

精通

王石磊◆编著

Android 5

多媒体开发



详细地讲解了 Android 多媒体开发的主要应用

Android 多媒体框架详解、音频系统框架详解、视频系统框架详解、照相机系统详解、Alarm 时钟系统详解、振动器系统详解、二维图像渲染详解、绘制二维图像、二维动画应用、渲染二维图像、开发音频应用程序、开发视频应用程序、OpenGL ES 系统基本应用、纹理映射、绘制不同的三维形状、坐标变换和混合、开发一个屏保系统、开发一个音乐播放器、开发一个闹钟系统等。



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

精通



Android 5

多媒体开发

王石磊◆编著

人民邮电出版社

北京

图书在版编目(CIP)数据

精通Android 5多媒体开发 / 王石磊编著. — 北京 :
人民邮电出版社, 2015. 11
ISBN 978-7-115-38758-5

I. ①精… II. ①王… 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(2015)第073608号

内 容 提 要

在整个 Android 应用开发体系中, 图像、音频、视频、渲染和 3D 处理一直是其核心应用。本书分为 5 篇, 共计 24 章, 循序渐进地讲解 Android 多媒体应用开发的基本知识, 遵循从底层原理开始到顶层应用结束的开发过程, 全程剖析 Android 多媒体应用开发的所有核心知识点。本书从获取并编译 Android 源码开始讲起, 依次讲解基本技术、系统分析、典型应用、三维技术、综合实战这 5 大部分的知识。在讲解每一个知识点时, 都从基础理论开始入手, 遵循由浅入深的写作方法, 按照运作流程逐步分析 Android 多媒体应用的方方面面。本书几乎涵盖 Android 多媒体系统的所有主要内容。

本书适合 Android 爱好者、Android 初学者、Android 应用开发者、Android 视频/音频开发者、Android 游戏开发者, 也可以作为相关培训学校和大专院校相关专业的教学用书。

◆ 编 著 王石磊

责任编辑 张 涛

责任印制 张佳莹 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京中新伟业印刷有限公司印刷

◆ 开本: 787×1092 1/16

印张: 36.5

字数: 1016 千字

印数: 1—2 500 册

2015 年 11 月第 1 版

2015 年 11 月北京第 1 次印刷

定价: 89.00 元

读者服务热线: (010) 81055410 印装质量热线: (010) 81055316

反盗版热线: (010) 81055315

广告经营许可证: 京崇工商广字第 0021 号

前言

Android 系统是一款于 2007 年问世的基于 Linux 平台的开源手机操作系统，该平台由操作系统、中间件、用户界面和应用软件组成，是首个为移动终端打造的真正开放和完整的移动软件。

本书的内容

本书依次讲解了 Android 技术概述、Android 技术核心框架分析、获取并分析 Android 源码、Android 多媒体框架、音频系统框架、视频系统框架、照相机系统、Alarm 时钟系统、振动器系统、二维图像渲染、绘制二维图像、二维动画应用、渲染二维图像、开发音频应用程序、开发视频应用程序、OpenGL ES 系统初步、OpenGL ES 基本应用、纹理映射、绘制不同的三维形状、坐标变换和混合、OpenGL ES 进阶、开发一个屏保系统、开发一个音乐播放器、开发一个闹钟系统等知识。

本书版本

Android 系统自 2008 年 9 月发布第一个版本 1.1 以来，截至 2014 年 10 月发布的最新版本 5.0，一共存在十多个版本。由此可见，Android 系统升级频率较快，一年之中最少有两个新版本诞生。如果过于追求新版本，会造成力不从心的结果。在此建议广大读者不必追求最新的版本，只需关注最流行的版本即可。据官方统计，截至 2014 年 12 月 15 日，占据前 3 位的版本分别是 Android 4.4、Android 4.3 和 Android 4.2，其实这 3 个版本的区别并不是很大，只是在某领域的细节上进行了更新。

本书的内容以笔者撰稿时的最新版本 Android 5.0 为基础，并且兼容了 Android L 及其以前的版本。

本书特色

本书内容丰富、全面。我们的目标是通过一本图书提供多本图书的价值，读者可以根据自己的需要有选择地阅读。在内容的编写上，本书具有以下特色。

(1) 结构合理。

从用户的实际需要出发，合理安排知识结构，内容由浅入深，详细地讲解了和 Android 多媒体应用开发有关的知识。

(2) 遵循“理论介绍—演示实例—综合演练”主线。

为了使广大读者彻底弄清楚 Android 多媒体应用开发的每一个知识点，在讲解时依次剖析了基本理论、演示实例分析、综合实战演练等内容，遵循了从理论到实践的原则，实现了实践教学这一目标。

(3) 易学易懂。

本书内容条理清晰，语言简洁，读者既可以按照本书编排的章节顺序进行学习，也可以根据自己的需求对某一章节进行有针对性的学习。

(4) 实用性强。

本书彻底摒弃枯燥的理论和简单的操作，注重实用性和可操作性，详细讲解了各个知识点的实战知识。

读者对象

初学 Android 编程的自学者；

多媒体开发人员；

游戏开发人员；

大中专院校的老师和学生；

做毕业设计的学生；

Android 编程爱好者；

相关培训机构的老师和学员；

从事 Android 开发的程序员。

在编写本书的过程中，得到了人民邮电出版社工作人员的大力支持，正是各位编辑的求实、耐心和效率，才使得本书在这么短的时间内出版。另外，也十分感谢我的家人在我写作的时候给予的巨大支持。本人毕竟水平有限，本书如有纰漏和不尽如人意之处，诚请读者提出意见或建议，以便修订并使之更臻完善。另外，我们提供了答疑和源程序下载支持网站 <http://www.toppr.net/>，读者如有疑问可以在此提出，一定会得到满意的答复。

编 者

目 录

第一篇 基础技术篇

第 1 章 Android 技术概述.....2

- 1.1 智能手机系统介绍.....2
 - 1.1.1 何谓智能手机.....2
 - 1.1.2 看当前主流的智能
手机系统.....2
 - 1.1.3 Android 5.0 的突出变化.....3
- 1.2 搭建 Android 应用开发环境.....4
 - 1.2.1 安装 Android SDK 的
系统要求.....4
 - 1.2.2 安装 JDK.....5
 - 1.2.3 获取并安装 Eclipse 和
Android SDK.....8
 - 1.2.4 安装 ADT.....10
 - 1.2.5 设定 Android SDK Home.....12
 - 1.2.6 验证开发环境.....13
 - 1.2.7 创建 Android 虚拟
设备 (AVD).....13
 - 1.2.8 启动 AVD 模拟器.....16
 - 1.2.9 解决搭建环境过程中的
常见问题.....18

第 2 章 Android 技术核心框架分析.....21

- 2.1 简析 Android 安装文件.....21
 - 2.1.1 Android SDK 目录结构.....21
 - 2.1.2 android.jar 及内部结构.....22
 - 2.1.3 阅读 SDK 帮助文档.....22
 - 2.1.4 常用的 SDK 工具.....23
- 2.2 演示官方实例.....24
- 2.3 剖析 Android 系统架构.....28
 - 2.3.1 Android 体系结构介绍.....28
 - 2.3.2 Android 应用工程文件组成.....30
- 2.4 简述五大组件.....32
 - 2.4.1 用 Activity 来表现界面.....32
 - 2.4.2 用 Intent 和 IntentFilter
实现切换.....33
 - 2.4.3 Service 为你服务.....33
 - 2.4.4 用 BroadcastReceiver
发送广播.....34
 - 2.4.5 用 ContentProvider 存储数据.....34

- 2.5 进程和线程.....34
 - 2.5.1 先看进程.....34
 - 2.5.2 再看线程.....35
 - 2.5.3 应用程序的生命周期.....35
- 2.6 第一段 Android 程序.....37

第 3 章 获取并分析 Android 源码.....42

- 3.1 获取 Android 源码.....42
 - 3.1.1 在 Linux 系统中获取
Android 源码.....42
 - 3.1.2 在 Windows 系统中获取
Android 源码.....43
- 3.2 分析 Android 源码结构.....45
- 3.3 编译 Android 源码.....46
 - 3.3.1 搭建编译环境.....47
 - 3.3.2 开始编译.....48
 - 3.3.3 在模拟器中运行.....49
 - 3.3.4 常见的错误分析.....49
 - 3.3.5 实践演练——演示两种编译
Android 程序的方法.....50

第二篇 系统分析篇

第 4 章 Android 多媒体框架.....56

- 4.1 Android 多媒体系统介绍.....56
- 4.2 OpenMax 框架详解.....57
 - 4.2.1 分析 OpenMax 框架构成.....58
 - 4.2.2 实现 OpenMax IL 层接口.....62
- 4.3 分析 OpenCore 框架.....68
 - 4.3.1 OpenCore 层次结构.....69
 - 4.3.2 OpenCore 代码结构.....70
 - 4.3.3 OpenCore 编译结构.....71
 - 4.3.4 操作系统兼容库.....74
 - 4.3.5 实现 OpenCore 中的
OpenMax 部分.....75
 - 4.3.6 OpenCore 扩展详解.....85
- 4.4 Stagefright 框架详解.....91
 - 4.4.1 Stagefright 代码结构.....91
 - 4.4.2 Stagefright 实现
OpenMax 接口.....91
 - 4.4.3 分析 Video Buffer 传输流程.....94

第 5 章	音频系统框架	99
5.1	音频系统基础	99
5.2	分析音频系统的层次	100
5.2.1	层次说明	100
5.2.2	Media 库中的 Audio 框架	101
5.2.3	本地代码	103
5.2.4	分析 JNI 代码	105
5.2.5	Java 层代码简介	106
5.3	Audio 系统的硬件抽象层	106
5.3.1	Audio 硬件抽象层基础	107
5.3.2	分析 AudioFlinger 中的 Audio 硬件抽象层的实现	108
5.3.3	真正实现 Audio 硬件 抽象层	113
5.4	分析编码/解码过程	113
5.4.1	AMR 编码	114
5.4.2	AMR 解码	117
5.4.3	解码 MP3	120
第 6 章	视频系统框架	122
6.1	视频系统结构	122
6.2	分析硬件抽象层	123
6.2.1	Overlay 系统硬件抽象层 的接口	123
6.2.2	实现 Overlay 系统的硬件 抽象层	125
6.2.3	实现接口	126
6.3	实现 Overlay 硬件抽象层	127
第 7 章	照相机系统	129
7.1	Camera 系统的结构	129
7.2	Camera 驱动层实现详解	131
7.2.1	V4L2 驱动程序	131
7.2.2	硬件抽象层	137
7.3	实现 Camera 系统的硬件抽象层	141
7.3.1	Java 程序部分	141
7.3.2	Camera 的 Java 本地 调用部分	142
7.3.3	Camera 的本地库 libui.so	142
7.3.4	Camera 服务 libcameraservice.so	143
第 8 章	Alarm 时钟系统	147
8.1	Alarm 系统基础	147
8.2	分析 RTC 驱动程序	148
8.3	Alarm 驱动程序详解	149

8.3.1	分析文件 android_alarm.h	149
8.3.2	分析文件 alarm.c	151
8.3.3	分析文件 alarm-dev.c	160
8.4	JNI 层详解	167
8.5	Java 层详解	168
8.5.1	分析 AlarmManagerService 类	168
8.5.2	分析 AlarmManager 类	176

第 9 章	振动器系统	178
9.1	振动器系统结构	178
9.2	硬件抽象层实现详解	179
9.3	分析 Java 层部分	181
9.4	实现硬件抽象层	184

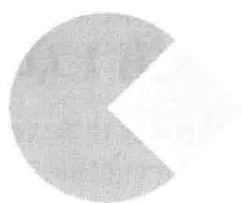
第三篇 典型应用篇

第 10 章	二维图像渲染	186
10.1	SurfaceFlinger 渲染管理器	186
10.1.1	SurfaceFlinger 基础	186
10.1.2	Surface 和 Canvas	188
10.2	Surface 渲染详解	189
10.2.1	渲染类 Surface 详解	189
10.2.2	分析 Layer 和 LayerBuffer	191
10.3	Skia 渲染引擎详解	196
10.3.1	Skia 基础	196
10.3.2	Android 中的 Skia	197
10.3.3	使用 Skia 绘图	205
10.3.4	Skia 的其他功能	206
第 11 章	绘制二维图像	208
11.1	绘图界面布局详解	208
11.1.1	View 视图组件	208
11.1.2	ViewGroup 容器	208
11.1.3	Layout 规划布局	209
11.2	Android 绘图基础	212
11.2.1	使用 Canvas 画布	212
11.2.2	使用 Paint 类	214
11.2.3	位图操作类 Bitmap	217
11.3	使用其他的绘图类	222
11.3.1	使用设置文本颜色类 Color	223
11.3.2	使用矩形类 Rect 和 RectF	224
11.3.3	非矢量图形拉伸类 NinePatch	228
11.3.4	使用变换处理类 Matrix	228
11.3.5	使用 BitmapFactory 类	231

11.3.6 使用 Region 类	233	14.4 播放音频	297
11.3.7 使用类 Typeface	234	14.4.1 使用 AudioTrack 播放音频	297
第 12 章 二维动画应用	235	14.4.2 使用 MediaPlayer 播放音频	300
12.1 使用 Drawable 实现动画效果	235	14.4.3 使用 SoundPool 播放音频	311
12.1.1 Drawable 基础	235	14.4.4 使用 Ringtone 播放铃声	316
12.1.2 使用 Drawable 实现动画效果	236	14.4.5 使用 JetPlayer 播放音频	318
12.2 Tween Animation 动画详解	237	14.4.6 使用 AudioEffect 处理音效	319
12.2.1 Tween 动画基础	237	14.5 语音识别技术	321
12.2.2 Tween 动画类详解	240	14.5.1 Text-To-Speech 技术	322
12.2.3 Tween 应用实战	243	14.5.2 谷歌的 Voice Recognition 技术	324
12.3 实现 Frame Animation 动画效果	246	14.6 实现振动效果	326
12.3.1 Frame 动画基础	246	14.6.1 Vibrator 类基础	326
12.3.2 使用 Frame 动画	246	14.6.2 使用 Vibrator 实现振动效果	327
12.4 Property Animation 动画	248	14.7 设置闹钟	332
12.4.1 Property Animation (属性) 动画基础	248	14.7.1 AlarmManage 基础	332
12.4.2 使用 Property Animation	250	14.7.2 开发一个闹钟程序	333
12.5 实现动画效果的其他方法	253	第 15 章 开发视频应用程序	338
12.5.1 播放 GIF 动画	254	15.1 使用 MediaPlayer 播放视频	338
12.5.2 实现 EditText 动画特效	256	15.2 使用 VideoView 播放视频	344
第 13 章 渲染二维图像	257	15.2.1 VideoView 基础	344
13.1 使用渲染类 Shader	257	15.2.2 使用 VideoView 播放手机中的影片	346
13.2 使用 SurfaceFlinger 渲染器	261	15.2.3 使用 VideoView 播放手机中的 MP4	348
13.2.1 SurfaceFlinger 基础	261	15.2.4 开发一个网络视频播放器	350
13.2.2 渲染 Android 手机屏幕中的图形	263	15.3 使用 Camera 拍照	356
13.3 使用 Skia 渲染引擎	265	15.3.1 Camera 基础	356
13.3.1 Skia 基础	265	15.3.2 总结 Camera 拍照的流程	360
13.3.2 使用 Skia 绘图	268	15.3.3 使用 Camera 预览并拍照	362
13.4 通过 Skia 绘制文字	275	15.3.4 使用 Camera API 方式拍照	366
第 14 章 开发音频应用程序	277	第四篇 三维技术篇	
14.1 音频应用接口类介绍	277	第 16 章 OpenGL ES 系统初步	372
14.2 AudioManager 类	278	16.1 OpenGL ES 介绍	372
14.2.1 AudioManager 基础	278	16.1.1 OpenGL ES 3.0 介绍	372
14.2.2 AudioManager 基本应用——设置短信提示铃声	280	16.1.2 Android 全面支持 OpenGL ES 3.0	373
14.2.3 AudioManager 基本应用——调节手机音量的大小	284	16.2 OpenGL ES 3.0 系统初步分析	373
14.3 录音处理	287	16.3 分析下层的包裹库	374
14.3.1 使用 MediaRecorder 接口录制音频	287	16.3.1 libGLESv1_CM.so 包裹库详解	374
14.3.2 使用 AudioRecord 接口录制音频	292	16.3.2 libGLESv2 包裹库详解	379

16.3.3	libEGL 包裹库详解	381	20.2.4	实现滤光器效果	469
16.4	加载并解析 OpenGL 库	383	第 21 章	OpenGL ES 进阶	474
16.4.1	开始加载并解析	383	21.1	实现摄像机和雾特效功能	474
16.4.2	库加载器 Loader 详解	384	21.1.1	摄像机基础	474
16.5	EGL 实现详解	388	21.1.2	雾特效基础	475
16.5.1	分析 EGL 的数据结构	388	21.1.3	实现雾特效和摄像机效果	475
16.5.2	分析 EGL 的 API	392	21.2	粒子系统	484
第 17 章	OpenGL ES 基本应用	401	21.2.1	粒子系统基础	484
17.1	OpenGL ES 的基本应用	401	21.2.2	实现粒子系统效果	484
17.1.1	使用点线法绘制三角形	401	21.3	镜像技术	487
17.1.2	使用索引法绘制三角形	405	21.4	实现旗帜飘扬效果	491
17.1.3	使用顶点法绘制三角形	409			
17.2	实现投影效果	411	第五篇	综合实战篇	
17.2.1	正交投影	411	第 22 章	开发一个屏保系统	494
17.2.2	透视投影	411	22.1	屏幕保护程序介绍	494
17.2.3	正交投影和透视投影 的区别	412	22.1.1	屏幕保护程序的作用	494
17.2.4	实现投影效果	412	22.1.2	手机中的屏幕保护程序	494
17.3	实现光照效果	415	22.2	开发屏保程序的原理	495
17.3.1	光源的类型	416	22.3	开发一个屏保程序	496
17.3.2	光源的颜色	416	22.3.1	准备素材图片	496
17.3.3	开启/关闭光照	417	22.3.2	编写布局文件	496
第 18 章	纹理映射	422	22.3.3	编写主程序文件	497
18.1	纹理映射基础	422	第 23 章	开发一个音乐播放器	507
18.1.1	纹理贴图和纹理拉伸	422	23.1	项目介绍	507
18.1.2	Texture Filter 纹理过滤	423	23.1.1	项目背景介绍	507
18.2	实现三角形纹理贴图效果	424	23.1.2	项目的目的	508
18.3	实现地月模型效果	427	23.2	系统需求分析	508
18.4	实现纹理拉伸效果	434	23.2.1	构成模块	508
第 19 章	绘制不同的三维形状	438	23.2.2	系统流程	512
19.1	绘制一个圆柱体	438	23.2.3	功能结构图	513
19.2	绘制一个圆环	444	23.2.4	系统功能说明	514
19.3	绘制一个抛物面效果	448	23.2.5	系统需求	514
19.4	绘制一个螺旋面效果	450	23.3	数据库设计	515
第 20 章	坐标变换和混合	454	23.3.1	字段设计	515
20.1	实现坐标变换	454	23.3.2	E-R 图设计	515
20.1.1	坐标变换基础	454	23.3.3	数据库连接	515
20.1.2	实现缩放变换	454	23.3.4	创建数据库	516
20.1.3	实现平移变换	458	23.3.5	操作数据库	517
20.2	使用 Alpha 混合技术	460	23.3.6	数据显示	518
20.2.1	基本知识	460	23.4	具体编码	518
20.2.2	实现简单混合	461	23.4.1	设置服务信息	518
20.2.3	实现光晕和云层效果	465	23.4.2	播放器主界面	519
			23.4.3	播放列表功能	529
			23.4.4	菜单功能模块	531

23.4.5	播放设置界面	533	24.2.1	布局文件	542
23.4.6	设置显示歌词	535	24.2.2	程序文件	543
23.4.7	文件浏览器模块	536	24.3	闹钟列表模块	552
23.4.8	数据存储	539	24.3.1	设置主界面	552
第 24 章	开发一个闹钟系统	541	24.3.2	设置闹钟界面	557
24.1	项目介绍	541	24.3.3	闹钟提醒模块	564
24.1.1	系统需求分析	541	24.3.4	重复设置	569
24.1.2	构成模块	541	24.3.5	闹钟数据操作	570
24.2	系统主界面	542	24.4	选择铃声音乐	573



第一篇

基础技术篇

- 第 1 章 Android 技术概述
- 第 2 章 Android 技术核心框架分析
- 第 3 章 获取并分析 Android 源码

第1章 Android 技术概述

Android 是一种智能手机操作系统，是建立在 Linux 开源系统基础之上的，能够迅速建立手机软件的解决方案。虽然 Android 外形比较简单，但是其功能十分强大，已经成为当前软件行业的一股新兴力量。从 2011 年开始到现在，Android 一直占据全球智能手机市场占有率第一的宝座。在本章的内容中，将简单介绍 Android 的发展历程和背景，并介绍搭建 Android 应用开发环境的基本知识，为进入本书后面知识的学习打下基础。

1.1 智能手机系统介绍

在 Android 系统诞生之前，智能手机这个新鲜事物大大丰富了人们的生活，得到了广大手机用户的青睐，各大手机厂商纷纷建立了各种智能手机操作系统来抢占市场份额。Android 系统就是在这个风起云涌的历史背景下诞生的。

1.1.1 何谓智能手机

智能手机具有像个人电脑那样强大的功能，拥有独立的操作系统，允许用户自行安装应用软件、游戏等第三方服务商提供的程序，并且通过移动通信网络接入到互联网络中。在 Android 系统诞生之前已经有很多优秀的智能手机产品，例如家喻户晓的 Symbian 系列和微软的 Windows Mobile 系列等。

1.1.2 看当前主流的智能手机系统

在当今市面中最主流的智能手机系统当属微软的 Windows Mobile、Symbian、Palm、BlackBerry、iOS 和本书的主角 Android。

1. 微软的 Windows Mobile

Windows Mobile 是微软公司的一款接触产品，Windows Mobile 将熟悉的 Windows 桌面扩展到了个人设备中。使用 Windows Mobile 操作系统的设备主要有 PC 手机、PDA、随身音乐播放器等。Windows Mobile 操作系统有 3 种，分别是 Windows Mobile Standard、Windows Mobile Professional、Windows Mobile Classic。当前的最新版本是 Windows Phone 7 和 Windows Phone 8。

2. iOS

iOS 作为苹果移动设备 iPhone 和 iPad 的操作系统，在 App Store 的推动之下，成为了世界上引领潮流的操作系统之一。原本这个系统名为“iPhone OS”，2010 年 6 月 7 日在 WWDC 大会上宣布改名为“iOS”。iOS 的用户界面的概念基础是能够使用多点触控直接操作。控制方法包括滑动、轻触开关及按键。与系统交互包括滑动（Swiping）、轻按（Tapping）、挤压（Pinching，通常

用于缩小)及反向挤压 (Reverse Pinching or Unpinching, 通常用于放大)。此外通过其自带的加速器, 可以令其旋转设备改变其 y 轴以令屏幕改变方向, 这样的设计令 iPhone 更便于使用。

3. Android

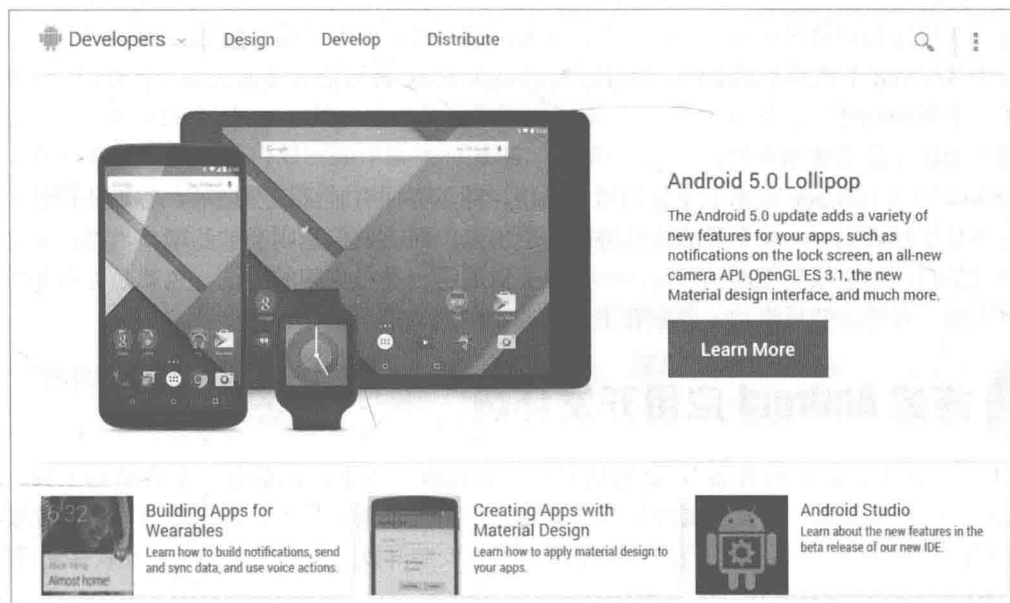
Android 是我们本书的主角, 是 2007 年 11 月 5 日宣布的基于 Linux 平台的开源手机操作系统, 该平台由操作系统、中间件、用户界面和应用软件组成, 号称是首个为移动终端打造的真正开放和完整的移动软件。

根据国际数据公司 (IDC) 公布的数据, 在 2013 年第一季度, Android 和 iOS 的装机量占有所有智能手机出货量的 92.3%。在 2013 年前 3 个月, 安装 Android 系统的新智能手机数量跃升至 1.621 亿部, 大大超过去年同期的 9 030 万部。这意味着, 在运往世界各地的所有新智能手机中, 谷歌的移动操作系统的市场占有率已经达到 75%, 比 2012 年第一季度的 59.1% 有显著提高。

到本书截稿之时, Android 的最新版本是 Android L。

1.1.3 Android 5.0 的突出变化

2014 年 10 月 15 日, 谷歌发布了下一代系统 Android 5.0, 并将在 26 日晚些时候提供给开发者下载, 如图 1-1 所示。



▲图 1-1 谷歌推出 Android 5.0

和以往版本相比, Android 5.0 版本的最突出特性如下所示。

(1) “Material” 主题。

Dave Burke 表示, 开发者在开发新应用时, 可选择一个被称为 “Material” 的主题。该主题支持新的动画效果、实时 3D 阴影显示以及其他多项新功能, 在 Demo 中, 他使用了拨号界面做介绍, 所有的操作都十分流畅。随后 Dave Burke 介绍了新的强化的通知中心, 通过下滑操作, 可以看到所有的通知。然后 Dave Burke 开始利用游戏介绍新的消息系统 heads up, 玩游戏时, 如果有电话拨打进来, 屏幕顶端会出现一个通知框。如果向左右滑动手指, 则可以忽略来电。这样的设计能尽量保证不打断用户的当前操作。

(2) 新 Android Wear 发布。

Android 工程部总监 David Singleton 登台介绍穿戴设备的相关开发。David Singleton 通过 LG G Watch 智能手表展示 Android Wear 系统，智能手表通过振动提醒穿戴者有消息、来电。用户可上下滑动屏幕来翻页通知内容。

完整的 Android Wear SDK 将会发布，其 API 与标准版 Android API 基本一致。开发人员移植应用不存在难度。Android Wear SDK 会自动同步通知到 Android。开发者可以开发语音回复和页面回复的应用程序。

(3) Android TV 发布。

Android TV 是一套可用于电视机顶盒的系统，有相应 SDK，从这里看出谷歌对它的重视程度不亚于智能手机和平板。Android TV 需要一个 D Pad 进行语音控制，其支持 HDMI 和接收器等视频信号输入。系统本身是覆盖在视频之上的，如搜索菜单、控制菜单等。Android TV 的核心优势是搜索（基于 Google Now）。用户可以用 Android Wear 智能手表设备来控制 Android TV。Android TV 支持谷歌 Cast 技术，也就是说用户可以通过这个系统把电视当作 ChromeCast 电视棒使用。谷歌 Play 也专门开辟了 Android TV 应用类别。

(4) Android Auto 系统发布：导航、通信和音乐成核心。

谷歌发布的 Android Auto 系统面向未来汽车市场。Android Auto 的核心将是导航、通信和音乐。当 Android 智能手机与 Android Auto 系统连接时，手机屏幕能投射到车载屏幕上。Android Auto 可以进行环境感知和语音控制，它的主界面跟 Google Now 并无二致。虽然 Android Auto 可以被看作基于 Android 系统的车载 GPS，但考虑到 Google Now 自然语言搜索的强大性能，“人车对话”达到了一个新的高度。

(5) 全新设计的通知系统。

Android 5.0 Lollipop 带来了全新的通知系统。除了界面有较大改变之外，谷歌还调整了通知中心的信息展示规则——最重要的信息将被显示出来，而次要信息则会被隐藏。当然，如果需要查看全部信息，则继续向下滑动即可——有些类似展示一叠扑克牌的手法，也就是你首先看到的是表面上的一张牌，然后滑动，隐藏在下方的扑克牌就会展示出来。

1.2 搭建 Android 应用开发环境

“工欲善其事，必先利其器”，意思是要想高效完成一件事，需要有一个合适的工具。对于 Android 开发人员来说，开发工具同样至关重要。作为一项新兴技术，在进行开发前首先要搭建一个对应的开发环境。而在搭建开发环境前，需要了解安装开发工具所需要的硬件和软件配置条件。

注意

Android 开发包括底层开发和应用开发。底层开发一般是指和硬件相关的开发，并且是基于 Linux 环境的，例如开发驱动程序。应用开发是指开发能在 Android 系统上运行的程序，如游戏、地图等程序。本书的重点是讲解多媒体应用开发，即使讲一些底层的知识，也是为上层的应用服务的。

因为开发 Android 应用程序最合适的系统是 Windows，所以本书只介绍在 Windows 下配置 Eclipse+ADT 的过程。

1.2.1 安装 Android SDK 的系统要求

在搭建之前，一定先确定安装 Android SDK 时对系统的要求，具体如表 1-1 所示。

表 1-1 安装 Android SDK 的系统要求

项 目	版本要求	说 明	备 注
操作系统	Windows XP 或 Vista; Mac OS X 10.4.8+Linux Ubuntu Drapper	根据自己的电脑自行选择	选择自己最熟悉的操作系统
软件开发包	Android SDK	选择最新版本的 SDK	截止到目前，最新手机版本是 5.0
IDE	Eclipse IDE+ADT	Eclipse 3.3 (Europa), 3.4 (Ganymede)ADT(Android Development Tools)开发插件	选择 “for Java Developer”
其他	JDKApache Ant	Java SE Development Kit 5 或 6 Linux 和 Mac 上使用 Apache Ant 1.6.5+, Windows 上使用 1.7+版本	单独的 JRE 是不可以的，必须要有 JDK，不兼容 Gnu Java 编译器 (gcj)

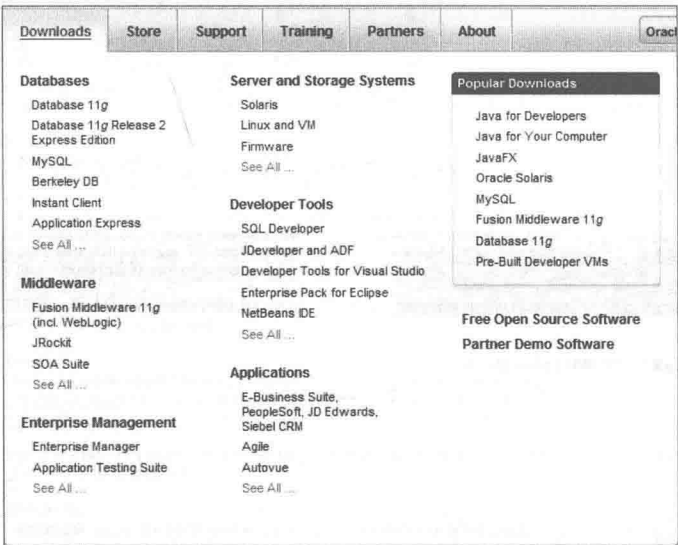
Android 工具是由多个开发包组成的，具体说明如下。

- JDK：可以到网址 <http://java.sun.com/javase/downloads/index.jsp> 下载。
- Eclipse（Europa）：可以到网址 <http://www.eclipse.org/downloads/> 下载 Eclipse IDE for Java Developers。
- Android SDK：可以到网址为 <http://developer.android.com> 的网站下载。
- 还有对应的开发插件。

1.2.2 安装 JDK

JDK（Java Development Kit）是整个 Java 的核心，包括了 Java 运行环境、Java 工具和 Java 基础的类库。JDK 是学好 Java 的第一步，是开发和运行 Java 环境的基础，当用户要对 Java 程序进行编译的时候，必须先获得对应操作系统的 JDK，否则将无法编译 Java 程序。在安装 JDK 之前需要先获得 JDK，获得 JDK 的操作流程如下所示。

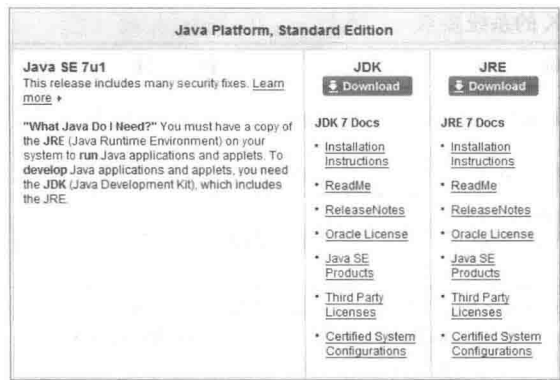
（1）登录 Oracle 官方网站，网址为 <http://developers.sun.com/downloads/>，如图 1-2 所示。



▲图 1-2 Oracle 官方下载页面

（2）在图 1-2 中可以看到有很多版本，例如在此选择 Java 7 版本，下载页面如图 1-3 所示。

（3）在图 1-2 中单击 JDK 下方的“Download”按钮，在弹出的新界面中选择将要下载的 JDK，笔者在此选择的是 Windows x86 版本。如图 1-4 所示。



▲图 1-3 JDK 下载页面



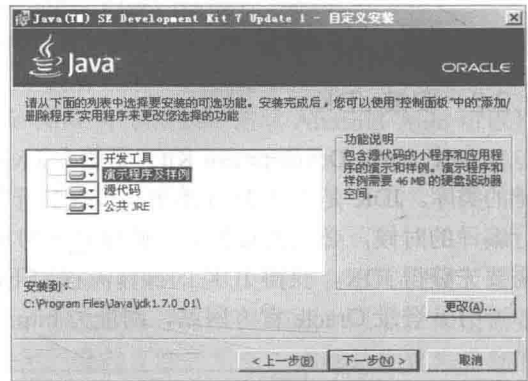
▲图 1-4 选择 Windows x86 版本

(4) 下载完成后双击下载的“.exe”文件进行安装，将弹出“安装向导”对话框，在此单击“下一步”按钮。如图 1-5 所示。

(5) 弹出“安装路径”对话框，在此选择文件的安装路径。如图 1-6 所示。



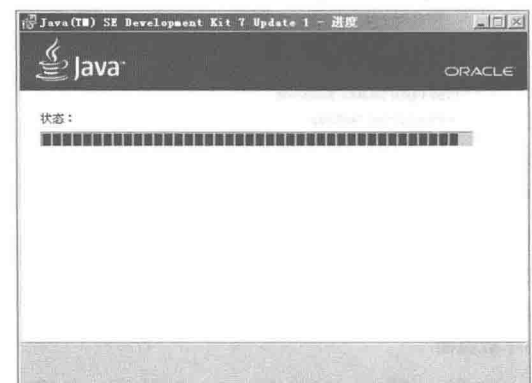
▲图 1-5 “安装向导”对话框



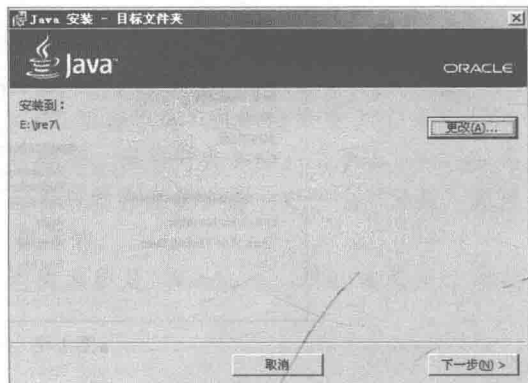
▲图 1-6 “安装路径”对话框

(6) 在此设置安装路径是“E:\jdk1.7.0_01\”，然后单击“下一步”按钮开始在安装路径解压压缩下载的文件。如图 1-7 所示。

(7) 完成后弹出“目标文件夹”对话框，在此选择要安装的位置。如图 1-8 所示。



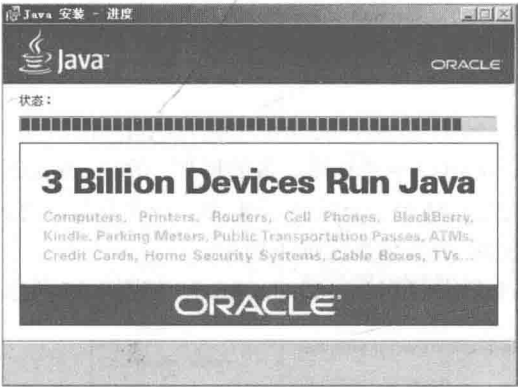
▲图 1-7 解压缩下载的文件



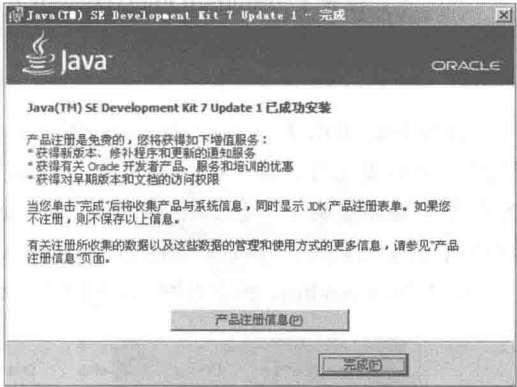
▲图 1-8 “目标文件夹”对话框

(8) 单击“下一步”按钮后开始正式安装，如图 1-9 所示。

(9) 完成后弹出“完成”对话框，单击“完成”按钮后完成整个安装过程。如图 1-10 所示。



▲图 1-9 正式安装

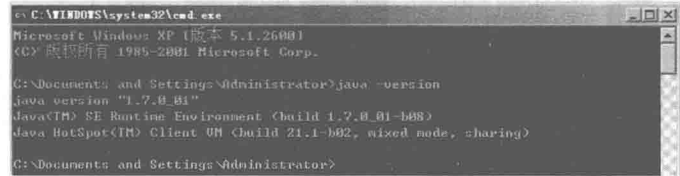


▲图 1-10 “完成”对话框

完成安装后可以检测是否安装成功，检测方法是依次单击【开始】|【运行】，在运行框中输入“cmd”并按下回车键，在打开的 CMD 窗口中输入 `java -version`，如果显示图 1-11 所示的提示信息，则说明安装成功。

如果检测没有安装成功，需要将其目录的绝对路径添加到系统的 PATH 中。具体做法如下所示。

(1) 右键依次单击【我的电脑】|【属性】|【高级】，单击下面的“环境变量”，在下面的“系统变量”处选择“新建”，在变量名处输入 `JAVA_HOME`，变量值中输入刚才的目录，比如设置为“`C:\Program Files\Java\jdk1.7.0_01`”。如图 1-12 所示。



▲图 1-11 CMD 窗口中检测是否安装成功



▲图 1-12 设置系统变量

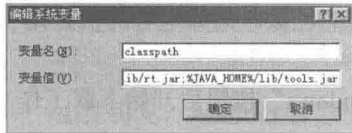
(2) 再次新建一个变量名为 `classpath`，其变量值如下所示。

```
.;%JAVA_HOME%\lib\rt.jar;%JAVA_HOME%\lib\tools.jar
```

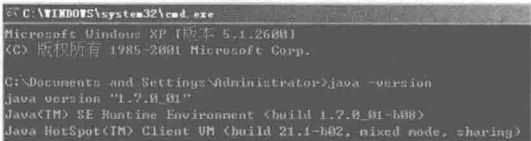
具体如图 1-13 所示。单击“确定”按钮找到 PATH 的变量，双击或单击编辑，在变量值最前面添加如下值。

```
%JAVA_HOME%\bin;
```

(3) 再依次单击【开始】|【运行】，在运行框中输入“cmd”并按下回车键，在打开的 CMD 窗口中输入 `java -version`，如果显示图 1-14 所示的提示信息，则说明安装成功。



▲图 1-13 设置系统变量



▲图 1-14 CMD 界面