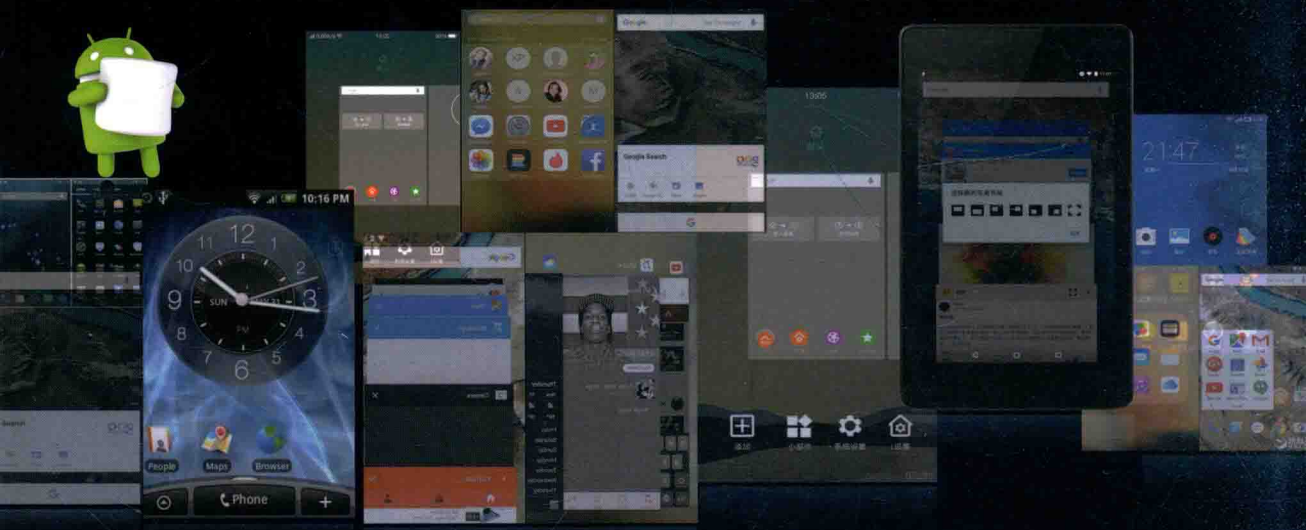


Android

开发完全讲义 (第三版)

李宁 著

- 资深一线Android开发工程师、极客，51CTO、cnblogs、CSDN超人气博主“银河使者”快意分享
- 本书前两版版权均输出到台湾，深受两岸数万Android学习者好评，被称为“Android第1书”
- 一本真正全面又系统的Android开发技术学习讲义，轻松不乏味
- 完美的实战环境搭建，Step by Step的案例讲解，真正消灭常见学习暗障
- 赠送全国首个技术测试APP金牌账号（账号获得方式：见geekculture技术支持微信公众号）



中国水利水电出版社
www.waterpub.com.cn

Android 开发完全讲义

（第三版）

李宁 著



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

Android 经典专著升级版, 全面介绍 Android 的应用开发技术。主要内容包括 Android 入门, 第一个 Android 程序, Android 应用程序架构, 建立用户接口, 控件详解, View 事件分发机制, 移动存储解决方案, App 之间的通信, 服务 (Service), 网络技术, 多媒体技术, Fragment, ActionBar, Android 5.x 新特性: 质感主题, Android 5.x 新特性: 阴影和视图裁剪, Android 5.x 新特性: 列表和卡片控件, Android 5.x 新特性: Drawable 资源, 其他 Android 5.X 新特性, 2D 动画, OpenGL ES 编程, 媒体特效 API, 资源、国际化与自适应, 访问 Android 手机的硬件, NDK 技术, 蓝牙技术, 有趣的 Android 技术, Android App 性能调优, 内存泄露检测, 项目实战: 超级手电筒, 项目实战: 基于 XMPP 的 IM 客户端。

本书适合: 有一定的 Java 基础, 想通过 Android 进入移动开发领域的读者; 已经有一定的 Android 开发经验, 想进一步提高 Android 开发能力的读者; 想将本书作为 Android 的参考手册, 随时随地查阅的读者; 对 Android 抱有浓厚兴趣的其他手机平台的开发人员; 正在学习 Android 的在校大学生以及培训学校的学员等使用。

图书在版编目 (C I P) 数据

Android开发完全讲义 / 李宁著. — 3版. — 北京: 中国水利水电出版社, 2015. 10
ISBN 978-7-5170-3663-0

I. ①A… II. ①李… 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223382号

策划编辑: 周春元 责任编辑: 张玉玲 加工编辑: 孙 丹 封面设计: 李 佳

书 名	Android 开发完全讲义 (第三版)
作 者	李宁 著
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658 (发行部)、82562819 (万水) 北京科水图书销售中心 (零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京泽宇印刷有限公司
规 格	184mm×240mm 16 开本 38 印张 975 千字
版 次	2010 年 6 月第 1 版 2010 年 6 月第 1 次印刷 2015 年 10 月第 3 版 2015 年 10 月第 1 次印刷
印 数	0001—4000 册
定 价	88.00 元

凡购买我社图书, 如有缺页、倒页、脱页的, 本社发行部负责调换

版权所有·侵权必究

第三版前言

从 2007 年的第一版 Android 系统，到现在为止（2015 年），已经整整 8 年时光了，正进入稳步发展的时期。Android 的版本也从 1.x、2.x、3.x、4.x 升级到现在的 5.x。在最新版的 Android 系统中，变化还是大大地。除了底层从 Dalvik 变到了 ART，SDK 层也发生了翻天覆地的变化。除了增加了数千 API，还引入了质感设计等新的 UI 风格。

由于现在关于 Android 5.x 的书仍然比较匮乏，所以在第二版的基础上，增加了 Android 5.x 的新特性部分。尤其是质感设计部分，读者可以通过这一部分内容，充分了解 Android 5.x 的主要变化。由于 Android 5.x 引入了 ART 技术，所以 App 的启动和运行效率明显提高，因此，未来的 Android 将会彻底摆脱 App 体验没有 iOS App 好的帽子。

本书是一本全面介绍 Android 各种常用开发技术的专著。全面介绍了 Android 5.x 的各种开发技术。

1. 内容丰富，知识面广

本书全面介绍了 Android 的各种应用开发技术，主要包括四大应用程序组件、UI 控件、Fragment、ActionBar、数据存储、网络、蓝牙、多媒体、NDK、GPS 等。读者通过本书可以深入了解 Android 应用开发技术的细节。

2. 深入讲解了 Android 5.x 带来的新技术

本书利用多章篇幅，详细介绍了 Android 5.x 给我们带来的新特性。例如，质感主题、新的控件、矢量动画等技术。通过这些内容，读者一定会喜欢上 Android 5.x 的。因为它不仅给我们带来了新的 API，还带来了惊喜。

3. 内容详实，深入浅出

本书绝大多数章节都配有大量的习题，采用了先理论后实战的方式进行讲解。让读者在理论和代码编写上都游刃有余。

4. 精彩综合实战案例

为了让读者可以综合运用本书知识，最后两章提供了两个综合案例：超级手电筒和基于 XMPP 的 IM 客户端。通过这两个综合案例，可以进一步消化本书的知识和技巧。

源代码下载：

本书源代码可以通过微信公众号（geekculture）下载，二维码如右图；或从中国水利水电出版社网站或万水书苑上免费

下载，网址为：<http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。



目 录

第三版前言

第 1 章 Android 入门	1	1.6 小结	18
1.1 Android 的基本概念	1	第 2 章 第一个 Android 程序	19
1.1.1 Android 简介	2	2.1 编写用于显示当前日期和时间的程序	19
1.1.2 Android 的系统构架	3	2.1.1 新建一个 Android 工程	19
1.2 Android 开发环境的搭建	4	2.1.2 界面控件的布局	20
1.2.1 开发 Android 程序需要些什么	4	2.1.3 编写实际代码	20
1.2.2 安装 Android SDK	5	2.2 调试程序	22
1.2.3 安装 Eclipse 插件 ADT	7	2.3 签名和发布应用程序	23
1.2.4 创建 AVD	8	2.3.1 使用命令行方式进行签名	23
1.2.5 启动 Android 模拟器	10	2.3.2 使用 ADT 插件方式进行签名	24
1.2.6 让 Android 模拟器飞 (X86 加速)	11	2.4 DDMS 透视图	25
1.2.7 测试 Android 开发环境是否 安装成功	12	2.5 小结	26
1.3 Android SDK 中的常用命令行工具	14	第 3 章 Android 应用程序架构	27
1.3.1 启动和关闭 ADB 服务 (adb start-server 和 adb kill-server)	14	3.1 Android 应用程序中的资源	27
1.3.2 查询当前模拟器/设备的实例 (adb devices)	14	3.1.1 资源存放在哪里	27
1.3.3 安装、卸载和运行程序 (adb install、 adb uninstall 和 am)	14	3.1.2 资源的种类	28
1.3.4 PC 与模拟器或真机交换文件 (adb pull 和 adb push)	15	3.1.3 资源的基本使用方法	28
1.3.5 Shell 命令	15	3.2 Android 的应用程序组件	29
1.3.6 创建、删除和浏览 AVD 设备 (android)	16	3.2.1 活动 (Activity) 组件	29
1.3.7 获取 Android 版本对应的 ID	17	3.2.2 服务 (Service) 组件	30
1.3.8 创建 SD 卡	17	3.2.3 广播接收者 (Broadcast receivers) 组件	30
1.4 Android 的学习资源	17	3.2.4 内容提供者 (Content providers) 组件	31
1.5 Google Play	18	3.3 AndroidManifest.xml 文件的结构	31
		3.4 小结	32
		第 4 章 建立用户接口	33
		4.1 建立、配置和使用 Activity	33
		4.1.1 建立和配置 Activity	33

4.1.2	Activity 的生命周期	35	4.4.1	用 Toast 显示提示信息框	77
4.1.3	Activity 生命周期的演示	37	4.4.2	Notification 与状态栏信息	78
4.2	视图 (View)	40	4.5	布局	81
4.2.1	视图简介	40	4.5.1	框架布局 (FrameLayout)	81
4.2.2	使用 XML 布局文件控制视图	40	4.5.2	霓虹灯效果的 TextView	81
4.2.3	在代码中控制视图	42	4.5.3	线性布局 (LinearLayout)	83
4.2.4	混合使用 XML 布局文件和代码 来控制视图	43	4.5.4	利用 LinearLayout 将按钮放在屏幕 的四角和中心位置	85
4.2.5	定制控件 (Widget) 的三种方式	45	4.5.5	相对布局 (RelativeLayout)	86
4.2.6	定制控件——带图像的 TextView	46	4.5.6	利用 RelativeLayout 实现梅花效果 的布局	86
4.2.7	定制控件——带文本标签的 EditText	49	4.5.7	表格布局 (TableLayout)	87
4.2.8	定制控件——可更换表盘的 指针时钟	52	4.5.8	计算器按钮的布局	88
4.3	使用 AlertDialog 类创建对话框	57	4.5.9	绝对布局 (AbsoluteLayout)	88
4.3.1	AlertDialog 类简介	57	4.5.10	查看 apk 文件中的布局	89
4.3.2	“确认/取消”对话框	58	4.6	小结	89
4.3.3	创建询问是否删除文件的 “确认/取消”对话框	58	第 5 章	控件详解	90
4.3.4	带 3 个按钮的对话框	60	5.1	显示和编辑文本的控件	91
4.3.5	创建“覆盖/忽略/取消”对话框	60	5.1.1	显示文本的控件: TextView	91
4.3.6	简单列表对话框	61	5.1.2	在 TextView 中显示 URL 及不同字体 大小、不同颜色的文本	93
4.3.7	单选列表对话框	61	5.1.3	带边框的 TextView	94
4.3.8	多选列表对话框	62	5.1.4	设置 TextView 控件的行间距	97
4.3.9	创建 3 种选择省份的列表对话框	62	5.1.5	输入文本的控件: EditText	98
4.3.10	水平进度对话框和圆形进度 对话框	67	5.1.6	在 EditText 中输入特定的字符	99
4.3.11	水平进度对话框和圆形进度 对话框演示	68	5.1.7	按 Enter 键显示 EditText	100
4.3.12	自定义对话框	71	5.1.8	自动完成输入内容的控件: AutoCompleteTextView	101
4.3.13	创建登录对话框	71	5.2	按钮与复选框控件	102
4.3.14	用 Activity 托管对话框	73	5.2.1	普通按钮控件: Button	103
4.3.15	创建悬浮对话框和触摸任何位置 都可以关闭的对话框	74	5.2.2	异形 (圆形、五角星、螺旋形和 箭头) 按钮	103
4.4	Toast 和 Notification	77	5.2.3	图像按钮控件: ImageButton	105
			5.2.4	同时显示图像和文字的按钮	105
			5.2.5	选项按钮控件: RadioButton	106

5.2.6	开关状态按钮控件: ToggleButton	107	5.5.13	垂直滚动视图控件: ScrollView	137
5.2.7	复选框控件: CheckBox	108	5.5.14	水平滚动视图控件: HorizontalScrollView	138
5.2.8	利用 XML 布局文件动态创建 CheckBox	108	5.5.15	可垂直和水平滚动的视图	138
5.3	日期与时间控件	110	5.5.16	网格视图控件: GridView	139
5.3.1	输入日期的控件: DatePicker	110	5.5.17	可循环显示和切换图像的控件: Gallery 和 ImageSwitcher	140
5.3.2	输入时间的控件: TimePicker	110	5.6	小结	142
5.3.3	DatePicker、TimePicker 与 TextView 同步显示日期和时间	111	第 6 章	View 事件分发机制	143
5.3.4	显示时钟的控件: AnalogClock 和 DigitalClock	112	6.1	事件分发的始作俑者	143
5.4	进度条控件	112	6.2	View 类中的事件分发引擎	144
5.4.1	进度条控件: ProgressBar	113	6.3	ViewGroup 类的事件分发引擎	146
5.4.2	拖动条控件: SeekBar	114	6.4	通过代码验证 View 事件分发机制	148
5.4.3	改变 ProgressBar 和 SeekBar 的颜色	115	6.4.1	实现一个派生自 Button 的类	148
5.4.4	评分控件: RatingBar	116	6.4.2	实现布局	149
5.5	其他重要控件	118	6.4.3	实现主窗口类	149
5.5.1	显示图像的控件: ImageView	118	6.5	单击事件 (onClick) 是如何被触发的	152
5.5.2	可显示图像指定区域的 ImageView 控件	119	6.6	Activity 中的 dispatchTouchEvent 方法	154
5.5.3	动态缩放和旋转图像	119	6.7	小结	157
5.5.4	列表控件: ListView	121	第 7 章	移动存储解决方案	158
5.5.5	可以单选和多选的 ListView	123	7.1	最简单的数据存储方式: SharedPreferences	158
5.5.6	动态添加、删除 ListView 列表项	125	7.1.1	使用 SharedPreferences 存取数据	158
5.5.7	改变 ListView 列表项选中状态 的背景颜色	128	7.1.2	数据的存储位置和格式	160
5.5.8	封装 ListView 的 Activity: ListActivity	129	7.1.3	存取复杂类型的数据	161
5.5.9	使用 SimpleAdapter 建立复杂的 列表项	130	7.1.4	设置数据文件的访问权限	163
5.5.10	给应用程序评分	131	7.1.5	可以保存设置的 Activity: PreferenceActivity	165
5.5.11	可展开的列表控件: ExpandableListView	133	7.2	文件的存储	168
5.5.12	下拉列表控件: Spinner	136	7.2.1	openFileOutput 和 openFileInput 方法	168
			7.2.2	SD 卡文件浏览器	169
			7.2.3	存取 SD 卡中的图像	173
			7.2.4	SAX 引擎读取 XML 文件的原理	175
			7.2.5	将 XML 数据转换成 Java 对象	175

7.3 SQLite 数据库	178	8.2.2 开机可自动运行的程序	208
7.3.1 SQLite 数据库管理工具	178	8.2.3 收到短信了, 该做什么	209
7.3.2 创建数据库和表	179	8.2.4 显示手机电池的当前电量	211
7.3.3 模糊查询	181	8.2.5 在自己的应用程序中发送广播	212
7.3.4 分页显示记录	181	8.2.6 接收联系人系统中发送的添加 联系人广播	213
7.3.5 事务	181	8.3 小结	214
7.4 在 Android 中使用 SQLite 数据库	182	第 9 章 服务 (Service)	215
7.4.1 SQLiteOpenHelper 类与自动 升级数据库	182	9.1 Service 起步	215
7.4.2 SimpleCursorAdapter 类与 数据绑定	183	9.1.1 Service 的生命周期	215
7.4.3 带照片的联系人管理系统	185	9.1.2 绑定 Activity 和 Service	218
7.4.4 将数据库与应用程序一起发布	188	9.2 系统服务	220
7.4.5 英文词典	189	9.2.1 获得系统服务	220
7.5 持久化数据库引擎 (db4o)	192	9.2.2 监听手机来电	221
7.5.1 什么是 db4o	192	9.2.3 来电黑名单	222
7.5.2 下载和安装 db4o	193	9.2.4 在模拟器上模拟重力感应	223
7.5.3 创建和打开数据库	193	9.2.5 手机翻转静音	225
7.5.4 向数据库中插入 Java 对象	194	9.3 时间服务	227
7.5.5 从数据库中查询 Java 对象	195	9.3.1 计时器: Chronometer	227
7.5.6 高级数据查询	195	9.3.2 预约时间 Handler	229
7.5.7 更新数据库中的 Java 对象	196	9.3.3 定时器 Timer	230
7.5.8 删除数据库中的 Java 对象	196	9.3.4 在线程中更新 GUI 组件	232
7.6 小结	196	9.3.5 全局定时器 AlarmManager	234
第 8 章 App 之间的通信	197	9.3.6 定时更换壁纸	234
8.1 Intent 与 Activity	197	9.3.7 多次定时提醒	237
8.1.1 用 Intent 启动 Activity, 并在 Activity 之间传递数据	197	9.4 跨进程访问 (AIDL 服务)	239
8.1.2 调用其他应用程序中的 Activity (拨打 电话、浏览网页、发 E-mail 等)	200	9.4.1 什么是 AIDL 服务	240
8.1.3 定制自己的 Activity Action	204	9.4.2 建立 AIDL 服务的步骤	240
8.1.4 将电子词典的查询功能共享 成一个 Activity Action	205	9.4.3 建立 AIDL 服务	240
8.2 接收和发送广播	207	9.4.4 传递复杂数据的 AIDL 服务	243
8.2.1 接收系统广播	208	9.5 小结	248
		第 10 章 网络技术	249
		10.1 可装载网络数据的控件	249
		10.1.1 装载网络数据的原理	250
		10.1.2 将网络图像装载到 ListView	

控件中	250	11.1 图形	295
10.1.3 Google 图像画廊 (Gallery)	253	11.1.1 图形绘制基础	296
10.2 WebView 控件	257	11.1.2 绘制基本的图形和文本	298
10.2.1 用 WebView 控件浏览网页	257	11.1.3 绘制位图	301
10.2.2 手机浏览器	258	11.1.4 用两种方式绘制位图	302
10.2.3 用 WebView 控件装载 HTML 代码	259	11.1.5 设置颜色的透明度	303
10.2.4 将英文词典整合到 Web 页中 (JavaScript 调用 Java 方法)	260	11.1.6 可任意改变透明度的位图	303
10.3 访问 HTTP 资源	262	11.1.7 旋转图像	304
10.3.1 提交 HTTP GET 和 HTTP POST 请求	262	11.1.8 旋转动画	305
10.3.2 HttpURLConnection 类	265	11.1.9 扭曲图像	306
10.3.3 上传文件	265	11.1.10 按圆形轨迹扭曲图像	307
10.3.4 远程 Apk 安装器	268	11.1.11 拉伸图像	310
10.3.5 调用 Webservice	270	11.1.12 拉伸图像演示	310
10.3.6 通过 Webservice 查询产品信息	271	11.1.13 路径	312
10.4 Internet 地址	275	11.1.14 沿着路径绘制文本	316
10.4.1 Internet 地址概述	275	11.1.15 可在图像上绘制图形的画板	318
10.4.2 创建 InetAddress 对象	276	11.2 音频和视频	323
10.4.3 判断 IP 地址类型	278	11.2.1 使用 MediaPlayer 播放 MP3 文件	323
10.5 客户端 Socket	279	11.2.2 使用 MediaRecorder 录音	324
10.5.1 Socket 类基础	280	11.2.3 使用 VideoView 播放视频	325
10.5.2 多种连接服务端的方式	282	11.2.4 使用 SurfaceView 播放视频	326
10.5.3 客户端 Socket 的超时	283	11.3 小结	327
10.5.4 Socket 类的 getter 和 setter 方法	283	第 12 章 Fragment	328
10.5.5 Socket 的异常	289	12.1 什么是 Fragment	328
10.6 服务端 Socket	290	12.2 Fragment 的设计原则	330
10.6.1 创建 ServerSocket 对象	290	12.3 Fragment 初步	331
10.6.2 设置请求队列的长度	291	12.3.1 Fragment 的使用方法	331
10.6.3 绑定 IP 地址	292	12.3.2 实例: 一个简单的 Fragment App	331
10.6.4 默认构造方法的使用	292	12.4 Fragment 的生命周期	335
10.6.5 读取和发送数据	293	12.4.1 生命周期详解	335
10.6.6 关闭连接	294	12.4.2 实例: Fragment 生命周期演示	340
10.7 小结	294	12.5 动态创建 Fragment	343
第 11 章 多媒体技术	295	12.6 Fragment 与 Activity 之间的交互	346
		12.7 回退栈	348
		12.8 小结	350

第 13 章 ActionBar	351	16.3.3 让 RecyclerView 控件横屏显示	392
13.1 ActionBar 简介	351	16.4 CardView 控件	393
13.2 ActionBar 基础	352	16.4.1 出现 R\$styleable 没找到错误的 原因	393
13.2.1 隐藏/显示 ActionBar	352	16.4.2 在布局文件中使用 CardView	395
13.2.2 Action 按钮	354	16.4.3 用 Java 代码来控制 CardView 控件	396
13.3 应用程序图标导航	357	16.5 小结	398
13.4 收缩和展开 Action View	358	第 17 章 Android 5.x 新特性: Drawable 资源	399
13.5 导航标签	361	17.1 着色	399
13.6 下拉导航列表	366	17.2 矢量 Drawable 资源	400
13.7 小结	368	17.3 矢量动画	402
第 14 章 Android 5.x 新特性: 质感主题	369	17.3.1 指针会动的时钟	402
14.1 使用不同的质感主题	369	17.3.2 笑脸表情	404
14.2 修改质感主题的默认属性值	371	17.4 Ripple Drawable 资源	406
14.3 小结	372	17.5 小结	409
第 15 章 Android 5.x 新特性: 阴影和 视图裁剪	373	第 18 章 其他 Android 5.x 新特性	410
15.1 阴影	373	18.1 以 Immersive 模式隐藏及显示状态栏 和导航条	410
15.1.1 高度和 Z 轴的位置	373	18.1.1 什么是 Immersive 模式	410
15.1.2 带有阴影的拖动效果	376	18.1.2 实现界面的布局	411
15.2 视图裁剪	377	18.1.3 隐藏和显示	411
15.3 小结	379	18.1.4 监听隐藏和显示状态	412
第 16 章 Android 5.x 新特性: 列表和 卡片控件	380	18.2 新的通知中心	412
16.1 RecyclerView 控件简介	380	18.3 续航与安全性	413
16.2 用 RecyclerView 控件实现垂直 列表效果	381	18.4 更多的新功能	414
16.2.1 建立 Model	381	18.5 小结	414
16.2.2 定制列表项的分隔条	382	第 19 章 2D 动画	415
16.2.3 实现 Adapter 类	383	19.1 帧 (Frame) 动画	415
16.2.4 如何使用 RecyclerView 控件	385	19.1.1 AnimationDrawable 与帧动画	416
16.2.5 用 RecyclerView 控件实现增加和 删除列表项的效果	386	19.1.2 通过帧动画方式播放 GIF 动画	417
16.3 用 RecyclerView 控件实现画廊的效果	389	19.1.3 播放帧动画的子集	420
16.3.1 为画廊提供数据	390	19.2 补间 (Tween) 动画	422
16.3.2 自定义 RecyclerView 控件	391	19.2.1 移动补间动画	422

19.2.2	循环向右移动的 EditText 与 上下弹跳的球	424	21.3.1	Brightness 特效	464
19.2.3	缩放补间动画	426	21.3.2	反差特效 (Contrast)	464
19.2.4	跳动的心	427	21.3.3	Crossprocess 特效	465
19.2.5	旋转补间动画	428	21.3.4	纪录片 (Documentary) 特效	465
19.2.6	旋转的星系	429	21.3.5	双色调 (Duotone) 特效	465
19.2.7	透明度补间动画	430	21.3.6	鱼眼 (Fish Eye) 特效	465
19.2.8	投掷炸弹	431	21.3.7	垂直翻转特效	466
19.2.9	振动效果	434	21.3.8	灰度特效	467
19.2.10	自定义动画渲染器 (Interceptor)	434	21.3.9	Lomoish 特效	467
19.2.11	以动画方式切换 View 的 控件 ViewFlipper	436	21.3.10	底片特效	467
19.3	小结	437	21.3.11	色调特效	467
第 20 章	OpenGL ES 编程	438	21.4	让特效生效	468
20.1	OpenGL ES 简介	438	21.5	小结	468
20.2	在 3D 空间中绘图	440	第 22 章	资源、国际化与自适应	469
20.2.1	绘制 3D 图形的第一步	440	22.1	Android 中的资源	470
20.2.2	定义顶点	442	22.1.1	Android 怎么存储资源	470
20.2.3	绘制三角形	443	22.1.2	资源的种类	470
20.2.4	三角形合并法绘制矩形	445	22.1.3	资源文件的命名	471
20.2.5	顶点法绘制矩形	447	22.2	定义和使用资源	471
20.2.6	顶点的选取顺序	448	22.2.1	使用系统资源	471
20.2.7	索引法绘制矩形	449	22.2.2	字符串 (String) 资源	473
20.2.8	基于 OpenGL ES 的动画原理	450	22.2.3	数组 (Array) 资源	474
20.2.9	旋转的矩形	450	22.2.4	颜色 (Color) 资源	476
20.3	视图	452	22.2.5	尺寸 (Dimension) 资源	477
20.3.1	有趣的比喻: 照相机拍照	453	22.2.6	类型 (Style) 资源	479
20.3.2	模型变换: 立方体旋转	454	22.2.7	主题 (Theme) 资源	480
20.3.3	用 gluLookAt 方法变换视图	457	22.2.8	绘画 (Drawable) 资源	482
20.4	颜色	458	22.2.9	动画 (Animation) 资源	483
20.5	小结	460	22.2.10	菜单 (Menu) 资源	483
第 21 章	媒体特效 API	461	22.2.11	布局 (Layout) 资源	486
21.1	实现主界面布局	461	22.2.12	属性 (Attribute) 资源	486
21.2	初始化主界面	463	22.2.13	改进可显示图标 的 IconTextView 控件	488
21.3	媒体特效 API 演示	464	22.2.14	XML 资源	490
			22.2.15	RAW 资源	491

22.2.16	ASSETS 资源	492	24.3.3	将文件中的小写字母转换成大写 字母 (NDK 版本)	529
22.3	国际化和资源自适应	492	24.3.4	配置 Android.mk 文件	532
22.3.1	对资源进行国际化	492	24.3.5	Android NDK 定义的变量	533
22.3.2	Locale 与国际化	494	24.3.6	Android NDK 定义的函数	533
22.3.3	常用的资源配置	494	24.3.7	描述模块的变量	534
22.4	小结	495	24.3.8	配置 Application.mk 文件	535
第 23 章	访问 Android 手机的硬件	496	24.4	小结	536
23.1	在手机上测试硬件	496	第 25 章	蓝牙技术	537
23.1.1	安装 Android USB 驱动	497	25.1	蓝牙简介	537
23.1.2	在手机上测试程序	498	25.2	打开和关闭蓝牙设备	538
23.1.3	在手机上调试程序	499	25.3	搜索蓝牙设备	539
23.2	录音	500	25.4	蓝牙数据传输	541
23.3	控制手机摄像头 (拍照)	500	25.5	蓝牙通信一定需要 UUID 吗	544
23.3.1	调用系统的拍照功能	501	25.6	小结	545
23.3.2	实现自己的拍照 Activity	503	第 26 章	有趣的 Android 技术	546
23.4	传感器在手机中的应用	507	26.1	手势 (Gesture)	546
23.4.1	在应用程序中使用传感器	507	26.1.1	创建手势文件	546
23.4.2	电子罗盘	509	26.1.2	通过手势输入字符串	547
23.4.3	计步器	510	26.1.3	通过手势调用程序	549
23.5	GPS 与地图定位	511	26.1.4	编写自己的手势创建器	550
23.5.1	Google 地图	511	26.2	让手机说话 (TTS)	551
23.5.2	用 GPS 定位到当前位置	515	26.3	动态壁纸	552
23.6	WIFI	516	26.4	小结	559
23.7	小结	519	第 27 章	Android App 性能调优	560
第 24 章	NDK 技术	520	27.1	刷新频率与丢帧	560
24.1	Android NDK 简介	520	27.2	开发者选项与查看 GPU 负载	561
24.2	安装、配置和测试 NDK 开发环境	521	27.3	GPU 渲染时间与性能调优	564
24.2.1	系统和软件要求	521	27.4	Overdraw 与区域绘制	565
24.2.2	下载和安装 Android NDK	522	27.5	内存抖动与性能	566
24.2.3	下载和安装 Cygwin	522	27.6	小结	568
24.2.4	配置 Android NDK 的开发环境	525	第 28 章	内存泄露检测	569
24.2.5	编译和运行 NDK 自带的例子	526	28.1	造成内存泄露的原因	569
24.3	Android NDK 开发	528	28.1.1	非静态内嵌类	569
24.3.1	JNI 接口设计	528	28.1.2	Handler 要用静态变量或弱引用	570
24.3.2	编写 Android NDK 程序的步骤	529			

28.1.3 线程引发的内存泄露·····	571	29.6 其他功能的实现·····	585
28.1.4 其他可能会造成内存泄露的情况··	572	29.7 小结·····	585
28.1.5 弱引用（WeakReference）和软引用（SoftReference）·····	572	第 30 章 项目实战：基于 XMPP 的 IM 客户端 ·····	586
28.2 内存泄露检测工具：Eclipse MAT·····	573	30.1 XMPP 简介·····	586
28.3 小结·····	576	30.2 Openfire 安装与配置·····	587
第 29 章 项目实战：超级手电筒 ·····	577	30.3 Spark 的安装和使用·····	590
29.1 手电筒 APP 简介·····	577	30.4 用户登录·····	591
29.2 手电筒的架构·····	578	30.5 获取好友信息·····	591
29.3 手电筒照明·····	579	30.6 添加好友·····	594
29.3.1 手电筒的布局·····	579	30.7 发送聊天信息·····	594
29.3.2 通过代码调整控制区域位置·····	580	30.8 接收聊天信息·····	595
29.3.3 打开和关闭闪光灯·····	580	30.9 其他功能·····	596
29.4 警告灯·····	581	30.10 小结·····	596
29.5 发送莫尔斯密码·····	583		

Android 入门

Google 于 2005 年并购了成立仅 22 个月的高科技企业 Android，展开了短信、手机检索、定位等业务，同时基于 Linux 的 Android 平台也进入了开发阶段。Google 在 2007 年 11 月 5 日发布了 Android 的第一个版本。在刚发布之初，Android 并没有引起业界太多的关注。但随着 Google 组建的开放手机联盟不断有新生力量加入，Android 这个初出茅庐的小子已成为与 iPhone 分庭抗礼的生力军。

在作者编写本书时，至少有数十家不同规模的手机厂商宣布加入 Android 阵营。基于 Android 的手机也是琳琅满目。现在让我们进入时空隧道，回到 2008 年 9 月 23 日（北京时间 2008 年 9 月 23 日 22:30）的美国纽约，Google 和运营商 T-Mobile 共同发布了世界上第一款安装 Android 系统的手机 T-Mobile G1。由于这款手机的出色表现，使 Android 真正成为了万众瞩目的焦点。正是因为 Android 及其他几项创新，在 17 个月后的 2010 年 2 月 25 日，美国著名商业杂志《Fast Company》评选的 2010 年全球最具创新力公司 50 强中，Google 位列移动领域十大最具创新力公司榜首。2011 年 Android 的全球市场占有率首次超过了 iPhone (iOS)，成为了全球使用率最高、最受欢迎的手机操作系统。



本章内容

- Android 的系统构架
- 搭建 Android 开发环境
- Android SDK 中的常用命令行工具（包括 adb、android 和 mkcdsard）

1.1 Android 的基本概念

Android 的中文意思是“机器人”。但在移动领域，大家一定会将 Android 与 Google 联系起来。Android 本身就是一个操作系统，只是这个操作系统是基于 Linux 内核的。也就是说，从理论上，基于 Linux 的软件移植到 Android 上是最容易的。Android 是由几十家科技公司和手机公司组成的“开放手机联盟”共同研发的，而且完全免费开源，这将大大降低新型手机设备的研发成本，甚至已成为“山寨”机的首选。

1.1.1 Android 简介

Android 作为 Google 最具创新的产品之一，正受到越来越多的手机厂商、软件厂商、运营商及个人开发者的追捧。目前 Android 阵营主要包括 HTC（宏达电）、T-Mobile、高通、三星、LG、摩托罗拉、ARM、软银移动、中国移动、小米、华为等。虽然这些机构有着不同的性质，但它们都在 Android 平台的基础上不断创新，让用户体验到最优质的服务。下面欣赏几款具有代表性的 Android 手机。第一款毫无疑问，就是世界上第一部 Android 手机 T-Mobile G1，如图 1-1 所示。这款手机带有一个物理键盘（硬键盘），可以通过侧滑拉出。第二款是创下了销售奇迹的 HTC Hero，也称为 G3，如图 1-2 所示。这款手机的显著特征是下方有一个突起的小“下巴”。除此之外，HTC Hero 绚丽的 Sense 界面也成为 Android 手机中一道亮丽的风景。最后一款则是带有 Google 字样的 Nexus S（由三星代工），俗称 Google 的二儿子，如图 1-3 所示。Nexus S 是 Google 用来测试最新版 Android 的，因此，Nexus S 总会比其他厂商的手机更早升级到 Android 的新版本。

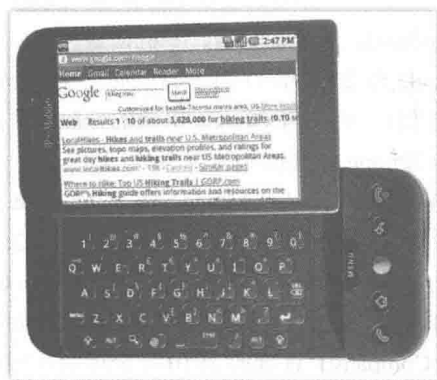


图 1-1 T-Mobile G1

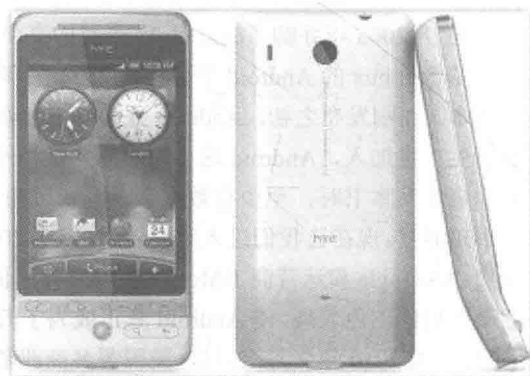


图 1-2 HTC Hero



图 1-3 Nexus S

欣赏完这么多“超酷”的手机，现在来看一下 Android 到底有什么魔力，可以让众多的粉丝为之疯狂。据粗略统计，Android 至少有如下 8 件制胜法宝：

- 开放性。Android 平台是免费、开源的。而且 Google 通过与运营商、设备制造商、开发商等机构形成的战略联盟，希望通过共同制定标准使 Android 成为一个开放式的生态系统。
- 应用程序的权限由开发人员决定。编写过 Symbian、Java ME 程序的读者应该能体会到这些程序在发布时有多麻烦。如果访问到某些限制级的 API，不是出现各种各样的提示，就是根本无法运行。要想取消这些限制，就得向第三方的认证机构购买签名，而且价格不菲。而 Android 平台的应用程序就幸福得多。要使用限制级的 API，只需要在自己的应用程序中配置一下即可，完全是 DIY。这也在某种程度上降低了 Android 程序的开发成本。
- 我的平台我作主。Android 上的所有应用程序都是可替换和扩展的，即使是拨号、Home 这样的核心组件也是一样。只要我们有足够的想象力，就可以缔造出一个独一无二、完全属于自己的 Android 世界。
- 应用程序之间的无障碍沟通。应用程序之间的通信一直令人头痛，而在 Android 平台上无疑是一种享受。在 Android 平台上，应用程序之间至少有 4 种沟通方式。很难说哪一种方式更好，但它们的的确托起了整个 Android 的应用程序框架。
- 拥抱 Web 的时代。如果想在 Android 应用程序中嵌入 HTML、JavaScript，那真是再容易不过了。基于 Webkit 内核的 WebView 组件会完成一切。更值得一提的是，JavaScript 还可以和 Java 无缝地整合在一起。
- 物理键盘和虚拟键盘双管齐下。从 Android 1.5 开始，Android 同时支持物理键盘和虚拟键盘，从而可大大丰富用户的输入选择。尤其是虚拟键盘，已成为 Android 手机中主要的输入方式。
- 个性的充分体现。21 世纪是崇尚个性的时代。Android 也紧随时代潮流，提供了众多体现个性的功能。例如，Widget、Shortcut、Live WallPapers，无一不尽显手机的华丽与时尚。
- 舒适的开发环境。Android 的主流开发环境是 Eclipse + ADT + Android SDK。它们可以非常容易地集成到一起，而且在开发环境中运行程序要比 Symbian 这样的传统手机操作系统更快，调试更方便。

虽然 Android 的特点还有很多，但这已经不重要。重要的是，现在 Android 已经成为万众瞩目的国际巨星，其未来将令人充满期望。

1.1.2 Android 的系统构架

通过上一节的介绍，我们对 Android 的特点已经有了一个初步的了解。本节将介绍 Android 的系统构架。来看看 Android 的体系结构，如图 1-4 所示。

从图 1-4 可以看出，Android 分为 4 层，从高到低分别是应用层、应用框架层、系统运行库层和 Linux 内核层。

下面将对这 4 层进行简单介绍：

- 应用层。该层由运行在 Dalvik 虚拟机（为 Android 专门设计的基于寄存器的 Java 虚拟机，运行 Java 程序的速度更快）上的应用程序（主要由 Java 语言编写）组成。例如，日历、地图、浏览器、联系人管理，都属于应用层上的程序。

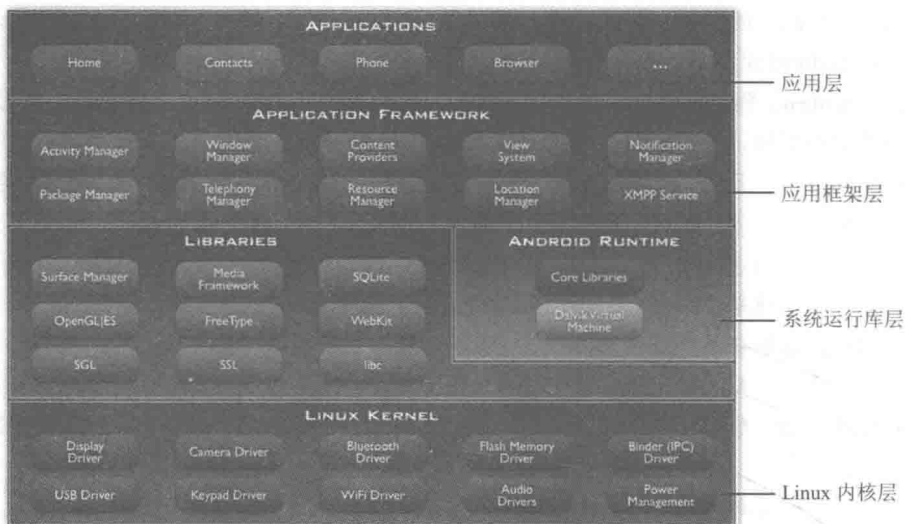


图 1-4 Android 的体系结构

- 应用框架层。该层主要由 View、通知管理器 (Notification Manager)、活动管理器 (Activity Manager) 等由开发人员直接调用的组件组成。
- 系统运行库层。Java 本身是不能直接访问硬件的。要想让 Java 访问硬件, 必须使用 NDK 才可以。NDK 是一些由 C/C++ 语言编写的库, 这些程序也是该层的主要组成部分。该层主要包括 C 语言标准库、多媒体库、OpenGL ES、SQLite、Webkit、Dalvik 虚拟机等。也就是说, 该层是对应用框架层提供支持的层。
- Linux 内核层。该层主要包括驱动、内存管理、进程管理、网络协议栈等组件。目前 Android 的版本基于 Linux 3.4 内核。

1.2 Android 开发环境的搭建

工欲善其事, 必先利其器。开发 Android 应用程序总不能直接用记事本开发吧(那些超级大牛除外)。找到合适的开发工具是学习 Android 开发的第一步。而更多地了解 Android 的开发环境将会对进一步学习 Android 保驾护航。

1.2.1 开发 Android 程序需要些什么

开发 Android 程序至少需要如下工具和开发包:

- JDK (建议安装 JDK1.6 及其以上版本)
- Eclipse
- Android SDK
- ADT (Android Development Tools, 开发 Android 程序的 Eclipse 插件)

其中 JDK 的安装非常简单, 读者可以在官方网站下载 JDK 的最新版, 并按着提示进行安装。Eclipse