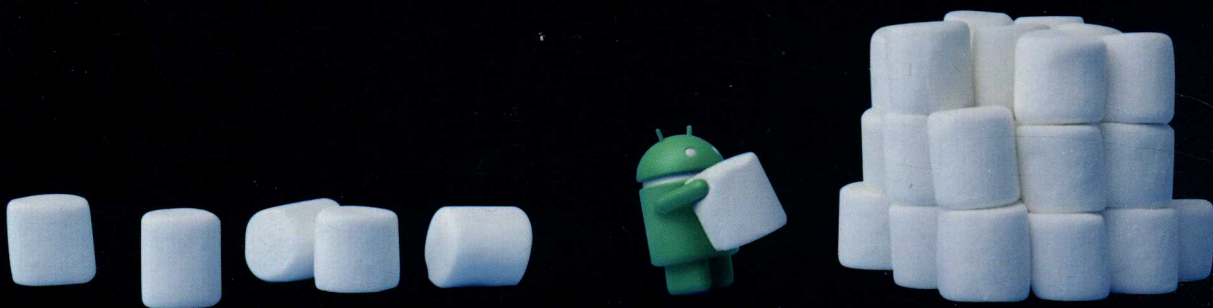


Android

卢海东 ◆ 编著

应用开发从入门到精通



基于 Android 6 系统

通过大量实例全面讲解了 Android 开发中的核心知识点：Android UI、常见控件和自定义控件、碎片、四大组件（Activity、Service、Broadcast 和 ContentProvider）、二维绘制、Canvas 和 Paint、多媒体开发等；

讲解了 Android 开发的高级技术：Camera、地图、GPS、传感器、录音、WiFi 设备等，OpenGL ES 的应用，着色和 3D 图形的绘制和纹理贴图，Android NDK 和 Android 网络编程，Android 进程间通信机制 Binder 原理等，最后通过一个完整项目的开发，帮助读者把所学的知识串起来，达到学以致用用的目的。



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



中国社会科学出版社

北京

图书在版编目 (C I P) 数据

Android应用开发从入门到精通 / 卢海东编著. --
北京 : 人民邮电出版社, 2016. 2
ISBN 978-7-115-41354-3

I. ①A… II. ①卢… 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(2016)第016927号

内 容 提 要

本书共分 17 章, 主要内容如下。讲解了 Android 系统架构, 让读者对 Android 系统有一个整体认识; 介绍了 Android 开发环境搭建和 Android 应用程序的基本结构; 通过第一个简单实例, 让读者对 Android 应用开发有一个全面的认知; 讲解了 Android UI 知识, 包括 Android 中常见控件和创建自定义控件; 介绍了 Android 中最重要的一个组件 Activity (活动), 并对碎片进行了全面讲解; 介绍了 Android 中 Service (服务) 组件及此组件的生命周期和跨进程通信; 介绍了 Android 的 Broadcast (广播接收器) 组件, 帮助读者掌握接收和发送广播的方式及工作原理; 讲解了 Android 中的本地数据存储技术; 介绍了 Android 的 ContentProvider (内容提供者) 组件; 介绍了 Android 提供的二维绘制特性, 包括使用 Canvas 和 Paint 在屏幕上进行图形和文本的绘制; 介绍了 Android 多媒体开发的知识; 讲解了 Android 的特色开发技术, 包括 Camera、地图、GPS、传感器、录音、WiFi 设备等; 讲解了在 Android 应用程序中使用 OpenGL ES 的几种不同的方法, 以及对 2D 物体的绘制、着色和 3D 图形的绘制和纹理贴图; 介绍了 Android NDK 和 Android 网络编程, 以及 Android 底层架构, 如 Android 进程间通信机制 Binder 原理, 并通过实例让读者能够真正体会 Native Service 的魅力; 本书最后通过讲解一个完整项目的开发, 帮助读者把所学的知识串起来, 达到学以致用目的, 并教会读者如何打包 App 和上架等。

本书的内容通俗易懂、由浅入深, 既适合初学者, 又适合 Android 程序员, 也可以作为大专院校相关专业师生和培训学校的教学用书。

-
- ◆ 编 著 卢海东
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>
北京昌平百善印刷厂印刷
 - ◆ 开本: 787×1092 1/16
印张: 26
字数: 701 千字
印数: 1—3 000 册
-

定价: 59.00 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

笔者已从事 Android 开发工作多年。笔者认为，写书不仅仅是将知识点进行梳理总结，更重要的是体现一种责任，要能系统地、循序渐进地将知识点讲清楚，让读者看明白。出于这个初衷，特意撰写了本书。本书是一本介绍 Android 应用开发的实用图书，全面介绍了 Android 6 平台上应用开发的知识，覆盖了 Android 开发中的大部分知识点；不仅有理论知识，还包括大量实例。

本书内容

本书共分 17 章，下面简单介绍一下每章的重要知识点。

第 1 章对 Android 系统架构进行解析，使读者对 Android 系统有一个整体的认识。

第 2 章介绍了如何搭建 Android 开发环境、Android 应用程序的基本结构以及 Android 的开发工具。

第 3 章通过一个简单实例，让读者对 Android 应用开发有一个全面的认知，同时讲解了 Intent 和 LogCat 的使用。

第 4 章主要是对 Android UI 方面的知识进行讲解，帮助读者学习 Android 中一些常见控件的用法，并且掌握创建自定义控件技术。

第 5 章介绍 Android 中最重要的一个组件——Activity（活动），不管你以前有没有接触过它，相信学完本章后，会对这个组件有一个全新的认识，并对碎片有一个全面的了解。碎片是自 Android 3.0 之后引入的全新概念，目前已广泛应用于 Android 手机和平板电脑的开发当中。同时读者会对 Android 多线程编程有一个全新的认识。

第 6 章介绍 Android 中另一个重要的组件——Service（服务），了解 Service 的生命周期，以及如何跨进程通信，同时学习 AIDL 的编写过程。

第 7 章介绍 Android 中第三个重要的组件——Broadcast（广播接收器），讲解了接收和发送广播的方式，并且掌握 Android 广播的工作原理。

第 8 章对 Android 中的本地数据存储技术进行了讲解，包括 SharedPreferences 存储、数据

库存储，以及文件存储。

第 9 章介绍 Android 中的第四个组件——ContentProvider（内容提供器）的应用知识。

第 10 章介绍 Android 提供的二维绘制特性，包括使用 Canvas 和 Paint 在屏幕上进行图形和文本的绘制，还讲解了如何在屏幕上显示动画的各种方法。

第 11 章介绍 Android 多媒体方面的知识，包括如何播放音乐、视频文件，以及 Android 系统提供的多媒体的统一处理机制。

第 12 章针对 Android 手机的特色开发进行讲解，包括 Camera、地图、GPS、传感器、录音、WiFi 设备等开发知识。

第 13 章讲解了在 Android 应用程序中使用 OpenGL ES 的几种不同的方法，以及如何在独立的线程中初始化 OpenGL ES，同时介绍了 OpenGL ES 对 2D 物体的绘制、着色，以及 3D 图形的绘制和纹理贴图。

第 14 章会带领读者进入一个全新的开发领域，Android 应用编程不仅局限于 Java 语言，在这一章中读者可以感受到用 Android NDK 编程带来的乐趣。

第 15 章介绍了 Android 网络编程知识，主要讲解了如何使用 HTTP 协议和服务器进行通信，以及 Android Socket 编程等。

第 16 章帮助读者深入了解 Android 的底层架构，通过对 Android 进程间通信机制 Binder 的原理性分析及实例解析，让读者能够真正体会 Native Service 的魅力。

第 17 章将带领读者一起编写一个完整的项目，并教会读者如何把程序打包、上架到谷歌市场等。通过整本书的学习，读者将有能力开发出一款不错的应用程序，并能对它进行经营和盈利。

本书各章的内容相对比较独立，读者除了可以循序渐进地学习之外，还可以把它当成一本参考手册，随时查阅。

读者对象

本书的内容通俗易懂、由浅入深，既适合初学者阅读，也适合专业人员学习。学习本书内容之前读者并不需要有任何的 Android 基础，但是需要有一定的 Java 和 C 基础。因为 Android 开发会使用到 Java 和 C 语言，但本书并不介绍 Java 和 C 语言方面的知识。

阅读本书时，读者可以根据自身的情况来决定如何阅读。如果是初学者，建议读者从第 1 章开始循序渐进地阅读，这样更容易理解。如果读者已经有了一定的 Android 基础，也可以选择感兴趣的章节跳跃式地阅读。本书读者答疑、视频和源程序下载服务 QQ 群：492323603。编辑联系邮箱 zhangtao@ptpress.com.cn。

致谢

在家人和朋友的帮助和支持下，这本书终于出版了。本书在编写过程中得到了人民邮电出版社编辑的很多帮助和建议，得到了我妻子的鼎力支持，在此表示感谢。

编者

目 录

第 1 章 揭开神秘面纱——Android 系统简介	1
1.1 认识 Android 系统	2
1.1.1 Android 成长历程	2
1.1.2 发行版本	3
1.1.3 得到大家的认可——Android 系统的市场份额	3
1.2 Android 架构解析	4
1.2.1 Android 系统架构图	4
1.2.2 应用程序 (Applications)	5
1.2.3 应用程序框架层 (Framework)	6
1.2.4 系统运行库 (Libraries)	7
1.2.5 Linux 内核	11
1.3 小结	11
第 2 章 工欲善其事, 必先利其器——Android 开发初识	12
2.1 开发准备工作	13
2.2 开发环境搭建	13
2.3 创建第一个 Android 项目	16
2.3.1 创建新项目	16
2.3.2 程序工程架构图	18
2.3.3 HelloWorld 程序架构解析	19
2.4 Android 开发工具	22
2.4.1 Android 开发工具 Adb	22
2.4.2 Android 开发工具 DDMS	23
2.5 小结	25
第 3 章 亲密接触——Android 入门程序开发	26
3.1 水果的营养成分分析器	27
3.2 手把手教你开发项目	27
3.2.1 手动创建 Activity	28
3.2.2 创建和加载布局	28
3.2.3 在 Activity 中使用 Toast	32
3.2.4 在 Activity 中加入菜单 (Menu)	32
3.2.5 在 Activity 中使用 Dialog	35
3.2.6 Spinner 控件的使用	35
3.2.7 界面切换	38

3.3	Intent 的使用	40
3.3.1	使用显式 Intent	40
3.3.2	使用隐式 Intent	41
3.3.3	隐式 Intent 的多种用法	42
3.3.4	向 Activity 传递数据	44
3.3.5	返回数据给 Activity	46
3.4	Android 日志	48
3.4.1	添加 LogCat 到 Eclipse	48
3.4.2	使用 Android 的日志工具 Log	48
3.5	小结	51
第 4 章	美丽由我掌控——UI 的开发	52
4.1	常用控件	53
4.1.1	EditText	53
4.1.2	TextSwitcher	55
4.1.3	ImageView	56
4.1.4	ImageSwitcher	57
4.1.5	ListView 的使用	58
4.1.6	提升 ListView 的运行效率	60
4.1.7	RecyclerView	63
4.1.8	ViewStub	69
4.2	Android 七种常见布局	72
4.2.1	LinearLayout	72
4.2.2	RelativeLayout	75
4.2.3	FrameLayout	77
4.2.4	TableLayout	78
4.2.5	GridLayout	79
4.2.6	AbsoluteLayout	81
4.2.7	CardView	82
4.3	自定义控件	86
4.3.1	加载布局	87
4.3.2	自定义控件	89
4.4	Android View 详解	90
4.5	Android Surface 详解	93
4.6	小结	98
第 5 章	组件之一——活动 (Activity)	99
5.1	活动的生命周期	100
5.2	活动的启动模式详解	105
5.2.1	standard 模式	106
5.2.2	singleTop 模式	107
5.2.3	singleTask 模式	109
5.2.4	singleInstance 模式	110
5.3	平板手机都兼顾——碎片 (Fragments)	112

5.3.1	概述	112
5.3.2	设计思想	112
5.3.3	Activity 添加 Fragment 界面	113
5.3.4	动态替换 Fragment	115
5.3.5	Fragment 生命周期	117
5.3.6	一个简单的阅读器	120
5.4	Android 中的多线程	127
5.4.1	Android Handler	129
5.4.2	使用 AsyncTask	131
5.5	小结	134
第 6 章	默默无闻的后台——服务 (Service)	135
6.1	Java 层服务	136
6.2	服务的生命周期	137
6.3	服务的启动	140
6.3.1	startService() 启动 Service	140
6.3.2	bindService() 启动 Service	144
6.4	Android AIDL	148
6.4.1	创建一个 AIDL 文件	150
6.4.2	实现接口	150
6.4.3	使用 parcelables 进行参数的值传递	152
6.4.4	客户端获取接口	153
6.4.5	IInterface 接口类	156
6.5	提高手机电池续航能力——JobScheduler 技术	157
6.6	小结	162
第 7 章	广播机制——Android 广播接收器	163
7.1	静态注册监听广播	164
7.2	动态注册接收 Broadcast 广播	166
7.3	本地广播	167
7.4	小结	169
第 8 章	数据仓库——Android 数据存储	170
8.1	轻量级存储类——SharedPreferences	171
8.1.1	使用 SharedPreferences 存取数据	171
8.1.2	存取复杂 (图像、音频) 类型的数据	173
8.2	PreferenceActivity 的使用	174
8.2.1	PreferenceActivity 实现	174
8.2.2	PreferenceFragement 实现	177
8.3	SQLite Databases	178
8.3.1	SQLite 在 Windows 系统中的使用	179
8.3.2	SQLite 基本操作	180
8.3.3	SQLite 事务	180
8.4	Android 中使用 SQLite 数据库	180
8.4.1	SQLite 常用的操作方法	181

8.4.2	SQLiteOpenHelper 类	182
8.4.3	在 SQLite 中执行查询	185
8.4.4	在 Activity 中显示和操作 SQLite 数据库	187
8.5	Android 文件的使用	189
8.5.1	将数据保存到文件	189
8.5.2	存取 SD 卡中的文件	190
8.5.3	读取 XML 文件	192
8.6	小结	194
第 9 章	跨程序共享数据——内容提供者 (ContentProvider)	195
9.1	获取用于通信录的电话及电话内容	197
9.1.1	从通信录中获取好友数据	197
9.1.2	通信录中添加好友	198
9.2	创建共享数据	199
9.2.1	创建共享数据	200
9.2.2	外部应用访问	204
9.3	小结	206
第 10 章	美图秀秀——开发 2D 绘图	207
10.1	屏幕绘图基础	208
10.1.1	Canvas 类	208
10.1.2	Paint 类	210
10.1.3	Color 类	212
10.1.4	Path 类	213
10.1.5	Drawable 类	216
10.2	使用文字	217
10.3	使用位图	219
10.4	Frame 动画	222
10.5	Tween 动画	224
10.6	小结	227
第 11 章	多媒体开发	228
11.1	音频播放	229
11.1.1	支持的音频格式	229
11.1.2	MediaPlayer 播放 MP3 文件	230
11.1.3	AudioTrack 播放原始音频文件	230
11.1.4	SoundPool 播放音频文件	231
11.2	视频播放	232
11.2.1	SurfaceView 播放视频	232
11.2.2	VideoView 视频播放器	234
11.3	多媒体存储	235
11.4	小结	237
第 12 章	Android 手机特色开发	238
12.1	使用手机 Camera	239
12.1.1	调用摄像头拍照	239

12.1.2	使用摄像头拍照	240
12.2	使用百度地图	242
12.2.1	申请 API Key	243
12.2.2	显示百度地图	245
12.2.3	标注覆盖物	247
12.3	定位服务	248
12.3.1	LocationManager 的基本用法	249
12.3.2	确定自己的位置	250
12.4	使用手机中的 Sensor	252
12.5	使用手机录音	255
12.5.1	AudioManager	256
12.5.2	音频处理	259
12.6	手机 Wifi 的使用	261
12.7	小结	267
第 13 章	三维游戏的基础——OpenGL ES 3.0	268
13.1	OpenGL 常识普及	269
13.2	GLSurfaceView 的使用	272
13.3	绘图概念	274
13.4	平面图形	277
13.5	三维图形	279
13.6	平面纹理	286
13.7	三维纹理	289
13.8	小结	294
第 14 章	C++的魅力——Android 本地编程	295
14.1	Android NDK 简介	296
14.2	使用命令行编译 Native 部分	296
14.3	C++ (HelloWorld) 项目	297
14.4	Eclipse 编译 C++ (HelloWorld) 项目	299
14.5	HelloJNI 实例解析	301
14.6	Android.mk 文件语法规范	304
14.7	Application.mk 文件语法规范	308
14.8	Android JNI 简介	309
14.9	Android Native Activity	312
14.9.1	NativeActivity	313
14.9.2	Android Native 生命周期	314
14.10	使用 C/C++编写 OpenGL ES 程序入门	315
14.10.1	初始化 OpenGL	316
14.10.2	创建自己的 Renderer	316
14.11	小结	318
第 15 章	Android 网络编程	319
15.1	WebView 控件用法	320
15.2	使用 HTTP 协议访问资源	321

15.2.1	URLConnection 的使用	321
15.2.2	HttpClient 的使用	324
15.3	Android Socket	325
15.3.1	Android TCP 通信	326
15.3.2	Android UDP 通信	330
15.3.3	Native TCP/UDP 通信	334
15.4	小结	338
第 16 章 Android 的灵魂——底层服务		339
16.1	Android 进程间通信	340
16.2	Android IPC binders	344
16.2.1	Binders 概述	344
16.2.2	面向对象的 Binder IPC	345
16.2.3	Binder 通信模型	345
16.3	Binder 的 Native 实现 (libbinder)	347
16.4	Android Native Service	349
16.4.1	Native Service 接口介绍	351
16.4.2	简单的 Native Service 项目	354
16.5	Android Parcel	364
16.6	小结	367
第 17 章 综合实例解析——音乐播放器		368
17.1	音乐播放器	369
17.2	设计原则	369
17.3	框架分析	370
17.4	Application	371
17.5	程序主界面	372
17.6	音乐播放界面	374
17.7	Music Service 的编写	385
17.8	FavouriteActivity 的编写	394
17.9	MusicOnlineActivity 的编写	394
17.10	发布 App	397
17.10.1	生成正是签名的 APK 文件	397
17.10.2	申请 Google Play 账号	399
17.10.3	上传和发布应用程序	401
17.11	小结	404

第1章

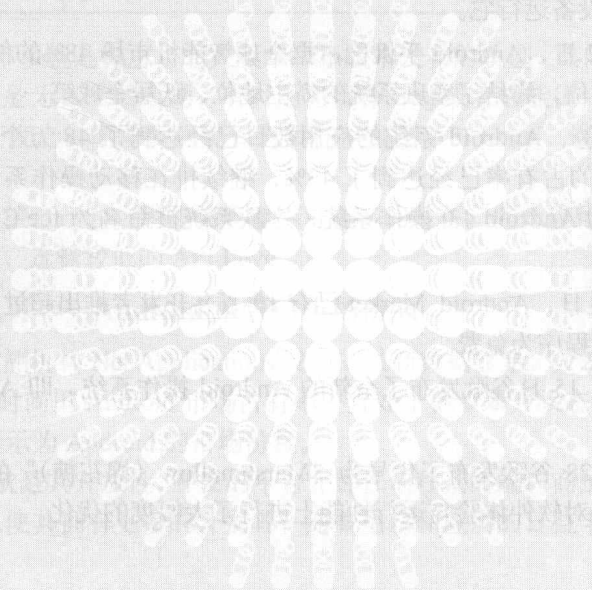
揭开神秘面纱——Android 系统简介

从不浪费时间的人，没有工夫抱怨时间不够。——杰弗逊

Android 系统已经成为全球应用最广泛的手机操作系统，三星、摩托等手机厂商早已经通过 Android 取得巨大成功。目前，国内对 Android 开发人才的需求在迅速增长，智能手机已经给各个行业带来了巨大的变革。因此，Android 手机应用软件必将在未来 IT 行业中占有举足轻重的地位。从趋势上看，Android 软件人才的需求会越来越大。

本书所介绍的平台是 Android 6.0，Android 6.0 是基于 Android 5.0 的升级版本，Android 5.0 是 Android 历史上改动最大的一个版本，新增和修改的 API 多达 5000 个。Android 5.0 重新设计了 UI 风格、重新设计了运行时库（ART runtime）架构并且引入了 Volte 项目，在 GPU 的性能方面也做出了巨大改进，所以，Android 5.0 从用户界面到底层代码结构都有很大的改进。Android 6.0 版本相对于 Android 5.0 在系统层面改动不大，只是新增了应用的权限管理和设置，支持 USB 接 MIDI 口，以及 Doze 模式，用来提升手机续航和全局快速搜索功能，Android 6.0 代号 Marshmallow，即棉花糖，本书将全面介绍 Android 6.0 平台的软件开发。

本章首先让读者了解 Android 系统的概念，详细介绍 Android 系统的历史背景，以及 Android 系统的架构组成，让读者对 Android 系统有一个初步的认识。



1.1 认识 Android 系统

Android 一词的本义指“机器人”，同时也是 Google 于 2007 年 11 月 5 日宣布的基于 Linux 平台的开源手机操作系统的名称，它由 Google 主导的开放手持设备联盟（Open Handset Alliance）进行维护，该组织的口号是“助力移动设备的变革，以更低廉的价格，为用户提供更强大的功能和更好的体验”。该平台由操作系统、中间件、用户界面和应用软件组成。

Android 系统最初由安迪·鲁宾（Andy Rubin）开发，开发这个系统的目的是利用其创建一个“智能手机”生态圈。Android 操作系统于 2005 年 8 月被美国科技企业 Google 收购。2007 年 11 月，Google 与 84 家硬件制造商、软件开发商及电信运营商成立开放手持设备联盟来共同研发改良 Android 系统，随后，Google 以 Apache 免费开放源代码许可证的授权方式，发布了 Android 的源代码。让生产商推出搭载 Android 的智能手机，Android 操作系统后来更逐渐拓展到平板电脑及其他领域上。

现在，Android 正主导着移动领域的一场革命。这是第一个真正与硬件无关的开放系统，使得同一个应用能够轻松兼容于大量不同的设备之上，为用户和开发者营造一个更富生机的生态圈。

1.1.1 Android 成长历程

了解 Android 就需要从它的历史开始，如下是过去几年的大事记。

- ❑ 2005 年 Google 收购 Android, Inc., 被认为是 Google 进军智能手机市场的开始。
- ❑ 2007 年开放手持设备联盟宣布成立，Android 正式踏入开源阵营。
- ❑ 2008 年 Android SDK 1.0 发布。G1 手机随之而来，它由 HTC 生产，并由无线运营商 T-Mobile USA 发售。
- ❑ 2009 年基于 Android 的设备如雨后春笋般出现。Android 系统发布了几个新版本，有 Cupcake(1.5)、Donut(1.6)、Eclair(2.0 及 2.1)。已有超过 20 款设备运行 Android。
- ❑ 2010 年 Android 成为仅次于黑莓的第二大智能手机平台。Froyo(Android 2.2)发布，已有超过 60 款设备运行它。
- ❑ 2011 年 8 月 2 日，Android 手机已占据全球智能机市场 48% 的份额，并在亚太地区市场占据统治地位，终结了塞班系统的霸主地位，跃居全球第一。
- ❑ 2011 年 9 月份，Android 系统的应用数目已经达到了 48 万个，在智能手机市场，Android 系统的占有率已经达到了 43%，继续排在移动操作系统首位。这时 Google 发布了全新的 Android 4.0 操作系统，这款系统被命名为 Ice Cream Sandwich（冰激凌三明治）。
- ❑ 2012 年 1 月 6 日，Android Market 已有 10 万名开发者推出超过 40 万个活跃的应用，大多数的应用程序为免费。
- ❑ 2014 年 10 月 15 日谷歌发布了全新的 Android 操作系统，即 Android 5.0。界面风格全新。
- ❑ 2015 年 9 月 28 谷歌发布了代号为“Marshmallow（棉花糖）”的 Android 6.0 系统，Android 6.0 在对软件体验与运行性能上进行了大幅度的优化。

1.1.2 发行版本

随着 Android 的改善，在不同的时间上推出了不同的版本。随着版本的更新，Android 功能和性能也在不断得到改进，下面我们来看看截止到本书写作的 Android 版本号，如表 1.1 所示。

Android 版本		
Android version	API level	Nickname
Android 1.0	1	
Android 1.1	2	
Android 1.5	3	Cupcake
Android 1.6	4	Donut
Android 2.0	5	Eclair
Android 2.0.1	6	Eclair
Android 2.1	7	Eclair
Android 2.2	8	Froyo (frozen yogurt)
Android 2.3	9	Gingerbread
Android 2.3.3	10	Gingerbread
Android 3.0	11	Honeycomb
Android 3.1	12	Honeycomb
Android 3.2	13	Honeycomb
Android 4.0-4.0.2	14	Ice Cream Sandwich
Android 4.0.3-4.0.4	15	Ice Cream Sandwich
Android 4.1	16	Jelly Bean
Android 4.2	17	Jelly Bean
Android 4.3	18	Jelly Bean
Android 4.4.2	19	KitKat
Android 4.4W	20	KitKat
Android 5.0	21	Lollipop
Android 5.1.1	22	Lollipop
Android 6.0	23	Marshmallow

Android 版本号差不多就是标志了它大版本和小版本发布的轨迹，其间以 API Level 最为重要。版本号的更新有时是为 API 的改动，有时是为修补 Bug 或者性能优化。

身为应用程序的开发者，你需要明确知道目标平台的 API Level，这决定了你的应用能否兼容。

一般来说，你会希望自己的应用能够兼容尽可能多的设备，因此应留意 Android 的版本在诸多现存设备上的分歧，选择较低的 API Level。

1.1.3 得到大家的认可——Android 系统的市场份额

著名互联网流量监测机构 Net Applications 发布的最新数据显示，从 2013 年 9 月到 2014 年 7 月，在这将近一年的时间里 Android 市场占有率一直处于稳步攀升状态，从最初的 29.42% 攀升至 44.62%。图 1.1 所示为 Android 的市场份额。

Android 平台首先就是其开放性，开发的平台允许任何移动终端厂商加入到 Android 联盟中来。显著的开放性可以使其拥有更多的开发者，随着用户和应用的日益丰富，Android 平台已经走向成熟。

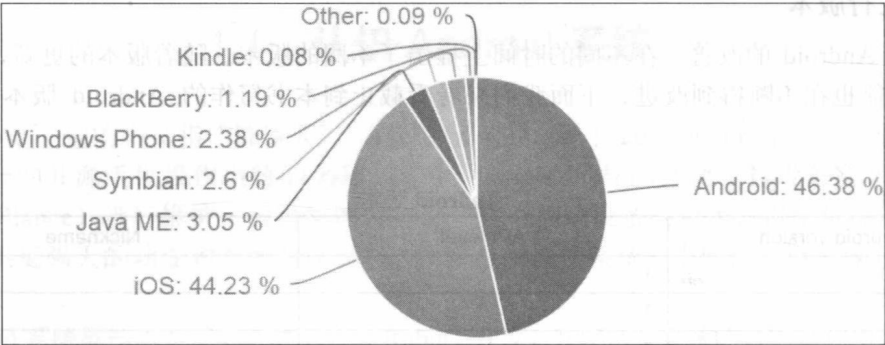


图 1.1 Android 的市场份额

1.2 Android 架构解析

Android 系统的底层建立在 Linux 系统之上,该平台采用一种称为软件叠层(Software Stack)的方式进行构建。这种软件叠层结构使得层与层之间相互分离,明确各层的分工。这种分工是软件工程中常说的低耦合高内聚的设计概念。

1.2.1 Android 系统架构图

Android 作为一个移动设备的平台,其软件层次结构包括了内核层、中间件和应用程序。下面看看 Android 的系统架构图,如图 1.2 所示。

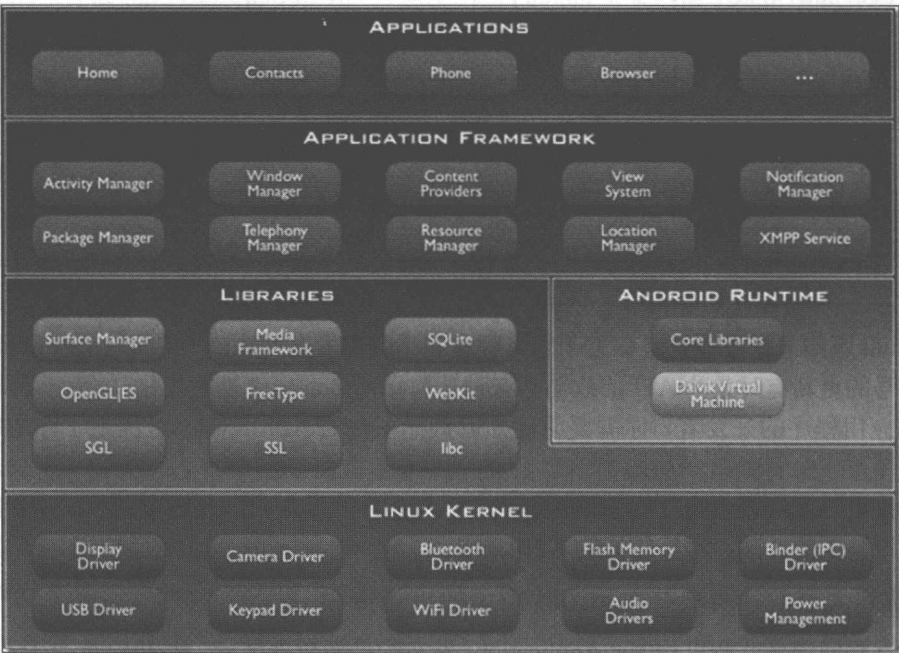


图 1.2 Android 架构图

如图 1.2 所示,Android 系统采用分层的架构。从架构看,Android 分为 4 层,从高层到底层分别是应用程序层、应用程序框架层、系统运行库层和 Linux 核心层。

下面对 Android 框架图进行庖丁解牛式的分析,同样从高层到底层,首先是应用程序。

1.2.2 应用程序 (Applications)

Android 会同一系列核心应用程序一起发布,包括 E-mail 客户端、SMS 短消息程序、日历、地图、浏览器、联系人管理程序等。所有的应用程序都是使用 Java 语言编写的。通过打包工具将应用程序的字节码文件、资源文件和清单文件打包到一个以“apk”为后缀的文件中,该文件就是应用程序的载体,被复制到 Android 设备上之后,会被解析成机器能识别的机器码。那么开发应用是不是有设计规则呢?回答是肯定的,请看下面 3 点。

(1) 每个应用程序都运行在自己的进程空间中。在需要执行该应用程序时,Android 将启动该进程;当不再需要该应用程序,并且系统资源不够分配时,系统将终止该进程。

(2) 每个进程都有自己的运行库 (ART), 所以,任何应用程序的代码与其他的应用程序的代码是相互隔离的。

(3) 每个应用程序被分配给一个唯一的 ID。所以,应用程序的文件只能对该应用程序可见。

Android 平台的一个核心要点是,一个应用程序能够利用其他应用程序的组件。例如,一个程序 A 用于显示指定文件夹中的全部数据库名,而另外一个程序 B 用于查看某一指定数据库的信息,如数据表名、数据表模式 (Schema)、数据表内容等。在程序 A 中,当用户单击列表中的某一数据库名称项时,可以调用程序 B 中的模块去显示指定数据库的信息,而无需重复开发。

而为了做到这一点,系统必须能够在应用程序需要调用指定模块时找到并启动包含该模块的组件。因此,不像其他系统的应用程序,Android 应用程序没有 main 方法,代码框架也必须遵照 Android 平台所定义的形式。所以,Android 应用程序需要包含系统能够识别并调用的一些基本组件。Android 系统四大组件分别是 Activity (活动)、Service (服务)、Broadcast Receiver (广播接收器) 和 Content Provider (内容提供者)。

1. Activity (活动)

Activity 表现为用于与用户进行交互的可视化界面,是所有 Android 应用程序的门面,凡是在应用中看得到的东西,都是放在 Activity 中的。例如,一个 Activity 可以是系统登录界面,另外一个 Activity 可以是显示已登录用户信息的列表。

用户定义的每一个 Activity 都继承于父类 Activity,代码如下所示。

```
public class MainActivity extends Activity {}
```

一个应用程序可能由一个或多个 Activity 组成,Android 平台通过 Activity 栈来对 Activity 进行管理。

2. Service (服务)

服务是一类无需可视用户界面,更适合在后台长期运行的应用程序,如背景音乐播放器或后台数据处理服务等。

同 Activity 一样,用户定义的每一个服务都继承于父类 Service,该父类由 Android 平台框架预定义。代码如下所示。

```
public class TouchService extends Service {}
```

3. Broadcast Receiver (广播接收器)

广播接收器是一类只接收和处理广播消息的组件。广播接收器也没有显示用户界面,但是,可以在响应其接收信息时启动一个 Activity,或者通过通知管理器来显示提示界面从而警示用户。

一个应用程序可能有任意数量的广播接收器来响应任何它认为重要的通告。所有用户定义的广播接收器都继承于父类 BroadcastReceiver。该父类由 Android 平台框架预定义。

4. Content Provider (内容提供者)

内容提供者可以将指定的一组应用程序数据让其他应用程序使用。这些数据可以存储于文