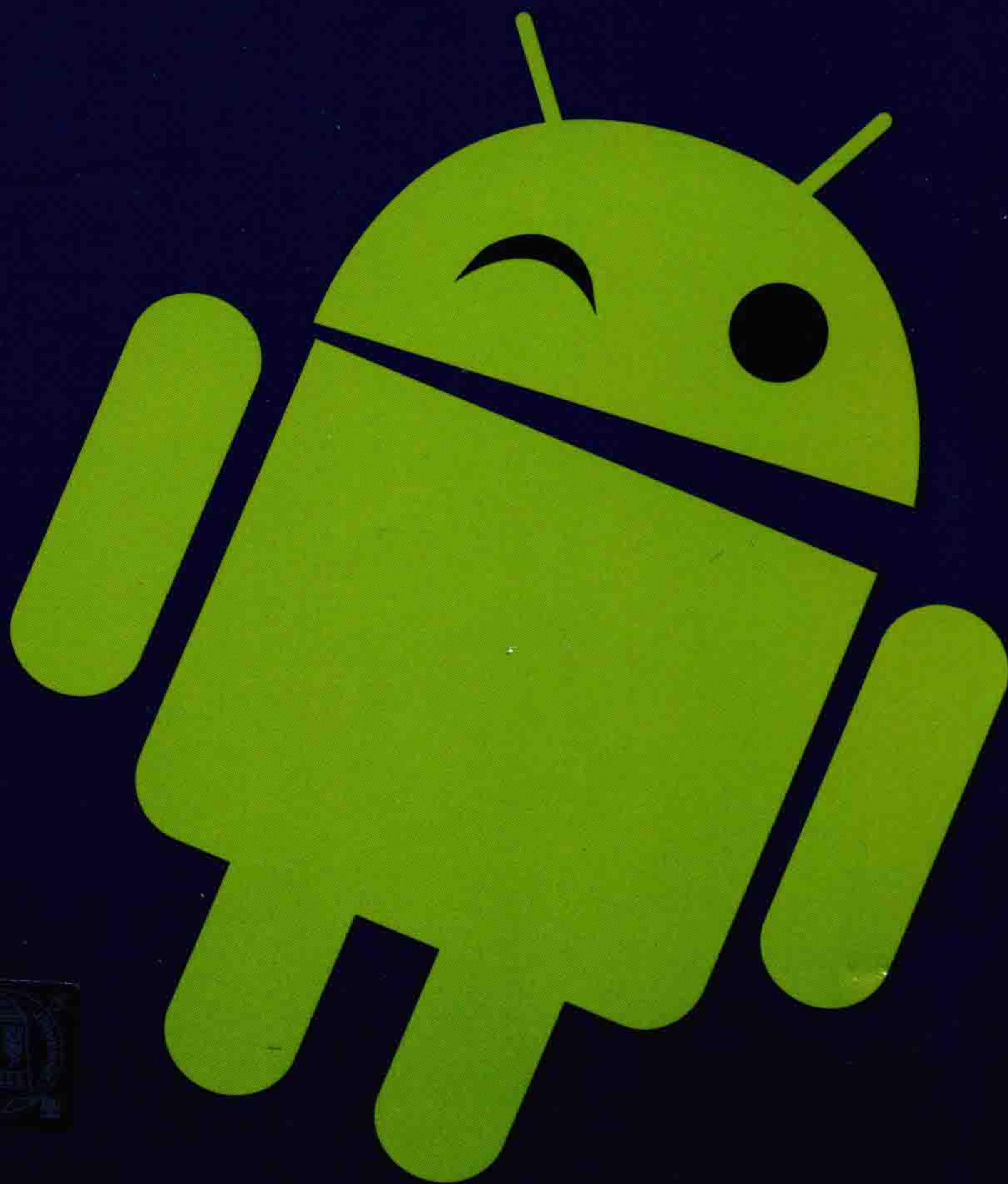


*Android Mobile Development
Technology*

Android 移动开发技术

张 勇 编著



清华大学出版社



Android Mobile Development Technology

Android 移动开发技术

张 勇 编著

清华大学出版社
北京

内 容 简 介

Android 系统是运行在智能移动设备上的嵌入式操作系统,包括 Linux 内核、系统库与 Java 运行时、应用程序框架层和应用程序层 4 部分,具有开源代码和免费使用的特点,是目前深受欢迎且全球用户数量最多的嵌入式操作系统。本书讲述基于 Android 系统的应用程序设计方法,全书分为 9 章,内容包括 Android 系统概述、Java 语言、Android 应用程序框架、单用户界面设计、多用户界面设计、数据访问技术、图形与动画、多媒体技术和通信应用技术。本书的特色在于原理讲解透彻,实例丰富且有代表性。

本书是作者近几年来从事嵌入式教学与研究的成果结晶,重点阐述了 Android 应用程序设计的理论与方法,适合作为普通高等院校软件工程、物联网、电子通信和智能控制等专业讲授 Android 移动开发技术的本科生教材或参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Android 移动开发技术/张勇编著. —北京:清华大学出版社,2017
ISBN 978-7-302-46659-8

I. ①A… II. ①张… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2017)第 036007 号

责任编辑:梁 颖 薛 阳

封面设计:常雪影

责任校对:白 蕾

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:清华大学印刷

经 销:全国新华书店

开 本:185mm×260mm

印 张:22.25

字 数:540 千字

版 次:2017 年 6 月第 1 版

印 次:2017 年 6 月第 1 次印刷

印 数:1~2000

定 价:69.00 元

产品编号:073219-01

前言

FOREWORD

智能手机的普及和移动计算技术的发展,促进了嵌入式操作系统的迅速发展。一般地,某类移动设备被称为“智能”的,就是指其加载了某种嵌入式操作系统,从而使其具有类似于计算机的功能,通过加载用户应用程序可以实现复杂的数据计算、控制处理和友好的人机交互工作。在智能移动设备中,嵌入式操作系统所起的作用与桌面视窗 Windows 系统类似,是用来管理设备系统软硬件资源的,而且嵌入式操作系统还具有实时性强、体积小和可裁剪的特点。目前,流行的嵌入式操作系统有 Google 的 Android、Apple 的 iOS、Microsoft 的 Windows CE(Windows 移动版)、嵌入式 Linux、WindRiver 的 VxWorks 和 Micrium 的 μ C/OS-III 等。其中,Android 和嵌入式 Linux 是免费且开源代码的, μ C/OS-III 是开源代码的。

Android 系统是基于 Linux 内核的嵌入式操作系统,严格地说是借用了 Linux 内核硬件驱动和线程调度功能,增加了与用户界面相关的应用程序设计框架和系统库。有些专家认为 Android 不过是类似于 QT 的用户界面程序。的确,Android 在用户界面方面具有强大的优势,其特色在于优化了图形显示技术并专门设计了图标。由于从程序员的角度出发,Android 系统向应用程序提供了完备的系统调用、进程管理与进程通信以及应用程序开发接口等,因此,Android 系统普遍被认可为嵌入式操作系统,并且在全球范围内得到用户和程序员的青睐。它的最大优势在于公开了源代码且可免费使用,并且其应用程序开发环境也是免费的。

本书基于 Google 推出的 Android Studio 集成开发软件,根据 Android 系统和应用软件(APP)设计技术的进展情况,阐述 Android 应用程序设计的最新原理与技术。全书共 9 章。

第 1 章为“概述”,介绍 Android 系统的发展历史和系统结构,并详细讲述 Java 语言程序设计的集成开发环境 Eclipse 和 Android 应用程序设计的集成开发环境 Android Studio。

第 2 章为“Java 语言”,介绍 Java 语言的语法和数据结构,深入讲解 Java 类、内部类和事件方法等概念。由于 Android 应用程序采用 Java 语言编写,因此,这一章的内容可使没有接触过 Java 语言的读者快速入门。

第 3 章为“Android 应用程序框架”,介绍 Hello World 工程及其结构,详细讲述 Hello World 工程的工作原理及应用程序框架的基本组成,并分析 Activity(活动界面)的生命周期。

第 4 章为“单用户界面应用设计”,讲述 Activity 的概念和使用方法,详细讲解应用程序布局方法和 Android 系统常用控件的种类及其使用方法。通过“计算器”工程深入分析单用

户界面程序设计的特点。

第5章为“多用户界面应用设计”，介绍 Intent 的概念和不同界面间的数据通信方法。与单用户界面程序相比，多用户界面程序需要进行界面间的数据通信，包括对话框与 Activity 之间、菜单与 Activity 之间以及两个 Activity 之间的数据通信，借助于内部类或 Intent 对象可实现这些通信方法。

第6章为“数据访问技术”，介绍 Android 系统的4种数据存储与访问方式，即 SharedPreferences 文件访问、流文件操作、SQLite 关系数据库和 Content Provider(内容提供者)等，并通过实例对比这4种方式的异同点。其中，Content Provider 可实现不同应用程序间的数据共享与通信。

第7章为“图形与动画”，详细讲述借助于 View 类和 SurfaceView 类进行图形绘制和动画设计的方法，介绍图形绘制的应用程序框架，阐述三种动画方式，即定时器动画、渐变动画和帧切换动画。

第8章为“多媒体技术”，介绍借助于 Media Player 类播放音频文件和视频文件的方法，并介绍 Service(服务)的程序设计方法。

第9章为“通信应用技术”，详细介绍基于 Android 智能手机进行短信通信的程序设计方法，本章工程的执行需要借助于真实的 Android 智能手机而非 Android 模拟器。

本书用于课内教学，建议理论学时为48，课内与课外实验学时为96。本书的特色在于 Android 应用程序设计原理讲解透彻，语言通俗易懂，程序实例丰富且具有代表性。书中全部工程实例的代码是完整的，可在学习和录入代码的过程中，体会到 Android 移动开发的乐趣，并试着对已有工程进行功能扩展和创新。本书实例代码约2.5GB，可以在百度云盘下载(链接：<http://pan.baidu.com/s/1ciM0s2> 密码：di67)。

特别强调的是：本书中的一些图像和音视频素材均来自于互联网，其版权归相关公司所有，书中引用的这些素材仅用于教学和研究。同时，作者保留其余内容的所有权利。

最后，感谢江西财经大学软件与通信工程学院唐颖军博士、陈滨博士和廖汉程博士等专家学者对本书提出的建设性意见，感谢研究生侯文刚、李雪倩和张琼阅读本书的初稿并订正了一些错误。由于作者水平有限，书中难免有纰漏之处，敬请同行专家和读者批评指正。

张 勇

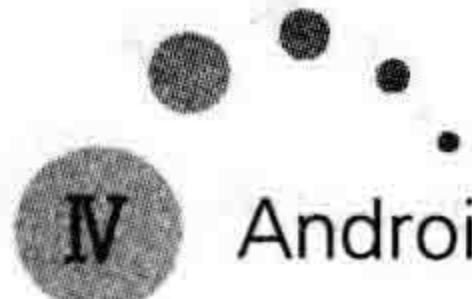
江西财经大学枫林园

2017年2月

目录

CONTENTS

第 1 章 概述	1
1.1 Android 操作系统	2
1.2 Android 系统结构	4
1.3 Java 开发环境	7
1.4 Android 开发环境	16
1.5 本章小结	34
第 2 章 Java 语言	35
2.1 Java 程序语法与控制	35
2.1.1 顺序方式	35
2.1.2 分支方式	40
2.1.3 循环方式	44
2.1.4 异常处理	47
2.2 Java 基本数据类型	49
2.2.1 数值	49
2.2.2 字符	50
2.2.3 字符串	51
2.2.4 布尔数	53
2.2.5 数组	53
2.3 Java 类	55
2.3.1 类与对象	56
2.3.2 继承与多态	57
2.3.3 接口	60
2.4 Java 文件操作	63
2.5 在命令行窗口中运行 Java 程序	67
2.6 Java 图形界面	69
2.6.1 事件响应方法	69
2.6.2 内部类	71



2.6.3 匿名内部类	72
2.7 本章小结	73
第3章 Android 应用程序框架	75
3.1 Hello World 工程	75
3.2 Hello World 应用工作原理	80
3.3 应用程序框架	86
3.3.1 应用程序框架基本组成	86
3.3.2 Android 配置文件 AndroidManifest.xml	87
3.3.3 Android 资源文件	90
3.3.4 Android 源程序文件	94
3.4 Activity 生命周期	97
3.5 本章小结	101
第4章 单用户界面应用设计	102
4.1 Activity 概念	102
4.2 布局与控件	103
4.2.1 布局软件 DroidDraw	103
4.2.2 控件事件响应方法	111
4.2.3 Android 常用控件	118
4.2.4 线性布局 LinearLayout	120
4.2.5 相对布局 RelativeLayout	122
4.2.6 框架布局 FrameLayout	125
4.2.7 表格布局 TableLayout 和 TableRow	126
4.2.8 约束布局 ConstraintLayout	129
4.3 “计算器”工程	135
4.4 本章小结	145
第5章 多用户界面应用设计	147
5.1 Intent 概念	147
5.2 对话框	150
5.2.1 AlertDialog 对话框	150
5.2.2 自定义对话框	155
5.2.3 Dialog 类	161
5.2.4 ProgressDialog 对话框	167
5.3 菜单	171
5.3.1 XML 布局菜单	172
5.3.2 动态菜单	175
5.3.3 上下文菜单	177

5.4	多用户界面设计	181
5.4.1	简单多用户界面显示	181
5.4.2	多用户界面数据传递	191
5.4.3	活动界面间双向数据通信	195
5.5	本章小结	207
第 6 章	数据访问技术	209
6.1	SharedPreferences 文件访问	209
6.2	流文件操作	224
6.3	SQLite 关系数据库	236
6.3.1	SQLite 数据库访问方法	236
6.3.2	SQLiteOpenHelper 类	243
6.4	内容提供者	250
6.5	本章小结	260
第 7 章	图形与动画	261
7.1	绘图	261
7.1.1	View 类绘图程序框架	262
7.1.2	SurfaceView 类绘图程序框架	271
7.1.3	基本图形与字符串	275
7.2	动画	276
7.2.1	定时器动画	276
7.2.2	渐变动画	287
7.2.3	帧切换动画	292
7.3	本章小结	296
第 8 章	多媒体技术	297
8.1	音频文件播放	297
8.2	服务	307
8.3	视频文件播放	312
8.4	本章小结	322
第 9 章	通信应用技术	323
9.1	短信息发送	323
9.2	短信息接收	328
9.3	短信息加密	334
9.4	本章小结	345
参考文献		346

第 1 章

概 述

Android 音译为“安卓”,Android 系统是安装在移动设备,例如智能手机、个人数字助理(PDA)、MP5 播放器、手持终端、平板计算机、上网本、电子书等上的操作系统软件,用于管理和调度移动设备的软硬件资源,其作用相当于个人计算机(PC)上安装的微软 Windows 视窗操作系统,与安装在智能手机上的 Windows Mobile Phone(Windows CE)操作系统相似。Android 系统与桌面 Windows 系统、Windows CE 操作系统的异同点列于表 1-1,从表 1-1 可以对 Android 系统有一个全面直观的认识。

表 1-1 Android 系统与桌面 Windows 系统、Windows CE 操作系统的异同点

比 较 项 目	Android 系统	桌面 Windows 系统	Windows CE 系统
用户界面	有	有	有
标准内存	256MB	2GB	256MB
系统大小	< 64MB	约 4GB	< 64MB
内核驻留	Flash 芯片	硬盘	Flash 芯片
实时性	强	弱	强
应用设备	嵌入式移动设备	个人计算机	嵌入式移动设备
功耗	低	高	低
应用软件体积	小	大	小
应用软件兼容性	极强	强	强
系统源代码	公开	不公开	核心不公开
内核独立性	基于 Linux	独立	独立
输入设备	触摸屏为主	键盘、鼠标为主	触摸屏为主
输出分辨率	相对低	相对高	相对低
英文直译	科幻机器人	视窗	便携式电子窗口
开发商	Google(谷歌)为主	微软(Microsoft)	微软

Android 系统由 Andy Rubin 首创,最初目的是设计一种新的开放性智能手机操作系统,2003 年美国已经有大量的移动设备嵌入式操作系统,新研发的操作系统想进入市场被用户认可是件很困难的事情。然而当时大部分嵌入式操作系统都不是开源的,维护十分困难,Rubin 等人因此提出这种开源的智能手机操作系统,希望借此挤进竞争激烈、商机无限的嵌入式操作系统市场中,赢利模式主要靠安装、维护和提供专业特色应用软件等技术服务。现在看来,Rubin 的做法成功了。2005 年 8 月 Google 收购 Android 加速了该开源嵌入式操作系统的发展,2007 年以 Google 为首组建了全球性的开放手机联盟(Open Handset Alliance),中国电信、中国移动和中国联通也是其中的成员,在全球范围内推动基于 Android 操作系统的手机开发计划。2008 年 10 月宏达电(HTC)公司推出了第一款 Android 系统的手机,命名为 HTC Dream(G1),如图 1-1(a)所示,这是一款被市场证实成功的手机。随后,几乎在全球范围内形成了研究 Android 操作系统的热潮,Android 操作系统功能和版本逐年提高,目前,已经是第 7.X 版,内部研发版本则更高。图 1-1(b)是基于 Android 2.3.3 版本的 Flyer 智能手机。2011 年初 Android 操作系统已经成为嵌入式操作系统领域最受欢迎的智能操作系统。

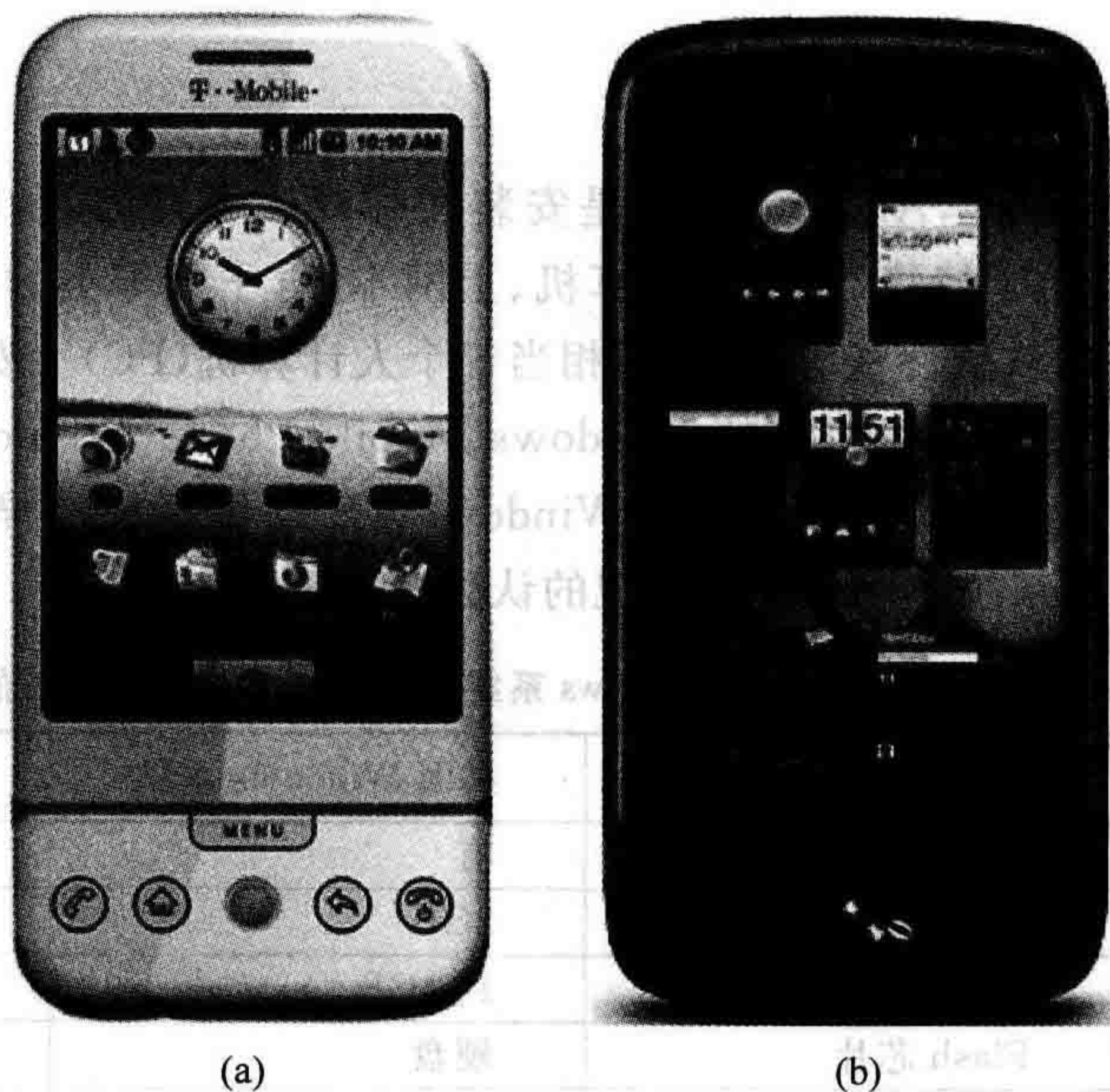


图 1-1 HTC Dream (G1)和 HTC Flyer

Android 操作系统是开放源代码的,并且拥有全球最多的研究人员和用户群,源文件中大量的 bug(问题)会被及时发现而纠正,因此,Android 系统版本号更新频繁。但是,基于 Android 系统的应用程序开发技术在各个版本中的方法完全相同,这些正是本书的内容(本书基于目前应用广泛的 Android 系统版本 4.4,同样适用于最新版本的 Android 系统)。

1.1 Android 操作系统

Android 操作系统是基于 Linux 内核的嵌入式操作系统,即其底层(称为第 I 层)为 Linux 操作系统及其驱动,该层源代码是 C 语言编写的;底层上面建构了系统库和 Java 运

运行时(即 Java 程序运行支持软件包或 Java 虚拟机,“运行时”是由 Runtime 意译而来,在很多书中均采用这一译法),称为第Ⅱ层,这一层是使用 C 或 C++代码写成的;第Ⅲ层为应用程序框架层,为用户开发 Android 程序直接提供 API(应用程序接口)函数,这一层是用 Java 代码实现的;第Ⅳ层为用户应用程序层,由于 Android 操作系统内置了许多用户应用程序,因此,有些专家认为应用程序层可以划分到 Android 操作系统中,当然,用户自己编写的应用程序也属于这一层,这一层的应用程序使用 Java 语言设计。Android 系统结构如图 1-2 所示。

第Ⅳ层 应用程序层 例如欢迎界面、浏览器、通话应用、联系人、日历等
第Ⅲ层 应用框架层 例如活动、窗口、通知、包、通信、资源、本地化管理器以及内容提供者、视图系统等
第Ⅱ层 系统库和Java运行时 例如图形界面管理器、多媒体框架、SQLite 数据库、C语言库、Java虚拟机等
第Ⅰ层 Linux内核 例如文件系统、显示驱动、电源管理、USB 驱动、蓝牙驱动等

图 1-2 Android 系统结构

Android 系统结构将在第 1.2 节中详细介绍。从图 1-2 中可以看出 Android 操作系统采用分层结构,且整个系统建立在 Linux 操作系统内核基础上,借助 Linux 内核硬件驱动进行硬件资源的管理。因此,Android 系统没有独立的硬件底层驱动部分,事实上,Android 系统的软件调度也借助了 Linux 内核进程调度实现,即两个显著的操作系统的特征在 Android 系统下没有得到体现,严格意义上讲,Android 系统应该隶属于应用软件系统的范畴。而与 Android 系统竞争市场的 Windows CE 操作系统则完全不同,它包括完整的内核层、驱动层和应用程序层,是真正意义上的嵌入式操作系统。如果从应用程序开发者的角度出发,而不去考虑图 1-2 所示的 Android 系统结构,此时,由于 Android 系统封装了各层间的通信和服务调用,向应用程序开发者提供了完备的系统调用(包括驱动程序开发)、进程管理与进程间通信和应用程序开发接口等,因此,从这个意义上说,Android 系统属于操作系统的范畴。现在,Android 系统研发者和应用程序开发者都普遍认可 Android 系统属于嵌入式操作系统,概念上将它与 Windows CE 等嵌入式操作系统等同。

Android 系统相对于其他嵌入式操作系统而言,具有两个明显的优点,即开放源代码和网络功能强大。前面提到了 Android 系统最初开放源代码的原因,从 2003 年到今天仍然保持着这一独特的优势,除了嵌入式操作系统领域市场竞争激烈外,Android 使用 Linux 作为其底层平台是其开源的另一个重要原因。Google 本身是互联网公司,其下的所有产品都是基于互联网模式发展的,Google 收购的 Android 系统也不会例外,伴随着 Android 系统的诞生和版本升级,Android 系统的网络功能越来越强大,这使得基于 Android 系统编写网络程序比基于其他任何嵌入式操作系统都更加容易。可以说,一部 Android 手机就是一部互联网终端,网上购物、新闻、旅游、导航、智能家居等应用的确给用户生活带来了极大的方便。

Android 系统使用 Java 语言编写应用程序,一定意义上可以说,Android 系统推动了 Java 语言的广泛应用。Java 语言属于面向对象的高级语言,Java 语言程序必须借助于 Java

虚拟机解释执行,比其他高级语言的可移植性都强,在 Android 模拟器上开发运行成功的应用程序,一定能够成功地部署和运行在 Android 系统终端机上,这使得 Android 系统应用程序开发变得异常方便。

Android 系统的图形界面也是它的一个亮点,严格地说,Android 系统不是基于可视化窗口的,而是直接基于图形的,也就是说,Android 系统界面是由一幅幅图画组合在一起的,因此,Android 系统界面比较“炫”。相比于 Windows CE 的视窗而言,人性化更强一些。Android 系统界面美观是其受到用户欢迎的最重要的原因,尽管如此,Google 对现有 Android 系统界面仍然不很满意,据说新版本的 Android 系统在用户界面上还要做大的创新。

目前最新的 Android 系统版本号为 7. X,研发代号为 Nougat(牛轧糖),重点在于扩展 Android 系统的人工智能应用。Android 系统是纯粹的商业性操作系统,在 GPL(General Public License)协议条件下源代码公开和免费使用,这意味着当用户免费使用 Android 系统开发软件产品时,其衍生的软件产品也必须是开源和免费的。需要指出的是,Android 虚拟机不是开源的。

1.2 Android 系统结构

图 1-2 为 Android 系统结构图,由于直接翻译 Android 系统组件的术语并不准确,因此,这里给出了经典的英文 Android 系统结构图,如图 1-3 所示。

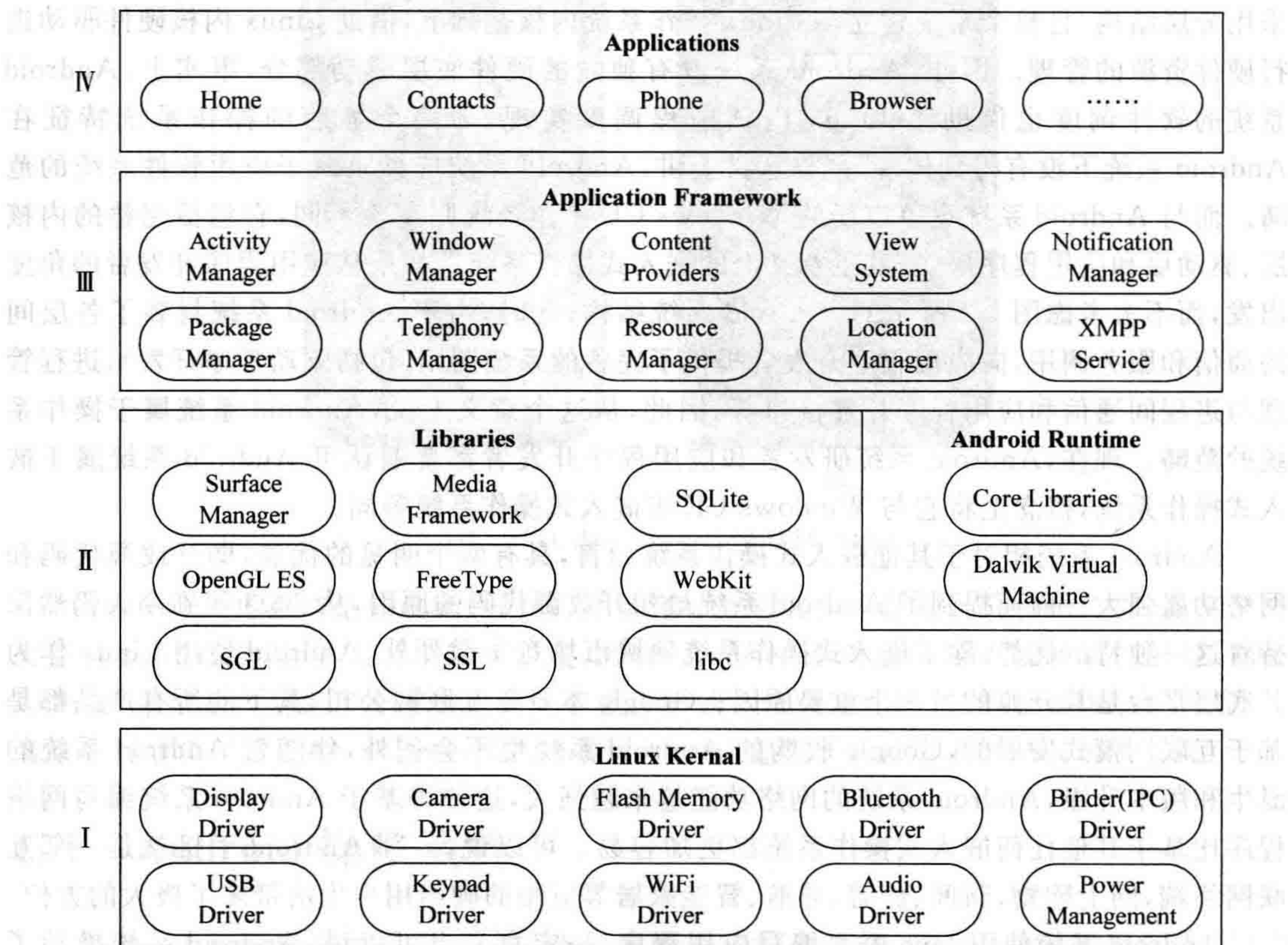


图 1-3 Android 系统结构图

由图 1-3 可知,Android 系统是基于 Linux 内核的操作系统,习惯上把 Linux 内核(Linux Kernal)层称为其第 I 层。Linux 是免费和公开源码的实时抢先式多任务操作系统,Linux 内核协助 Android 系统完成进程调度、进程间通信、内存管理、虚文件系统管理、系统安全管理和设备驱动等功能。图 1-3 中仅列举了 Linux 内核实现的 10 种设备驱动功能,即显示驱动、摄像驱动、Flash 存储驱动、蓝牙驱动、Binder IPC 驱动(用于进程间通信管理)、USB 驱动、键盘驱动、WiFi 驱动、音频驱动、功耗管理,事实上,Linux 内核还协助 Android 完成强大的网络管理和驱动等,被视为 Android 系统的一个硬件抽象层。正因为 Android 基于 Linux 内核,很多专家指出,要深入学习 Android 必须加强 Linux 系统的学习。如果重点放在 Android 应用程序设计上,即使不懂 Linux,对学习 Android 程序开发也影响不大。由于 Android 系统是架构在 Linux 系统之上的,Linux 系统不支持的处理器,Android 系统则同样无法支持,即 Android 系统只能移植到 Linux 系统可以运行的处理器上。好在 Linux 系统支持大多数流行的处理器,例如 x86 结构、ARM、MIPS 等,Linux 系统这种广泛的可移植性决定了 Android 系统具有广泛的可移植性。

第 II 层是 Android 系统库(Libraries)和 Android 运行时(Runtime),这一层是使用 C/C++代码编写的。Android 系统库包含了大量的类,通过第 III 层(即应用程序框架层)被应用程序开发者调用,应用程序开发者使用的大量 API(应用程序接口)函数来自于这些类。API 函数越丰富,则用户开发应用程序的工作量越小,随着 Android 系统版本的升级,API 函数级别也随之上升,如表 1-2 所示。

表 1-2 Android 系统与 API 级别的关系

Android 系统版本号	代 号	API 级别(level)
1.1	—	2
1.5	Cupcake	3
1.6	Donut	4
2.0	Eclair	5
2.0.1	Eclair	6
2.1	Eclair	7
2.2	Froyo	8
2.3.1	Gingerbread	9
2.3.3	Gingerbread	10
2.4	Gingerbread	未公开
3.0	Honeycomb	11
3.1	Honeycomb	12
3.2	Honeycomb	13
4.0	IceCreamSandwich	14
4.0.3	IceCreamSandwich	15
4.1	Jelly Bean	16
4.2	Jelly Bean	17
4.3	Jelly Bean	18
4.4	KitKat	19

续表

Android 系统版本号	代 号	API 级别(level)
4.4W	KitKat Wear	20
5.0	Lollipop	21
5.1	Lollipop	22
6.0	Marshmallow	23
6.X	Nougat	24

表 1-2 显示最新的 API 级别为 24,从表 1-2 中可以看出 Android 系统每个版本的研发代号都是一种小食品的名称,依次译为“杯形蛋糕”、“炸面圈”、“指形小饼”、“冻酸奶”、“姜饼”、“蜂巢”、“冰淇淋三明治”、“果冻豆”、“奇巧巧克力”、“棒棒糖”、“棉花糖”和“牛轧糖”,估计发起人也是一位美食家,可以预见下一个版本一定也是一种流行的受人喜爱的小食品名称,这正是 Android 系统研发的用意所在,希望得到系统研发者和应用程序开发者们的喜爱。

图 1-3 中第Ⅱ层 Android 系统库给出了 9 个组件,即 Surface Manager(界面管理器)、Media Framework(多媒体框架)、SQLite、OpenGL ES、FreeType、WebKit、SGL、SSL 和 libc。这 9 个组件都十分复杂,下面简要说明一下各个组件的作用。

Surface Manager(界面管理器)负责显示相关的操作,Android 系统界面是基于图形系统的,这种图形系统采用客户端/服务器的方式进行工作,“客户端”就是用户的应用程序界面,而“服务器”负责与这个应用程序界面相关的数据管理,由一个称为 Surface Flinger 的组件管理,“服务器”与“客户端”的通信需要借助于 Binder 类。界面管理器就是要管理这种工作方式,以实现二维或三维图形的显示。

Media Framework(多媒体框架)是个非常实用的类库,封装了大量处理多媒体数据的 API 函数,支持流行的绝大部分多媒体格式,使得应用程序开发者开发多媒体软件异常轻松,只需要调用多媒体框架中的 API 函数就行了。

SQLite 是 Android 集成的关系型数据库,SQLite 是开源代码的嵌入式数据库,支持 ANSI SQL92 标准的大部分 SQL(结构化查询语言)语句,速度快,体积小(约 250KB),最大支持数据库文件大小为 4TB。

OpenGL ES 是 Android 系统中二维和三维图形处理与加速的 API 函数集。

SGL 是 Skia Graphics Library 的首字母简写,是 Android 用来处理二维图形的向量图形引擎。所谓的“引擎”术语是从汽车借用过来的,在计算机软件中的引擎是指软件处理中的最核心部分,就像汽车发动机是汽车的核心一样,一个软件功能的升级主要取决于其“引擎”部分的升级。SGL 就是 Android 系统的二维图形引擎。

FreeType 是 Android 系统使用的字体引擎,FreeType 是公开源码和免费的,其优点在于提供简单、统一的 API 函数访问多种字体格式文件,例如位图字体和矢量字体,这使得 Android 系统处理字体非常方便。

SSL 是 Secure Sockets Layer 的首字母简称,即安全套接层,这里说明 Android 系统支持 SSL,即安全套接层协议。SSL 对发送的网络数据进行加密,防止数据在传送至合法目的地网络终端的过程中被非法用户使用或修改,这在电子商务和网上银行的数据交换中尤为

重要,SSL 通过对合法用户的认证和数据的加密,确保用户的信息安全,Android 系统自诞生以来,就支持 SSL 协议。事实上,所有的嵌入式操作系统浏览器都支持 SSL 协议。

WebKit 是公开源码的网络浏览器引擎,这个引擎稳定性好、兼容性好、效率高,不仅 Android 系统浏览器基于 WebKit 引擎,苹果的 iPhone 浏览器也基于此引擎。

libc 是通用的 C 语言库,供 Android 系统库调用。

Android 运行时(Runtime)包括核心库(Core Libraries)和 Dalvik 虚拟机(Virtual Machine)。核心库集成了绝大多数 Java 语言核心库的功能,供 Java 语言程序运行时调用。Dalvik 虚拟机解释并运行格式为 dex 的 Java 程序,dex 术语是 Dalvik eXecutable 的缩写,常规的 Java 语言程序(class 字节文件)通过 Android 系统内置的 dx 工具转化为 dex 格式,这种格式被优化为代码内存占用最小,因此,Android 可执行文件扩展名为 .dex。每个 Android 应用程序启动后都对应着一个进程,该进程属于它自己的 Dalvik 虚拟机实例,Dalvik 虚拟机可以同时高效地运行多个虚拟机实例,从而实现多任务处理。

第Ⅲ层为 Android 应用程序框架,进行 Android 应用程序开发必须熟练掌握其 4 种基本组件,即活动(Activity)、服务(Service)、广播接收器、内容提供者(Content Provider)的使用技术。应用程序框架层为开发 Android 应用程序提供了各种 API 函数,这些函数属于不同的类。图 1-3 中列出 10 类组件,即活动管理器(Activity Manager)、窗口管理器(Window Manager)、内容提供者(Content Providers)、视图系统(View System)、通知管理器(Notification Manager)、包管理器(Package Manager)、电话管理器(Telephony Manager)、资源管理器(Resource Manager)、XMPP 服务(XMPP Service)。其中,XMPP 是 Google Talk 的通信协议,简称 GTalk,是 Google 的即时通信方式,就是通常所说的文字或语音聊天,此外,GTalk 还支持 E-mail 功能。第Ⅲ层是进行应用程序开发的基础,也是本书涉及的主要内容,这一层的组件将在后续的章节中详细介绍。

第Ⅳ层为用户应用程序层,这一层的软件包括欢迎界面(Home)、联系人(Contacts)、电话(Phone)、浏览器(Browser)等用户直接使用的程序,当然,也包括用户自己开发的应用程序。本书将详细介绍该层应用程序的设计方法。

1.3 Java 开发环境

Android 程序的开发语言为 Java。Android 应用程序选择 Java 作为其开发语言,而不选用 C 或 C++,其原因在于通过 Java 虚拟机可使 Android 应用程序运行于任意平台上,即 Java 程序具有比 C 和 C++ 语言程序更强的可移植性。

微软公司的 Microsoft Visual Studio(MVS)和 Borland 公司的 Borland Developer Studio(BDS)均集成了优秀的 Java 开发环境。除了 MVS 和 BDS 之外,开源的 Eclipse 是倍受欢迎的集成开发环境,其中集成了 Java 开发环境和 C/C++ 开发环境等。本书将使用 Eclipse 作为 Java 开发环境。

首先从 www.eclipse.org 上下载最新的 Java 开发环境,截至 2016 年 7 月,最新的 Java 开发环境代号为 Neon,建议读者使用最新版本的 Eclipse 软件。如果使用了 64 位的 Windows 视窗系统,则下载其相应的 Windows 64 位版本,压缩包为 eclipse-java-neon-win32-x86_64.zip(该压缩包文件名对应 Neon 版本号,不同的版本号将对应不同的文件

名)。现将 eclipse-java-neon-win32-x86_64.zip 解压缩到 D:\Program Files 目录下,如图 1-4 所示。

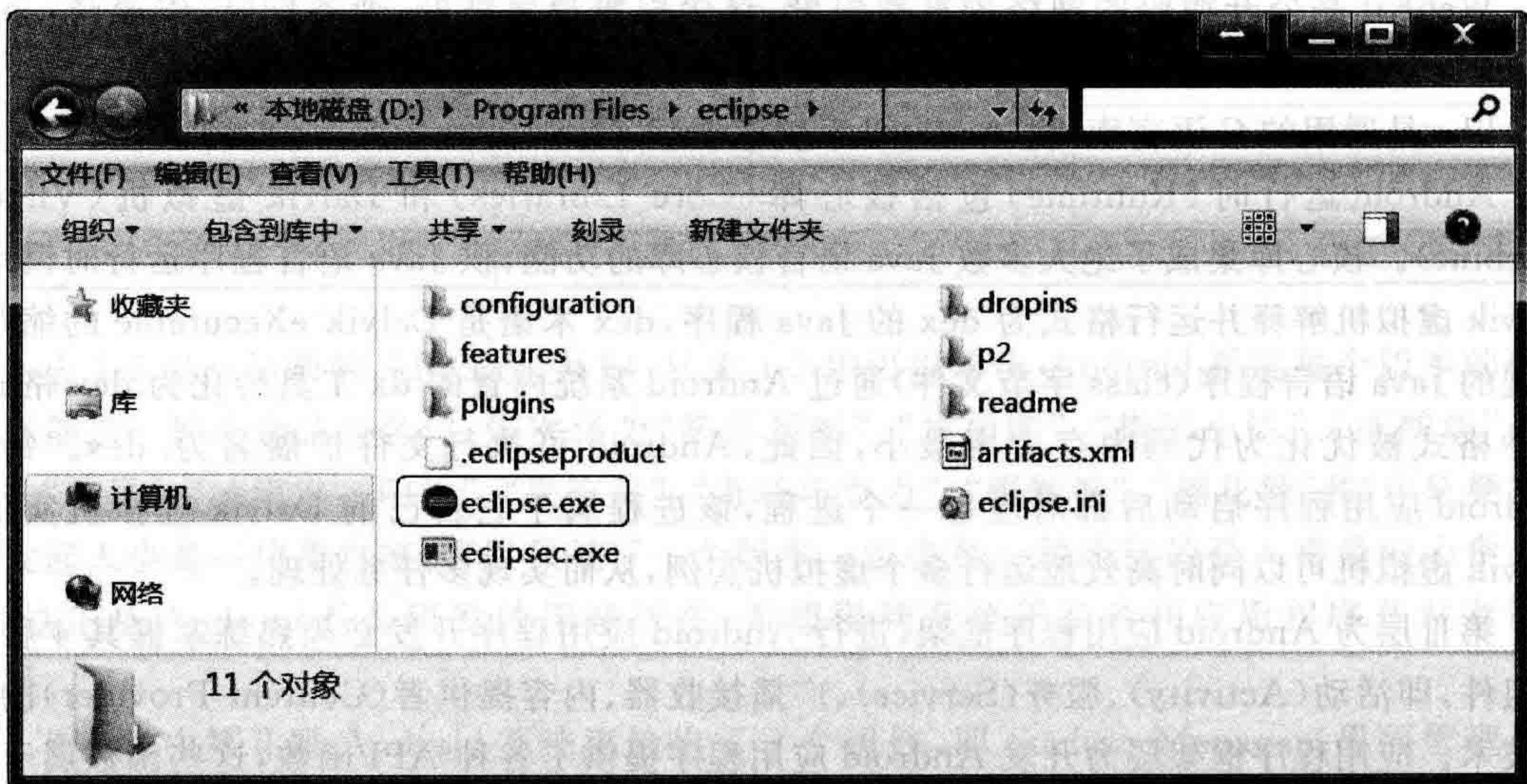


图 1-4 Eclipse Neon 文件目录结构

将图 1-4 中的 eclipse.exe 发送到桌面快捷方式,修改快捷方式的名称为 Eclipse Neon,如图 1-5 所示。

在图 1-5 中双击 Eclipse Neon 图标可进入 Java 开发环境。但在运行 Eclipse 软件之前,需要先安装 Java 运行支持环境,即通常所说的 Java 虚拟机,Java 虚拟机是解释并运行 Java 程序的软件包,通过 Java 虚拟机可使 Java 程序运行在 Windows 系统上,如图 1-6 所示。



图 1-5 Eclipse Neon 桌面快捷方式

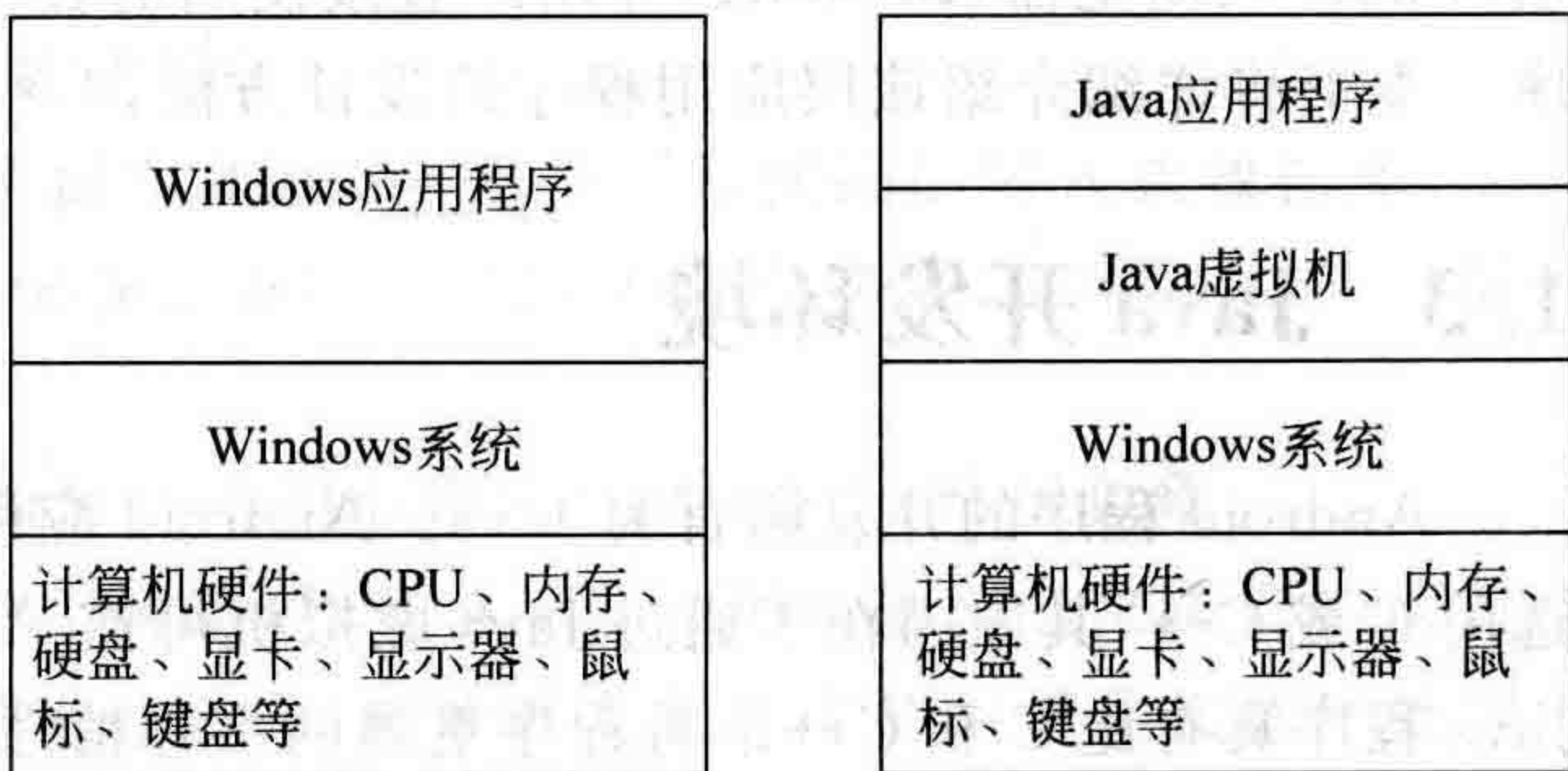


图 1-6 Java 虚拟机的桥梁作用

在 www.oracle.com 上下载 Java 虚拟机,Java 虚拟机更新非常频繁,这里使用了截至 2016 年 7 月的最新版本 Java SE Development Kit 8u102,其 Windows 64 位版本的下载文件为 jdk-8u102-windows-x64.exe,文件名中的 8u102 为版本号。

双击 jdk-8u102-windows-x64.exe 文件(对于 Windows 7 以上系统,尽量使用“以管理员身份运行”),进入图 1-7 所示的安装界面。

在图 1-7 中,单击“下一步”按钮,进入图 1-8 所示的界面。

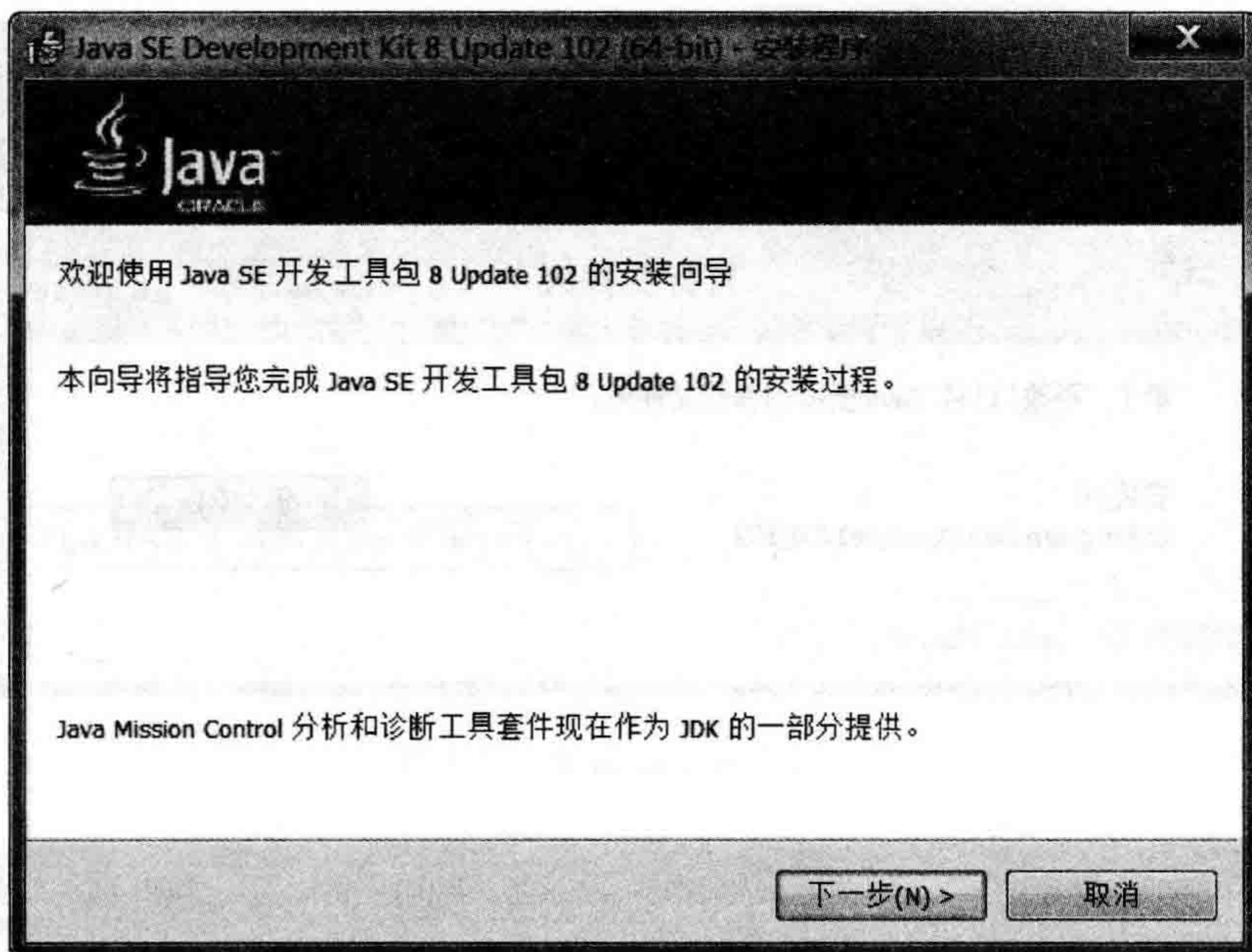


图 1-7 Java 虚拟机安装向导

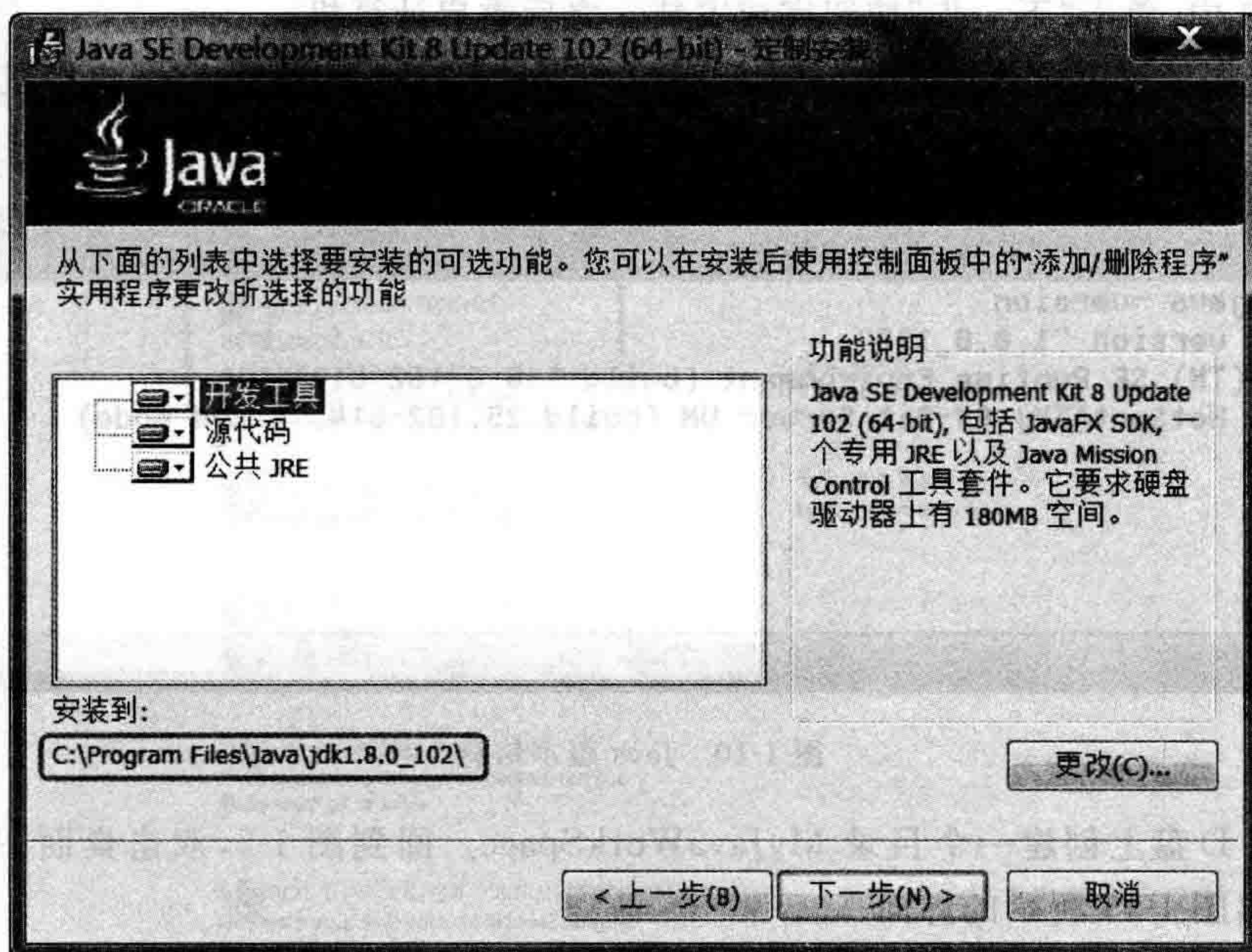


图 1-8 Java 虚拟机安装目录

在图 1-8 中,设定好安装目录,单击“下一步”按钮,然后按照后续的提示,将 Java 虚拟机安装到计算机上。

安装过程中,会提示安装 Java JRE(Java 运行时,即 Java 运行支持库,是严格意义上的 Java 虚拟机),如图 1-9 所示。