1	记录过滤接口	135
9.5.2	记录比较接口	135
9.5.3	记录侦听接口	136
9.5.4	记录枚举接口	137
9.6	完整的 RMS 编程实例	138
9.7	本章小结	141

第 10 章 MIDP 的网络编程	142
10.1 HTTP 基础	143
10.1.1 网络服务器和网络客户端	143
10.1.2 HTTP 工作原理	144
10.1.3 连接重用	145
10.1.4 跨请求的状态保持	146
10.2 创建 HTTP 连接	146
10.2.1 HttpURLConnection 接口	147
10.2.2 获取一个 HttpURLConnection 对象	147
10.2.3 GET 和 HEAD 请求	148
10.2.4 POST 方法	149
10.3 回复处理	149
10.4 HTTP 连接的状态代码	151
10.5 MIDP 应用与 Servlets 交互	151
10.6 本章小结	151
第 11 章 使用网络连接请求获取图像数据	152
11.1 显示地图的 Canvas	152
11.2 MIDlet 主程序	154
11.3 编译	159
11.4 本章小结	160
第 12 章 MIDP 中的多线程	161
12.1 线程	161
12.1.1 多线程	162
12.1.2 Java 中的多线程	162
12.2 线程类	163
12.2.1 MIDP 中的 Thread 类的方法	163
12.2.2 Runnable 接口	164
12.3 实例代码	165
12.4 本章小结	173
第 13 章 MIDlet 程序中定时器的使用	174
13.1 Timer 类	174
13.2 TimerTask 类	175
13.3 java.util.Date 类	176
13.4 只运行一次的定时器	177
13.5 可重复运行的定时器	178
13.6 使用 Timer 控制的 Gauge 的实例	180

13.7 使用时钟屏幕保护的实例	183
13.8 本章小结	186
第 14 章 MIDlet 的高级主题	187
14.1 无线网络与设备安全	187
14.1.1 无线网络更加不安全	187
14.1.2 设备丢失或被窃	188
14.2 XML	188
14.2.1 XML 技术应用的必要性	188
14.2.2 XML 解析器	188
14.3 代码优化	189
14.3.1 最小化类的数量	189
14.3.2 使用内置类	189
14.3.3 对象数目最少化	190
14.3.4 无用信息收集器的辅助工作	190
14.3.5 使用混淆器	190
14.4 本章小结	191
附录 A J2ME 的网络安全措施	192
A1 常用有线网络的安全措施——SSL	192
A1.1 SSL 简介	192
A1.2 公共钥匙安全体系简介	193
A1.3 HTTPS 和 SSL	194
A2 Sun 无线应用的网络安全措施——KSSL	195
A2.1 SSL 特性	195
A2.2 SSL 实现要点	195
A2.3 使用 KSSL 编程	195
A3 JCE—Java 加密扩展	196
A3.1 安全供应者	196
A3.2 Bouncy Castle J2ME 加密包	196
A3.3 JCE 实现	197
A3.4 Bouncy Castle J2ME 加密包进行 DES 加密传输	198
A4 小结	202

第 1 章 初识 J2ME

自上个世纪末,Java 语言以其独特的魅力征服了几乎所有的程序员,尤其是在年青一代的人的眼里,学会使用 Java 语言编程就等于拥有了可以拿到更高的地位和“薪水”的证明。J2EE、EJB、设计模式等名词也是伴随着 Java 的发展,逐渐地令人们耳熟能详。也许,还有一位 Java 家族的成员不应该被忽视,那就是 J2ME(Java 2 Micro Edition即 Java 微型版)。

作为学习 J2ME 编程技术的入门篇,本章的主要内容如下:

- Java 语言的发展简史
- Java 的版本以及面向不同的划分
- CLDC 和 MIDP 规范概述
- 目前市场支持 J2ME 技术的手机设备
- 使用比较多的 J2ME 手机程序的开发软件

1.1 Java 平台和 J2ME 技术概况

Java 语言出现时间并不算长,但是却被许多人认为是软件产业发展的规范和希望,并因此而成为编程人员和 IT 产业谈论的热点。Java 的优点有很多,但其中最为人所知的应该是使用 Java 编写的程序可以“一次编写,随处运行”,说的正是 Java 语言所具有的良好跨平台特性。

1.1.1 Java 语言的发展和现状

Sun 公司的 Java 语言开发小组成立于 1991 年,其开发的目的是开拓消费类电子产品市场,例如交互式电视和烤面包箱等智能化家用电器。刚开始编写 Java 语言时,人们并没有将设计思路局限于仅仅扩充语言机制本身,而是更加注重于语言所运行的软硬件环境。也就是说,Java 语言的设计者是要建立一个系统,这个系统可以运行于一个巨大的、分布和异构的网络环境中,而且整个语言系统对于网络都是兼容的,从而可以完成各种电子设备之间的通信与协同工作。因此工程师在语言设计过程中采用了虚拟机器码(Virtual Machine Code)的运行方式。由于虚拟机的出现,中

性浏览器 HotJava 诞生了。这种浏览器不依赖于任何硬件和软件平台,是一种实时、可靠、安全而且可交互的浏览器。紧随其后,小应用程序——Applet 出现在网络世界中,它可以作为一种独立的应用程序运行在浏览器之中。

Java 语言最早应用于 Sun 公司的一个名为“Green”的项目,该项目的目的是为家用电器开发一个分布式代码系统,使用这个系统的用户就可以利用电冰箱、电视机等家用电器收发电子邮件,并且可以对这些电器进行一些比较高级而又十分必要的控制,还能与它们进行信息交流。刚开始的方案设计是计划采用 C++ 语言,但是 C++ 语言过于复杂,而且安全性较差,最后工程师们在 C++ 基础之上开发出了一种新的语言 Oak,这种语言是 Java 的前身。Oak 是一种用于网络的精巧而安全的语言, Sun 公司曾经用这种语言的技术参与一个交互式电视机项目的招标,最后却被 SGI 公司打败了。可怜的 Oak 几乎无家可归,就在即将被抛弃的时候, Mosaic 和 Netscape 浏览器的问世,启发了 Oak 项目组成员,他们编制了上文中提到的 HotJava 浏览器,得到了 Sun 公司首席执行官 Scott McNealy 的支持,Java 开始正式进军 Internet 网络。关于 Java 名字的来历也有一个趣闻,一天,几位 Java 成员组的工程师讨论给这个新的语言起个什么名字,当时他们正在咖啡馆喝着 Java(爪哇——岛名,该岛盛产一种非常不错的咖啡豆)咖啡,有个人灵机一动说就叫 Java,立即得到了所有人的认可,从此 Java 的名字被确定了下来。

Java 语言的发展过程,如图 1-1 所示, Sun 公司始终没有放弃嵌入式系统的市场,在 Java1.1 推出的时候相继推出了嵌入式 Java(Embedded Java)和个人应用 Java(Personal Java),两者不同的是,前者针对资源非常有限且无显示装置的嵌入式电子设备;后者则是为能上网并拥有显示系统的设备所量身制作的。

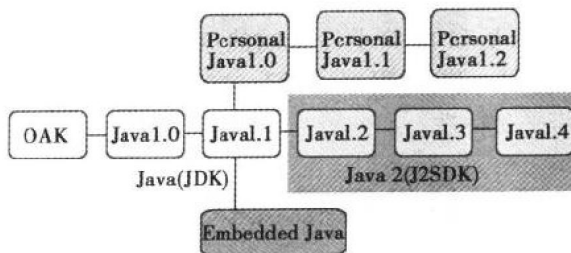


图 1-1 Java 版本发展简图

嵌入式语言虽然没能在嵌入式家电领域里受到热烈的欢迎,但因特网的大行其道却给了它无限的生机。众所周知,当众多编程语言只限于 Windows 平台的时候,Java 虚拟机的思想刚好符合了服务器软件需要跨平台运行的特性。正是网络促进了 Java 语言的极大的进步。时至今日,按照对函数库的支持程度来说,Java 家族已经是拥有 J2EE、J2SE 和 J2ME 和 Smart Card 四个各具特色成员的大家庭了。

目前,Java 的最新版本是 Java Platform。它的四个成员的作用和区别如下:

- J2EE(Java 2 Platform Enterprise Edition)

企业版 Java 主要用于开发企业服务端应用程序解决方案。

- J2SE(Java 2 Platform Standard Edition)

标准版 Java 主要用于开发一般台式计算机应用程序。

- J2ME(Java 2 Platform Micro Edition)

微型版 Java 主要用于开发信息家电应用程序。

- Java for Smart Card

Smart Card 主要用于智能芯片和 IC 卡的开发。

Java 4 种版本各自的目标市场及技术分类,如图 1-2 所示。

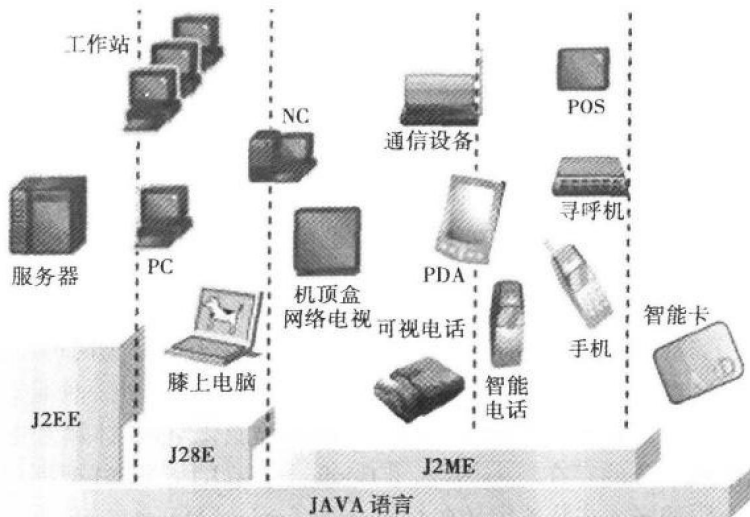


图 1-2 Java 语言分类应用

在以网络为中心的计算机时代,不支持 HTML 和像 Java 这样优秀的跨平台语言,就意味着程序应用的范围只能限于某一个或者几个固定的环境,面临被市场淘汰的危险。在未来的发展中,用 Java 开发的各种应用程序运行在不同的机器、不同的操作系统,甚至冰箱、烤箱等电子设备之中,它们通过网络连接起来,通过不同的通信协议进行交流,甚至可以进行智能性的代理(Agent)交换,实现信息交换发展的更为高级的阶段。

1.1.2 J2ME 概述

从 Java 的发展简史中可以发现一件非常有趣的事情,Java 本来是为嵌入式电子设备和智能家电开发的,却在网络应用中发展壮大起来,现在又“重操旧业”,大举进军智能和嵌入式家电并取得了出人意料的成功。是偶然或是必然?也许只能用一句“山重水复疑无路,柳暗花明又一村”来解释了。但是,就目前而言,J2ME 作为对嵌

入式设备和智能家电应用领域的最佳解决方案已经成为不争的事实。

J2ME 是建立在一个由概要描述(Profiles),配置(Configuration),Java 虚拟机(Java Virtual Machine)和操作系统(Hosting Operation System)等部分所组成的标准架构之上的,这些内容共同组成了完整的 J2ME 实时运行环境。

图 1-3 为以常见的手机平台为例的 J2ME 架构图(图形来自 Sun 网站)。

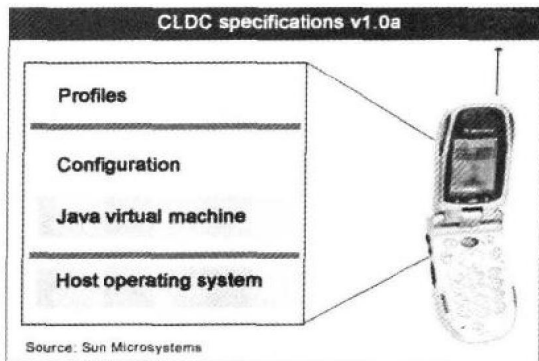


图 1-3 J2ME 架构

J2ME 遵循着“针对种类繁多而各不相同的装置,制造一个单一的开发系统是不可能也没有意义的”基本原则,先将所有的嵌入式装置大体上区分为两种:一种是运算能力有限、电力供应也有限的嵌入式装置(比如 PDA、手机等);另外一种则是运算能力相对较好、电力供应相对比较充足的嵌入式装置(比如智能化空调、冰箱和电视机机顶盒等)。由于这两种区分的存在,Java 引入了 Configuration 的概念,并把那些运算能力有限、电力供应有限的嵌入式装置定义在 Connected Limited Device Configuration(CLDC)规格之中;而另外一种装置则规范为 Connected Device Configuration(CDC)规格。

CDC 是应用在 2MB 以上内存的高端信息家电。CDC 主要是针对有限和间歇的无线连接设备。CDC 与 CVM(同样也是 Java 虚拟机的一种)和 Foundation profiles 共同定义了完全的 J2ME 环境。CVM 是为用户和嵌入设备设计的完全功能性 JVM,它支持所有的 Java 2 v1.3 VM 安全防护,JNI、JVMDI、RMI 等引用功能和相关的类库。从本质上说,CVM 支持桌面系统 JVM 具有的所有功能。CDC 由 J2SE 中最小的 Java 数据包组成。Foundation profiles 由用户设备所需要的保持类库和 API 所组成。

配置为应用程序提供了运行基础,概要描述则提供了应用程序的框架。概要描述定义了一整套 J2ME 应用程序可用的 API 类。Sun 公司推出了 2 种可用的 J2ME:基础概要描述(Foundation Profiles)和 MID(Mobile information Device)概要描述。

Sun 公司使 J2ME 体系实现了其“一次编写,随处运行”的特性,现在整个体系规范已经达到 2.0 版本,参与其中的公司和实体也越来越多,相信一定会有更广阔的前景有待被挖掘。

1.1.3 CLDC 和 MIDP

由于硬件和网络的一些限制,CLDC 与 MIDP 是目前市场支持 J2ME 的手机平台中使用最多的两个规范。

1. CLDC

CLDC 提供了一个适合于小型的,资源受限的,连接受限的设备上使用的标准 Java 平台。这些设备的典型特征是具有一个 16 位或者 32 位的微处理器和用来支持虚拟机和类库的 160KB 到 512KB 的总内存(按照 CLDC1.0 的标准),它们通常使用电池供电,并接入某种特定的网络中,联网一般基于无线,使用的带宽多小于 9600b/s 的,而且是时断时续的连接方式。CLDC 的核心基础是 KVM。“K”代表其大小是以 Kilobytes(千字节)衡量的。

CLDC 与 J2SE 相比,没有下面几个特征:

- AWT(抽象窗口开发包),Swing 或其他图形库。
- 用户定义类装载器。
- 类实例的最终化。
- 弱引用。
- RMI(远程调用接口)。
- Reflection(映射)。

CLDC 有 4 个包:java.lang、java.util、java.io 和 java.microedition。除了 microedition 包以外,其他的这几个包都是 J2SE 包的核心子集。CLDC 采用了这些 J2SE 类库,但是把其中一些在微型设备中用不到类、属性、方法去掉了。因此 CLDC 类库与 J2SE 有许多细微的差别,详情请参阅相关文档或本书后面的章节。

想要理解为什么 CLDC 去除这么多 J2SE 中重要的类和特征,请回想一下与 CLDC 相关的两条基本原理。首先,它只有 512KB 的内存空间,而像远程调用接口和映射需要的内存空间消耗实在太大了。其次,配置必须满足为一组通用设备提供最小的 Java 平台。在个人移动信息设备领域中,许多系统都不能支持 J2SE 中的众多的高级特征,例如许多低端信息家电产品不能支持浮点数,因此 Float(浮点类)和 Double(双精度类)都不得不被去掉了。再如,许多系统没有或不提供访问一个文件系统的功能或权限,因此与文件有关的类也被丢弃了。又如,错误处理是一个代价非常高的过程处理,在许多低端信息家电设备中,故障恢复是很难甚至是不可能的,所以在 CLDC 中,许多错误处理类也被删除了。

java.microedition 程序包提供了一个一般的结构来替代许多 J2SE 网络输入/输出类。CLDC 的通用连接框架(GCF,Generic Connection Framework)还定义了一个 Connector 类,允许许多不同类型的连接能够使用静态方法。使用同一个 Connector 类创建和类创