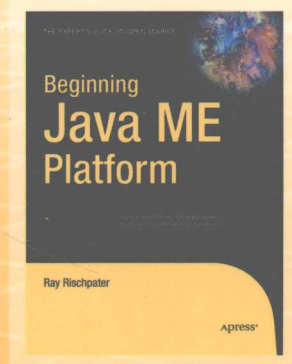


Beginning Java ME Platform

# Java ME基础教程

[美] Ray Rischpater 著  
杨越 张猛 王桂英 等译

- 世界知名移动技术专家力作
- 深入阐述最新Java移动平台技术
- 全面、实用、丰富的经典实例



人民邮电出版社  
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Java 系列

Beginning Java ME Platform

# Java ME 基础教程

[美] Ray Rischpater 著  
杨越 张猛 王桂英 等译

人民邮电出版社  
北 京

人民邮电出版社

样 书

专 用 章

## 图书在版编目 (CIP) 数据

Java ME 基础教程 / (美) 里斯查佩特 (Rischpater, R.)  
著; 杨越等译. —北京: 人民邮电出版社, 2009.9  
(图灵程序设计丛书)  
书名原文: Beginning Java ME Platform  
ISBN 978-7-115-21158-3

I. J… II. ①里…②杨… III. JAVA 语言—程序设计  
IV. TP312

中国版本图书馆CIP数据核字 (2009) 第121295号

## 内 容 提 要

Java ME (Java Platform, Micro Edition) 以往称作 J2ME, 是为机顶盒、移动电话和 PDA 之类的嵌入式消费电子设备提供的高度优化的 Java 语言平台, 包括虚拟机和一系列标准化的 Java API。本书全面介绍了构成 Java ME 的 Connected Limited Device Configuration (CLDC) 和 Connected Device Configuration (CDC), 并通过相互联系的 5 部分内容, 讲述了移动软件市场的整体动态、Java ME 平台上设计和实现的基本思想以及 Java ME 开发人员使用的最重要的 API。

本书内容全面, 实例丰富, 主要面向刚刚接触该平台的初学者, 经验丰富的开发人员也可使用它提高自己的 Java ME 开发技能。

图灵程序设计丛书

## Java ME基础教程

- ◆ 著 [美] Ray Rischpater
- 译 杨 越 张 猛 王桂英 等
- 责任编辑 傅志红
- 执行编辑 边晓娜
- ◆ 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
- ◆ 开本: 800×1000 1/16
- 印张: 26.75
- 字数: 672千字 2009年9月第1版
- 印数: 1—3 000册 2009年9月北京第1次印刷

著作权合同登记号 图字: 01-2009-2908号

ISBN 978-7-115-21158-3/TP

定价: 59.00元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

# 版 权 声 明

Original English language edition, entitled *Beginning Java ME Platform* by Ray Rischpater, published by Apress L.P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

Copyright © 2008 by Ray Rischpater. Simplified Chinese-language edition copyright © 2009 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Apress L.P. 授权人民邮电出版社独家出版。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。

版权所有，侵权必究。

## 译者序

Java ME为在移动设备和嵌入式设备（比如手机、PDA、电视机顶盒和打印机）上运行的应用程序提供了一个稳定、灵活的环境，包括灵活的用户界面、健壮的安全模型、众多内置的网络协议，同时支持各种可以动态下载的联网和离线应用程序。基于Java ME规范的应用程序具有高度的可移植性，只需编写一次，就可用于多个设备，而且可利用每个设备的本机功能。正因如此，Java ME取得了巨大的成功，它从一诞生就势如破竹，迅速成为手机游戏开发的最常用工具之一。

随着Java ME的诞生，介绍Java ME的书籍应运而生，出版一本能让初学者一窥Java ME神秘面容的书势在必行，因此我们选了这本专门面向Java ME初学者的书进行翻译，以便让更多有志使用Java ME开发程序的新手尝到将梦想变为现实的乐趣。但也正因如此，有些比较深入的问题需要读者查阅其他资料。

本书分为5个部分，共18章。第1~3章介绍Java ME的组成，所有Java ME平台通用的API以及可用的工具。第4~8章介绍MIDlet、面向Java ME开发人员的用户界面组件层次结构、应用程序用于持久存储的Java ME记录存储模型、一个可选的Java ME API和JavaMobile Game API，还将介绍使用Java ME编写独立于平台的简单游戏的方法。第9~11章介绍目前大量应用于固定消费电子产品（如机顶盒）的Java ME部分，Java ME平台上可用的其他执行模型以及某些Java ME设备使用远程方法调用与网络上的其他服务交互的方法。第12~14章分别介绍如何使用GCF，在使用XML（当前许多通信应用程序的一个重要方面）时有什么样的选择方案，以及如何访问多个Java ME设备所具有的广域无线联网功能。第15~18章介绍一些常见的可选API。

除了与Java ME相关的技术知识，本书还将介绍有关Java ME的业务信息。优秀的软件开发人员必须了解移动软件开发市场的动态，因为这些动态一直以来都影响着Java ME，而且这种影响将长期存在。

本书由杨越、张猛、范书义、王桂英等翻译，安晓梅审校。在翻译过程中得到了欧阳宇、张波、盛海艳及人民邮电出版社图灵公司编辑的热心帮助，在此一并致谢。由于译者的知识水平有限，加之时间比较仓促，书中难免会出现一些疏漏，恳请广大读者给予批评指正。



# 前言

当我开始撰写本书时，朋友和同事的言论常常让我吃惊不已。许多人问我等到这本书出版的时候，其他平台（如Android或iPhone）的发展是否会让Java ME技术过时，有些人甚至怀疑Java ME会不会消失。也有人指出，随着Java各产品线的不断聚合，对专门介绍Java ME的书的需求会萎缩。还有些人严厉地批评说，Java书籍的市场已经饱和，浪费时间再写一本书毫无意义。当你决定是否阅读本书时，可能也会提出这样的问题。你也许对将Java ME作为部署现有产品的平台感兴趣，或者只是考虑是否要学一学Java ME技能。

Java ME平台非常成功。如今，消费者使用的运行Java ME的设备已经达到了数十亿，而且还有更多的设备正加入到这个行列中来，包括手机、机顶盒和其他目前正在开发的你想象不到的设备。Java ME在市场上已经站稳了脚跟，更在JCP（Java Community Process）的推动下迅速发展演进，以应对现有的和新的竞争平台（包括Qualcomm BREW、Android和Apple iPhone）带来的各种挑战。

Java ME、Java SE（Java Platform, Standard Edition）和Java EE（Java Platform, Enterprise Edition）之间的交叉影响（cross-pollination）已得到广泛认可，而且会延续下去。JCP成员在引入可以在这些Java平台之间共享的API的时候非常谨慎，许多Java ME的API都是Java SE的API的子集。在某些情况下，则正好相反：Java ME API被引入Java SE，如用于通信和联网的Java ME框架。随着设备的功能越来越强大，各种Java产品线会不断聚合，但是移动设备特定的限制（包括随时随地的网络接入、体积小且电量不足的电池、内存和处理器资源的局限性）将在Java平台内产生一种特殊的需求。Java ME和JCP正好为供应商提供了满足这种需求的框架。

市面上已有很多关于J2ME（Java ME的前身）的优秀图书，以及部分关于Java ME的好书。但是，Java ME平台正以惊人的速度发展和演变，而Java ME又是一种规模庞大的产品，这使得初学者找到一本全面而基础的好书来学习Java ME相当困难。本书致力于全面介绍构成Java ME的两个方面，要想掌握Java ME，必须了解这两个方面。与此同时，我明确区分了哪些Java ME API是必需的，哪些是可选的。我认为在学习基础知识的过程中，需要理解一些Java ME中反复出现的基本原理，但是不必记住每个可选的Java ME类中的所有方法。毕竟每个人的精力有限，故应该首先掌握平台基本原理，以便以后专攻自己感兴趣的领域。简言之，对你来说，我未在书中介绍的某些内容可能与本书内容同样重要。

## 阅读本书的理由

我基本上已经回答了这个问题，但在此有必要重申一下。Java ME是移动计算市场中不可或

缺的部分，而且是每个使用移动设备的软件开发人员都应该熟悉的平台。无论你是需要在日常工作中使用它，或是将其视为竞争对手所用的平台，还是只想了解它与你现在开发应用程序所使用的平台有怎样的差别，了解Java ME的基本原理都能使你成为一名更出色的移动软件开发人员。

无论是移动应用程序的开发新手，还是有在其他平台上编写移动应用程序经验的人，如果你现在有兴趣了解成为Java ME开发人员需要学习哪些内容，就应该阅读本书。Java ME的配置分为CLDC（Connected Limited Device Configuration，有限连接配置）和CDC（Connected Device Configuration，连接设备配置）两种，对二者予以同等关注，将有助于在日益繁荣的手机市场或崭露头角的机顶盒和高端手机市场中，开发具有高级用户界面和其他功能的软件。因为从根本上说Java ME设备是支持网络的设备，所以我花了大量篇幅介绍Java ME中用于支持应用程序通信的API，还介绍了在未来几年Java ME可能用到的新的通信方案。阅读本书后，你将能够掌握Java ME开发人员使用的最重要的API，还将了解Java ME平台上设计和实现的基本思想，以及移动软件市场的整体动态。

然而，读者也应具备一定的基础。本书假定你以前至少接触过Java SE，包括语言和它所支持的一些主要的类。你可以不知道HashMap和TreeMap之间的区别，但至少应该了解Java语法、Java包系统以及java.lang和java.util包中的一些基本类。UML（Unified Modeling Language，统一建模语言）是一种重要的沟通工具，因此你应该对其有所了解，我经常使用UML类、状态机和序列图来帮助说明各种Java ME组件之间的关系。

如果你是移动软件开发新手，也不必担心。本书的一个主要目的是帮助你了解移动软件开发市场的动态，这些动态一直影响着Java ME，将来也必定会继续影响。我坚信一名优秀的软件开发人员不仅要了解平台，而且要了解市场背后的行业发展。我并不期待你是Java专家：仅用最少的Java特有的语言特性就能编写出可靠的代码。如果文中出现闭包或匿名内部类，我会提前说明。我写这本书的目的是让你学习编写移动应用程序，而不是成为公司的Java权威。

坦诚地讲，本书不适合于想成为Java权威的人。我没有详细讨论每个可选的Java ME API（例如，没有讨论Java Mobile 3D Graphics API和Java ME对蓝牙的支持），因为其他书籍已经对此进行了详细的介绍，而且并不是每个Java ME开发人员都需要了解这些内容。反过来说，即使你已经拥有丰富的Java ME开发经验，仍可从本书中学到一些东西，但是最好将时间花在深入研究自己感兴趣的一组特定的可选API上。其他资源可能更适合于你，比如描述特定API的JSR（Java Specification Request，Java规范请求）或关于某个具体主题的图书。本书主要面向新接触Java ME平台的初学者，向他们介绍一些基本规则和相关知识，并不深入到每个具体细节。

## 本书的使用方法

向初学者介绍Java ME极具挑战性，因为从多个角度来看，Java ME实际上都应用于两个平台：一个是在手机上大获成功，另一个则是部署于其他消费类电子产品上。作为一名工程师，我深知工程师的繁忙程度，他们希望书中只提供解决具体平台（如运行Java ME CDC的机顶盒）上的问题集所需的信息。因此，我将本书分成了5部分，你可以从中选择最需要的信息。

- 第一部分 介绍所有Java ME开发人员都应了解的信息：Java ME是如何组织的，哪些API是所有Java ME平台通用的，以及有哪些可用的工具。强烈建议通读本部分所包含的3章

内容，以了解Java ME市场和发展趋势。

- 第二部分 详细介绍Java ME CLDC和移动信息设备配置文件（Mobile Information Device Profile, MIDP）。这种配置和相关的配置文件构成了世界上部署最为广泛的移动应用程序平台，如果你对编写手机或其他无线终端应用程序感兴趣，就需要牢固地掌握这些内容。
- 第三部分 探索Java ME CDC，它支持当今许多消费类电子设备，包括电视机顶盒和某些高端手机。CDC甚至在蓝光光盘（Blu-ray Disc）标准中占有一席之地。此部分中的信息在其他介绍Java ME的资料中往往被忽略，但CDC在Java ME开发中的作用越来越重要，有必要探讨一下。
- 第四部分 介绍Java ME如何支持应用程序与Web上的其他应用程序通信。你将学习GCF（Generic Connection Framework，通用连接框架）（它是Java世界中的一个重要成员）以及Java ME如何支持你使用Internet协议和无线消息传送协议。
- 第五部分 介绍一些每个Java ME开发人员应知道的可选API。了解这些接口很重要，因为它们提供几乎每个应用程序都要具有的功能（如安全和信任接口），且这些接口提供其他可选Java API扩展（如Mobile Media API）的基本框架。此部分的最后一章研究可选API如何将Java ME分割成独立部分，以及Java社区如何协作以解决这一部分。

每部分前面有一段简短的篇首语，帮助你了解书中内容。这5部分共由18章组成。

- 第1章介绍Java ME开发人员必须理解的一些关键词汇和业务概念。
- 第2章介绍Java技术在Java SE和Java ME之间经历的主要变化。经验丰富的Java SE开发人员应该仔细阅读本章，因为它讲述了以前熟悉的哪些语言特性和类在Java ME中同样适用。Java初学者应该跳过本章，需要时再返回阅读。
- 第3章的内容让你能够使用开发Java ME的重要SDK（Software Development Kit，软件开发工具包）构建自己的第一个Java ME应用程序。你将了解为什么选择NetBeans作为开发Java ME应用程序的环境，以及如何使用NetBeans从头开始构建两个简单的应用程序。后面章节中的许多示例都以这两个示例应用程序为基础。即使你以后决定换用其他SDK，本章也能帮助你了解Java ME开发工具是如何集成在一起的。在构建这两个简单应用程序的过程中，你将看到Java ME的主要功能。
- 第4章开始探索软件世界中最成功的一个应用程序平台——MIDlet，它是大多数Java ME设备上的应用程序执行单元。
- 第5章介绍仅对Java ME开发人员可用的用户界面组件的层次结构。你将了解Java ME所提供组件的工作方式和交互方式，以及扩展Java ME组件层次结构的方式。
- 第6章介绍应用程序可用于持久存储的Java ME记录存储模型。即便在没有传统文件系统的设备上也可使用记录存储，它提供了一种以可搜索、持久的方式存储相似数据记录的功能。
- 第7章首次揭开了一个可选Java ME API的神秘面纱，这个API不是在所有平台上都可用的。它是一种非常重要的API，应该尽早掌握。你需要理解该API与前一章中介绍的记录存储模型的工作方式。
- 第8章介绍Java移动游戏API以及如何使用Java ME编写平台无关的简单游戏。游戏开发是



一个复杂的主题，本章并未纠结于不引人关注的游戏开发细节，而是强调了Java ME的基础知识，因为它们与游戏开发密切相关。

- 第9章介绍迄今为止大量应用于固定消费类电子产品（如机顶盒）的Java ME部分。你将学习这些设备支持的应用程序模型，以及它们提供的接口。
- 第10章介绍Java ME平台上可用的其他执行模型，此外还介绍了对遗留Java applet和某些Java ME设备上可用的Swing修改版的支持。
- 第11章介绍某些Java ME设备如何使用RMI（Remote Method Invocation，远程方法调用）与网络上Java提供的其他服务交互。
- 第12章开始接触所有Java ME设备都支持的通信框架，介绍如何使用此框架和Internet协议通过网络访问数据和服务。
- 第13章在第12章基础上，介绍Java ME的可选API和开源软件包如何使应用程序能够使用XML（Extensible Markup Language，可扩展标记语言）和HTTP访问Web服务。
- 第14章介绍如何使用很多Java ME设备上可用的无线消息传送接口。利用这些接口，可以通过SMS（Short Message Service，短消息服务）之类的协议发送和接收消息。
- 第15章介绍提供扩展（如密码系统和访问智能卡）的可选Java ME接口，以及支持移动商务的接口，如用于读取RFID（Radio-Frequency Identification，射频识别）卡和条形码的可选API。
- 第16章介绍Java ME用于呈现多媒体内容的方法。介绍Java ME设备可提供的移动媒体API，以及用于静态和动态显示SVG（Scalable Vector Graphics，可缩放矢量图形）图像的可选API。
- 第17章介绍应用程序用于确定设备位置的可选Java ME接口。
- 第18章讨论了在应用程序开发人员确定能够为应用程序提供必要功能的充足设备时，Java ME设备提供的众多可选API会带来怎样的挑战。本章还解释了Java社区如何通过其他设备配置文件来应对这种挑战，如无线行业（Wireless Industry）和MSA（Mobile Service Architecture，移动服务架构）的Java技术。
- 附录提供了一份表格，其中包括有价值的移动技术和定义这些技术支持的JSR。读完本书后，如果希望进一步学习某项技术，并了解该技术如何与Java ME交互，可以使用该表格来确定学习的起点。

阅读本书的理想方式是先通读第一部分到第四部分，然后根据兴趣阅读第五部分，Java ME初学者尤其适合这种方式。也可采用其他方式阅读本书。例如，如果对具体的Java ME配置感兴趣，可先读第一部分，然后读第二部分或第三部分，接着读第四部分和第五部分。不管怎样，因为有些内容要求你掌握一些基础知识，所以在深入学习本书后面的某些内容之前，需要先阅读前面的部分。

## 如何开始学习

Apress网站（<http://www.apress.com>）上提供了本书中所有示例应用程序的源文件。请首先阅读第1章和第2章，然后从<http://www.netbeans.org>下载NetBeans SDK。如果着急的话，也可立即下

载SDK，然后通读第3章，这样可以尽快熟悉Java ME应用程序开发的全部过程。

建议在学习本书内容的基础上查阅其他资源。一个非常优秀的资源站点是Java Community Process网站(<http://www.jcp.org>)，从中可找到介绍Java ME平台(和其他Java平台及Java平台扩展)的JSR。如果你对前沿技术感兴趣，另一个优秀的资源站点是NetBeans的wiki(<http://wiki.netbeans.org>)，它非常适合于喜欢NetBeans环境的用户。最后，我的个人网站(<http://www.lothlorien.com>)上也有一些有用的资源。

## 致谢

任何一本书的出版都是很多人共同努力的结果，这样一本技术书更是如此。在本书的出版过程中，我得到了很多人的帮助，也许有些人根本没有意识到他们给了我多大的帮助，还有一些人我甚至叫不上名字。

自我的写作生涯之初，我的儿子Jarod就一直陪伴在我身边，我的第一本书几乎是和他同时诞生的。现在他已经长大了，开始写自己的散文和计算机程序，这让我们有机会分享共同学习的快乐，这种机会是多么地宝贵。他常常会问我正在做什么、怎么做，从中体现出他对写作这门艺术的敬意，这对我来说是弥足珍贵的。与Jarod在一起，我常常 would 忘却大型项目带来的那种挫折感，而这只是他带给我的许多无价的礼物之一。

我的妻子Meg从一开始就支持我完成这个项目，并一直给我鼓励，虽然她知道这意味着我将有很长一段时间不能陪她，因为我要研究和撰写本书的示例和文字。我经常不在她身边，有时即使人在，嘴里却还在不停地嘀咕着移动应用开发的一些细节，对于这一切她十分宽容。我非常珍惜我们之间的感情，它使我开始反思如何在相互支持的过程中不断成长，从而迈向成功。

Apress出版社的全体工作人员成就了本书。Steve Anglin和Richard Dal Porto从项目启动开始直至项目成功完成，一直发挥着关键的作用。Richard帮助协调本书出版过程中的各项工作，即使我偶尔会拖期，他也能使工作平稳进行。文字编辑Nicole Abramowitz对待工作耐心细致，在她的把关下本书质量极大提高。生产编辑Katie Stence使本书的审读过程很愉快，并让我第一次看到图书的实际付印效果。还要感谢那些我从未谋面的Apress成员，因为没有他们的奉献，Apress出版社不会成为这样易于合作的成功企业。

本书的技术编辑Chris King是我的同事兼好友。他对细节的关注常常超出了程序语法错误的范围，还帮我改进了书中对许多概念的阐述。他耐心运行(和阅读)本书中的每一个示例，甚至还帮我改进了许多示例中使用的伪代码。我一直都很喜欢和Chris在专业方面进行交流，本书的出版更是让我对他的专业知识佩服不已。

我还要感谢我在Rocket Mobile(现在属于Buongiorno集团)的同事们，不仅因为他们带来额外的Java ME开发体验，为本书示例提供素材，更因为他们的耐心和支持。我要向Erik Browne、Levon Dolbakian、Graham Darcy、Jonathan Jackson和Rajiv Ramanasankaran表示我的歉意，因为我在白天工作时间编写C代码，而晚上写书时编写Java代码，他们要忍受我长期在C和Java之间频繁地切换。包括Young Yoon、Scott Sumner、Jim Alisago和Wayne Yurtin在内的管理层给了我特权，使我将软件开发和写作结合起来，并提供了一种在这两个方面都可取得成功的环境。谢谢他们！

我身边的很多人都对我提供了支持，无论他们是否意识到了这种支持。**Brad Holden**、**Connie Rockosi**、**Chris Haseman**和**Shane Conder**给我的支持最多，他们为我提供了写作所必需的空间和时间。由于我拥有全职工作，在技术生涯外还有多彩的生活，所以很多时候我并不确信自己是否还有时间能够写书，每当此时，他们总是积极地鼓励我。在此由衷地感谢他们所做的这一切。

# 目 录

## 第一部分 入 门

|                                   |    |
|-----------------------------------|----|
| 第 1 章 开始 Java 之旅.....             | 2  |
| 1.1 Java ME市场简介.....              | 2  |
| 1.1.1 从设备制造商角度来看.....             | 2  |
| 1.1.2 从运营商角度来看.....               | 3  |
| 1.1.3 从消费者角度来看.....               | 3  |
| 1.2 Java ME平台.....                | 4  |
| 1.2.1 对Java移动版本的需求.....           | 4  |
| 1.2.2 移动设备上的Java.....             | 5  |
| 1.3 配置.....                       | 7  |
| 1.3.1 CLDC简介.....                 | 7  |
| 1.3.2 CDC简介.....                  | 8  |
| 1.4 配置文件.....                     | 9  |
| 1.4.1 MIDP简介.....                 | 9  |
| 1.4.2 基础配置文件简介.....               | 9  |
| 1.4.3 个人基本配置文件简介.....             | 10 |
| 1.4.4 个人配置文件简介.....               | 10 |
| 1.5 包.....                        | 11 |
| 1.6 规划Java ME开发方法.....            | 11 |
| 1.6.1 选择适当的设备目标.....              | 11 |
| 1.6.2 销售应用程序.....                 | 12 |
| 1.7 小结.....                       | 13 |
| 第 2 章 精简 Java.....                | 14 |
| 2.1 精简Java: CLDC.....             | 14 |
| 2.1.1 理解当前版本: CLDC 1.1.....       | 15 |
| 2.1.2 回顾CLDC 1.0.....             | 16 |
| 2.1.3 展望CLDC的未来.....              | 16 |
| 2.2 精简Java: CDC.....              | 16 |
| 2.3 更改Java类库以适合CLDC.....          | 17 |
| 2.3.1 对java.lang包的更改.....         | 18 |
| 2.3.2 对java.util包的更改.....         | 20 |
| 2.3.3 对java.io包的更改.....           | 21 |
| 2.3.4 在CLDC中引入类.....              | 21 |
| 2.4 更改Java类库以适合CDC.....           | 22 |
| 2.5 小结.....                       | 22 |
| 第 3 章 NetBeans IDE 入门.....        | 24 |
| 3.1 选择NetBeans IDE.....           | 24 |
| 3.2 开始使用NetBeans IDE.....         | 25 |
| 3.3 创建第一个CLDC/MIDP应用程序.....       | 26 |
| 3.3.1 创建WeatherWidget.....        | 27 |
| 3.3.2 构建CLDC/MIDP应用程序.....        | 39 |
| 3.3.3 打包和执行CLDC/MIDP<br>应用程序..... | 40 |
| 3.4 创建第一个CDC应用程序.....             | 42 |
| 3.4.1 创建WeatherApplet.....        | 43 |
| 3.4.2 打包和执行CDC应用程序.....           | 58 |
| 3.5 小结.....                       | 60 |
| 第二部分 用 MIDP 进行 CLDC 开发            |    |
| 第 4 章 MIDlet 介绍.....              | 63 |
| 4.1 最简单的MIDlet.....               | 63 |
| 4.2 理解MIDlet的生命周期.....            | 64 |
| 4.3 将MIDlet打包.....                | 66 |
| 4.4 获得属性和资源.....                  | 67 |
| 4.5 管理启动事件和报警.....                | 67 |
| 4.6 小结.....                       | 73 |
| 第 5 章 构建用户界面.....                 | 74 |
| 5.1 理解Display和可视项对象之间<br>的关系..... | 74 |
| 5.2 使用命令控制应用程序流程.....             | 77 |

|                                |     |                                  |     |
|--------------------------------|-----|----------------------------------|-----|
| 5.3 基本可视项简介 .....              | 80  | 7.2.7 枚举目录的内容 .....              | 126 |
| 5.3.1 项简介 .....                | 81  | 7.2.8 监听文件系统变化 .....             | 127 |
| 5.3.2 管理选择项 .....              | 84  | 7.3 FCOP实践 .....                 | 128 |
| 5.4 Screen类及其子类简介 .....        | 85  | 7.4 PIM包简介 .....                 | 132 |
| 5.4.1 使用Form类搜集可视项 .....       | 86  | 7.5 使用PIM包 .....                 | 133 |
| 5.4.2 警告用户 .....               | 87  | 7.5.1 确保PIM包可用 .....             | 133 |
| 5.4.3 接受大量文本 .....             | 88  | 7.5.2 打开PIM数据库 .....             | 134 |
| 5.4.4 显示选项列表 .....             | 89  | 7.5.3 从PIM数据库读取记录 .....          | 134 |
| 5.5 使用Canvas和定制Item .....      | 90  | 7.5.4 从PIM记录中读取字段 .....          | 134 |
| 5.5.1 用定制Canvas控制绘图行为 .....    | 91  | 7.5.5 修改PIM记录 .....              | 138 |
| 5.5.2 创建供屏幕使用的定制Item .....     | 92  | 7.5.6 添加PIM记录 .....              | 139 |
| 5.5.3 实现定制项 .....              | 94  | 7.5.7 移除PIM项目 .....              | 139 |
| 5.6 小结 .....                   | 98  | 7.5.8 管理PIM数据库分类 .....           | 139 |
| 第6章 使用记录存储存储数据 .....           | 99  | 7.6 PIM包实践 .....                 | 140 |
| 6.1 记录存储的技术细节 .....            | 99  | 7.7 理解代码签名和验证的作用 .....           | 145 |
| 6.2 使用记录存储 .....               | 101 | 7.8 小结 .....                     | 145 |
| 6.2.1 打开和关闭记录存储 .....          | 101 | 第8章 使用Java 移动游戏 API .....        | 146 |
| 6.2.2 移除记录存储 .....             | 102 | 8.1 了解移动游戏API .....              | 146 |
| 6.2.3 获得有关记录存储的信息 .....        | 102 | 8.2 管理事件和绘图 .....                | 147 |
| 6.3 访问记录存储中的记录 .....           | 103 | 8.2.1 轮询击键 .....                 | 148 |
| 6.3.1 添加记录 .....               | 105 | 8.2.2 管理游戏的执行 .....              | 149 |
| 6.3.2 获取记录 .....               | 105 | 8.2.3 将GameCanvas绑定到MIDlet ..... | 150 |
| 6.3.3 枚举记录 .....               | 106 | 8.3 可视元素分层 .....                 | 151 |
| 6.3.4 更新记录 .....               | 107 | 8.3.1 管理层 .....                  | 152 |
| 6.3.5 移除记录 .....               | 107 | 8.3.2 使用贴图优化可视的层 .....           | 153 |
| 6.3.6 计算记录个数 .....             | 107 | 8.3.3 制作动画 .....                 | 155 |
| 6.3.7 监听记录存储的变化 .....          | 108 | 8.4 移动游戏API实践 .....              | 156 |
| 6.3.8 理解平台对记录存储的限制 .....       | 108 | 8.4.1 实现游戏MIDlet .....           | 157 |
| 6.4 记录存储应用实例 .....             | 108 | 8.4.2 实现游戏Canvas .....           | 159 |
| 6.5 小结 .....                   | 121 | 8.5 小结 .....                     | 165 |
| 第7章 访问文件和其他数据 .....            | 122 | 第三部分 CDC 开发                      |     |
| 7.1 FCOP简介 .....               | 122 | 第9章 Xlet 和个人基本配置文件简介 .....       | 169 |
| 7.2 使用FCOP .....               | 124 | 9.1 了解Xlet .....                 | 169 |
| 7.2.1 判断FCOP是否存在 .....         | 124 | 9.1.1 查看Xlet生命周期 .....           | 169 |
| 7.2.2 获得FileConnection实例 ..... | 124 | 9.1.2 扩展Xlet接口 .....             | 170 |
| 7.2.3 新建文件或目录 .....            | 125 | 9.1.3 使用Xlet上下文 .....            | 171 |
| 7.2.4 打开文件 .....               | 125 | 9.2 编写简单的Xlet .....              | 172 |
| 7.2.5 修改文件属性 .....             | 126 | 9.2.1 查看简单的Xlet .....            | 172 |
| 7.2.6 删除文件或目录 .....            | 126 |                                  |     |

|                                    |            |                                        |            |
|------------------------------------|------------|----------------------------------------|------------|
| 9.2.2 了解Xlet依赖性 .....              | 174        | 11.4 小结 .....                          | 214        |
| 9.3 使用PBP开发轻量级用户界面 .....           | 177        | <b>第四部分 外部通信</b>                       |            |
| 9.3.1 为窗口工具包实现自己的组件 .....          | 177        | <b>第12章 访问网络上的远程数据 .....</b>           |            |
| 9.3.2 编写简单的轻量级组件 .....             | 178        | 12.1 GCF简介 .....                       | 219        |
| 9.3.3 了解PBP的窗口工具包限制 .....          | 182        | 12.2 使用套接字和数据报通信 .....                 | 224        |
| 9.4 获取Xlet属性和资源 .....              | 182        | 12.2.1 在GCF中使用套接字 .....                | 224        |
| 9.5 与其他Xlet通信 .....                | 183        | 12.2.2 在GCF中使用数据报 .....                | 227        |
| 9.5.1 实现共享对象 .....                 | 184        | 12.3 使用HTTP进行通信 .....                  | 228        |
| 9.5.2 共享对象让其他Xlet查找 .....          | 186        | 12.3.1 HTTP回顾 .....                    | 229        |
| 9.5.3 使用共享对象 .....                 | 188        | 12.3.2 在GCF中使用HTTP .....               | 231        |
| 9.6 小结 .....                       | 190        | 12.3.3 使用HTTP .....                    | 235        |
| <b>第10章 Applet 和高级图形用户界面 .....</b> | <b>191</b> | 12.3.4 利用HTTPS确保HTTP事务<br>的安全 .....    | 244        |
| 10.1 为Java ME编写applet .....        | 191        | 12.4 为网络连接授权 .....                     | 245        |
| 10.1.1 applet生命周期 .....            | 191        | 12.5 小结 .....                          | 246        |
| 10.1.2 展示applet用户界面 .....          | 193        | <b>第13章 访问Web服务 .....</b>              | <b>248</b> |
| 10.1.3 访问applet上下文 .....           | 194        | 13.1 从客户机的视角查看Web服务 .....              | 248        |
| 10.1.4 在applet之间通信 .....           | 194        | 13.1.1 了解架构 .....                      | 249        |
| 10.2 使用AWT开发用户界面 .....             | 196        | 13.1.2 通过网络交换数据 .....                  | 250        |
| 10.2.1 使用AWT容器 .....               | 197        | 13.1.3 使用XML表示数据 .....                 | 252        |
| 10.2.2 使用AWT组件 .....               | 198        | 13.2 Java ME对Web服务中XML的支持 .....        | 256        |
| 10.2.3 处理AWT事件 .....               | 199        | 13.2.1 在Java ME应用程序中<br>生成XML .....    | 256        |
| 10.3 使用AGUI开发用户界面 .....            | 200        | 13.2.2 J2ME Web服务规范 .....              | 268        |
| 10.3.1 了解顶层窗口限制 .....              | 202        | 13.2.3 kXML解析器 .....                   | 276        |
| 10.3.2 使用AGUI的附加输入支持 .....         | 202        | 13.3 小结 .....                          | 282        |
| 10.3.3 了解对绘制算法的更改 .....            | 203        | <b>第14章 使用无线消息传送API传递<br/>消息 .....</b> | <b>284</b> |
| 10.4 小结 .....                      | 203        | 14.1 无线消息传送服务 .....                    | 284        |
| <b>第11章 使用远程方法调用 .....</b>         | <b>205</b> | 14.1.1 短消息服务 .....                     | 284        |
| 11.1 Java RMI .....                | 205        | 14.1.2 多媒体消息服务 .....                   | 285        |
| 11.1.1 Java RMI架构 .....            | 205        | 14.1.3 小区广播服务 .....                    | 285        |
| 11.1.2 Java RMI接口 .....            | 208        | 14.2 无线消息传送API .....                   | 286        |
| 11.2 Java RMI OP .....             | 208        | 14.2.1 创建消息 .....                      | 288        |
| 11.2.1 查看Java RMI OP的要求 .....      | 208        | 14.2.2 发送消息 .....                      | 289        |
| 11.2.2 查看Java RMI OP提供的内容 .....    | 209        | 14.2.3 接收消息 .....                      | 293        |
| 11.3 应用Java RMI .....              | 210        | 14.2.4 管理消息报头 .....                    | 293        |
| 11.3.1 为服务编写Java接口 .....           | 211        | 14.2.5 使用WMA所需的权限 .....                | 294        |
| 11.3.2 使用Java SE实现服务 .....         | 211        |                                        |            |
| 11.3.3 为Java SE生成存根类 .....         | 213        |                                        |            |
| 11.3.4 编写远程服务主机应用程序 .....          | 213        |                                        |            |
| 11.3.5 从客户机调用远程对象 .....            | 214        |                                        |            |



|                                            |            |                                                              |            |
|--------------------------------------------|------------|--------------------------------------------------------------|------------|
| 14.3 使用推注册 .....                           | 294        | 16.2.1 了解基本的SVG概念 .....                                      | 358        |
| 14.3.1 动态注册传入消息 .....                      | 296        | 16.2.2 了解SVGAPI的组织结构 .....                                   | 359        |
| 14.3.2 使用PushRegistry API .....            | 297        | 16.2.3 呈现SVG图像 .....                                         | 361        |
| 14.4 应用无线消息传送API .....                     | 297        | 16.2.4 修改SVG图像 .....                                         | 366        |
| 14.4.1 发送和接收SMS消息 .....                    | 297        | 16.2.5 结合使用NetBeans和SVG<br>图像 .....                          | 369        |
| 14.4.2 发送和接收MMS消息 .....                    | 303        | 16.3 应用MMAPI和SVGAPI .....                                    | 369        |
| 14.5 小结 .....                              | 311        | 16.3.1 播放音频和视频 .....                                         | 377        |
| <b>第五部分 其他 Java ME 接口</b>                  |            | 16.3.2 捕获图像 .....                                            | 378        |
| <b>第 15 章 保护 Java ME 应用程序 .....</b>        | <b>315</b> | 16.3.3 播放SVG内容 .....                                         | 379        |
| 15.1 理解安全需求 .....                          | 315        | 16.4 小结 .....                                                | 380        |
| 15.2 Java ME的安全与信任服务 .....                 | 317        | <b>第 17 章 定位 .....</b>                                       | <b>382</b> |
| 15.2.1 使用APDU API与加密硬件<br>通信 .....         | 317        | 17.1 了解基于位置的服务 .....                                         | 382        |
| 15.2.2 使用JCRMI与Java智能卡通信 .....             | 320        | 17.2 Location API简介 .....                                    | 383        |
| 15.2.3 利用SATSA高级API进行加<br>解密 .....         | 321        | 17.2.1 了解Location API .....                                  | 384        |
| 15.3 利用Bouncy Castle解决方案应对<br>安全挑战 .....   | 324        | 17.2.2 使用Location API确定设备<br>位置 .....                        | 385        |
| 15.3.1 使用Bouncy Castle API创建<br>消息摘要 ..... | 326        | 17.2.3 使用Location API管理路标 .....                              | 387        |
| 15.3.2 使用Bouncy Castle API加密和<br>解密 .....  | 326        | 17.2.4 理解安全性在LBS中扮演的<br>角色 .....                             | 388        |
| 15.4 使用无接触通信创建安全的商务<br>应用 .....            | 328        | 17.3 使用Location API .....                                    | 389        |
| 15.4.1 发现无接触目标 .....                       | 329        | 17.3.1 定位用户 .....                                            | 397        |
| 15.4.2 与无接触目标通信 .....                      | 331        | 17.3.2 在Sun Java Wireless Toolkit中<br>仿真Location API数据 ..... | 397        |
| 15.4.3 识别和生成可视标签 .....                     | 334        | 17.4 小结 .....                                                | 399        |
| 15.5 小结 .....                              | 338        | <b>第 18 章 寻找通用平台 .....</b>                                   | <b>400</b> |
| <b>第 16 章 呈现多媒体内容 .....</b>                | <b>340</b> | 18.1 理解JSR在碎片问题中扮演的角色 .....                                  | 400        |
| 16.1 MMAPI简介 .....                         | 340        | 18.1.1 JSR对碎片和统一问题的贡献 .....                                  | 400        |
| 16.1.1 理解基本多媒体概念 .....                     | 341        | 18.1.2 查阅JSR .....                                           | 401        |
| 16.1.2 理解MMAPI的组织结构 .....                  | 342        | 18.1.3 独立处理碎片问题 .....                                        | 403        |
| 16.1.3 开始呈现过程 .....                        | 345        | 18.2 了解JTWI .....                                            | 403        |
| 16.1.4 控制呈现过程 .....                        | 348        | 18.2.1 了解JTWI必需的元素 .....                                     | 404        |
| 16.1.5 捕获媒体 .....                          | 351        | 18.2.2 了解JTWI的可选元素 .....                                     | 404        |
| 16.1.6 播放单独的乐音 .....                       | 355        | 18.3 了解MSA .....                                             | 405        |
| 16.2 Java可缩放2D矢量图形API简介 .....              | 358        | 18.3.1 了解MSA 1.0 .....                                       | 405        |
|                                            |            | 18.3.2 未来演化: MSA2 .....                                      | 407        |
|                                            |            | 18.4 小结 .....                                                | 409        |
|                                            |            | <b>附录 查找 Java API .....</b>                                  | <b>410</b> |

# Part 1

## 第一部分

# 入 门

在开始编写Java ME代码之前，首先应该掌握其相关的基本知识。这正是本部分要介绍的内容：帮助你理解Java ME为何有用，如何使用NetBeans开始编写Java ME代码，以及Java ME与传统Java编程的不同之处。

### 本 部 分 内 容

- 第 1 章 开始 Java 之旅
- 第 2 章 精简 Java
- 第 3 章 NetBeans IDE 入门



虽然从本质上来看，Java ME只不过是 adapt 受限设备而对Java语言、类库和概念进行改编的结果，但事实上围绕Java ME开拓的业务是完全不同的。牢固掌握Java ME市场、平台和术语有助于你使用Java ME开发出成功的产品。

本章首先介绍Java ME的市场。接下来介绍Java ME平台，展示Sun如何识别和定义移动平台的基本需求，以及制造商、运营商和其他厂商是如何扩展这一基本平台的。最后讨论Java ME应用程序开发过程的不同之处，并说明了解你自己的消费群体、目标设备和分销渠道是多么地重要。

读完本章后，你会理解Java ME为何不同于Java，了解设备制造商、无线运营商和消费者如何看待Java ME，以及Java ME又是如何满足所有这些相关方需求的。具备这些知识后，你将能够更好地管理Java ME开发项目。

## 1.1 Java ME 市场简介

有三方势力支配着Java ME市场：试图在市场上使其产品具有独特优势的设备制造商，试图提供差异化服务并提高ARPU（average revenue per user，每用户平均收入）的无线运营商，以及使用新颖方式个性化其设备的消费者。

### 1.1.1 从设备制造商角度来看

设备制造商和无线运营商之间的相互影响较为复杂。制造商之间的持续竞争带来了差异化的产品，而与此同时，在许多市场中他们又求助于无线运营商来满足针对各种特性和功能的严格要求。

设备制造商大体上分为两类：OEM（Original Equipment Manufacturer，原始设备制造商）和ODM（Original Design Manufacturer，原始设计制造商）。OEM在其自有商标下制造设备，并将设备销售给消费者（可直接销售，也可通过运营商销售，或者二者兼而有之），而ODM代表其他厂商设计并制造硬件。虽然ODM和OEM都必须根据价格、质量和功能来区别其产品，但对于OEM来说，品牌和营销也是关键因素。

如今，许多无线运营商在提供给用户的大多数电话上只需要Java ME运行时。正如1.1.2节中所述，运营商正在寻求提高每个用户收入的方法，而数据服务就是其中的一种方法。如今，数