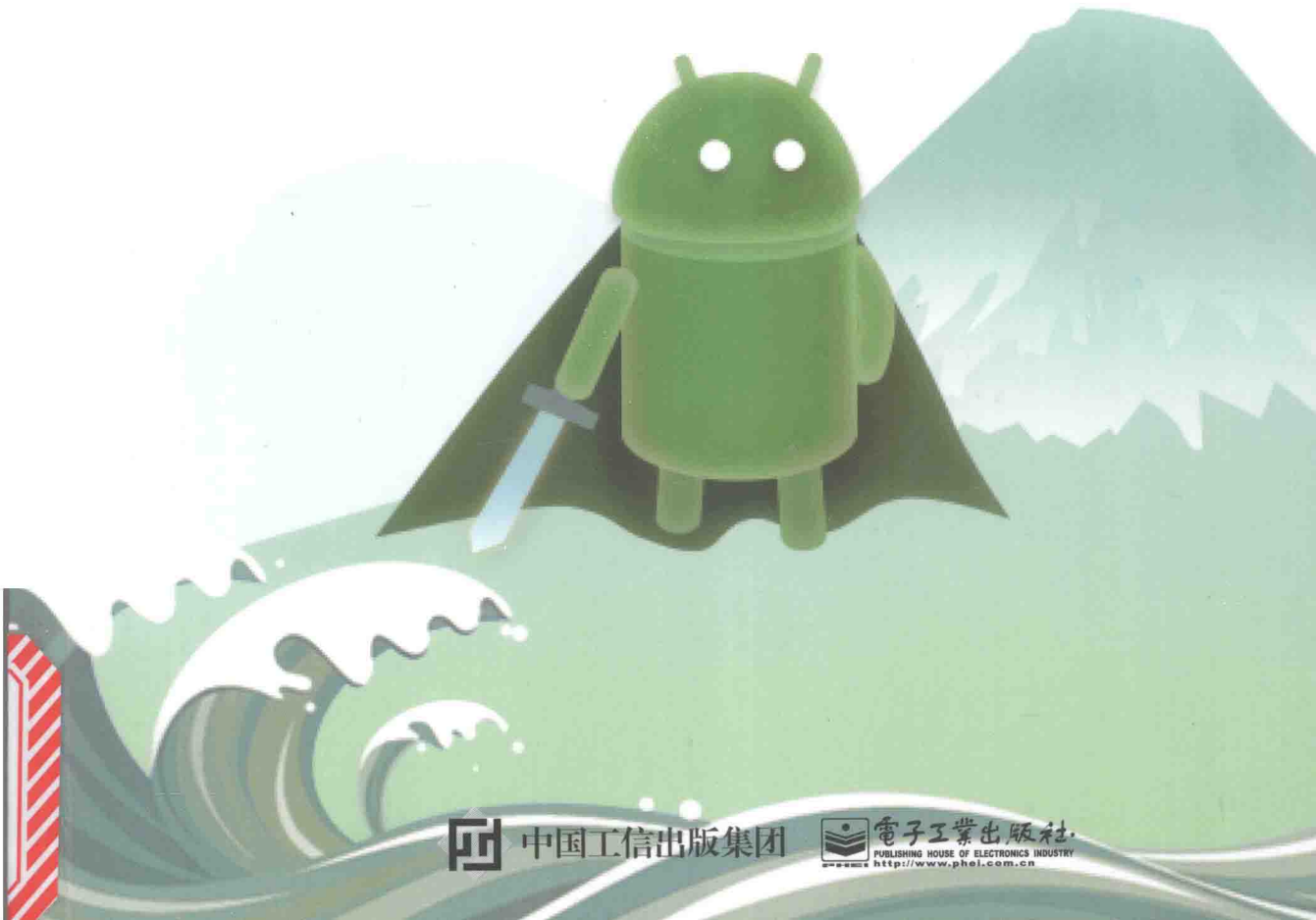
本书基于Android 7.0和Android Studio，对Android开发进阶要点进行深入讲解，为工程师的进阶之路带来指引和光明。

Broadview®
www.broadview.com.cn

Android 进阶之光

刘望舒 著



中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Android 进阶之光

刘望舒 著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书是一本 Android 进阶类书籍，书中各知识点由浅入深、环环相扣，最终这些知识点形成了一个体系结构。本书共分为 11 章。第 1 章介绍 Android 5.0 到 Android 7.0 的新特性。第 2 章介绍 Material Design。第 3 章介绍 View 体系，包括 View 的事件分发、工作流程、自定义 View 等知识点。第 4 章介绍多线程的知识。第 5 章介绍网络编程与网络框架的知识。第 6 章介绍常用的设计模式。第 7 章介绍事件总线。第 8 到第 10 章介绍架构设计所需要的知识点。第 11 章简单介绍 Android 系统框架与 MediaPlayer 框架。

本书详细并深入讲解 Android 开发者必备的和前沿的知识，适合有一定开发基础的开发者阅读，这有助于他们提高技术水平；同时，本书系统的知识体系结构也令高级开发者从中获益良多。

图书在版编目（CIP）数据

Android 进阶之光 / 刘望舒著. —北京：电子工业出版社，2017.7
ISBN 978-7-121-31530-5

Ⅰ. ①A… Ⅱ. ①刘… 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 108511 号

策划编辑：付　睿
责任编辑：李云静
印　　刷：北京京科印刷有限公司
装　　订：北京京科印刷有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×1092　1/16　　印张：30.75　　字数：764 千字
版　　次：2017 年 7 月第 1 版
印　　次：2017 年 7 月第 1 次印刷
定　　价：89.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819　faq@phei.com.cn。

前言

为什么写这本书

从 2008 年 Android 系统发布以来，Android 已经发展了 9 年。在此期间，Android 开发也相当火热。这时，大量人员涌入 Android 开发职场，并导致 Android 开发人才市场相对饱和。如此一来，很多 Android 开发者会发现工作越来越难找，企业对开发者的要求也越来越高，企业需求最多的不再是初中级别的 Android 工程师，而是 Android 高级工程师。但是，Android 高级工程师有限。有些人在从事了几年开发工作后，对很多技术的理解却仍停留在会用的阶段。他们对于原理不求甚解，这导致他们进入技术瓶颈期并长期无法得到提高。很多开发者为了突破技术瓶颈，看了大量的网络视频和博客。尽管如此，他们仍旧无法突破自身的技术瓶颈。其主要原因是，他们没有将学到的知识点形成体系化。因此，这就需要有一本成体系的进阶书来帮助这些开发人员成为 Android 高级工程师并突破自身的技术瓶颈。纵观市面上 Android 开发相关的书籍，其中大部分书籍是入门级别类图书，还有一部分系统源码、逆向分析和系统移植类图书，而关于应用开发进阶的书籍则少得可怜。本书正是一本成体系的应用开发进阶图书，书中所要传达的不仅仅是知识，其同时还会告诉读者以下几点。

1. 要关注 Android 新技术；
2. Java 基础和设计模式很重要；
3. 学习框架要深入其原理；
4. 要学习架构设计；
5. 要了解和学习系统源码。

本书内容

本书共分为 11 章，各章内容如下。

- 第 1 章介绍 Android 5.0、Android 6.0、Android 7.0 的新特性，包括 Android 5.0 的 RecyclerView、Android 6.0 的运行时权限机制和 Android 7.0 的多窗口模式等知识点。
- 第 2 章介绍 Material Design 以及 Design Support Library 常用的控件，并给出实例将 Design Support Library 中的常用控件结合在一起使用。
- 第 3 章介绍 View 相关的进阶知识，包括 View 的滑动、View 的事件分发和 View 的工作流程。最后结合以上知识点来介绍自定义 View。
- 第 4 章介绍多线程编程，本章不仅包括基础的线程知识，还会介绍线程同步和线程池等进阶知识点，最后结合这些知识点来分析 Android 7.0 的 AsyncTask 的源码。
- 第 5 章介绍网络编程的基础知识以及常用的网络框架：Volley、OkHttp 和 Retrofit 的使用方法和原理分析。
- 第 6 章将设计模式进行分类，并介绍每个分类中常用的设计模式。
- 第 7 章介绍事件总线 EventBus 和 otto 的使用方法和原理。
- 第 8 章介绍函数响应式编程 RxJava 的使用方法，包括 RxJava 的基本使用、操作符、使用场景和源码分析等知识点。
- 第 9 章介绍注解的知识点和依赖注入框架 ButterKnife 和 Dagger2 的使用方法以及原理。
- 第 10 章介绍 Android 应用架构设计，包括 MVP 框架以及 MVP 结合 RxJava 和 Dagger2，还有 MVVM 框架相关的 Data Binding 支持库。
- 第 11 章主要是指引读者进行 Android 系统源码阅读并带其入门，介绍 Android 系统框架、源码目录和阅读源码工具，并以分析 MediaPlayer 框架的源码作为示例。

本书特色

本书主要有以下特点。

- 本书整体结构由浅入深，从最简单的第 1 章到难一些的第 11 章，其难度是逐步加深的。
- 本书为了分析一些框架的原理，会介绍一些知识点做铺垫，比如为了更好地介绍依赖注入框架，需要首先了解注解相关的知识点。再比如要分析 AsyncTask 的源码，则需要了解线程池和阻塞队列等知识点。
- 本书的知识点环环相扣，比如要介绍 MVP 框架的设计，就需要先学习 Retrofit、RxJava 和 Dagger2 的相关知识点。
- 本书对于很多知识点都有很深入的讲解。其中，对于常用的框架，比如 OkHttp、Retrofit、EventBus 和 RxJava 等不只是讲解了如何使用，而且更加深入地介绍了其原理。
- 本书是目前市场上详细介绍有关 Android 新特性、Material Design、网络框架、事件总线、RxJava、依赖注入框架和应用架构设计的难得一见的图书。

读者对象

本书的章节设计是由浅入深的，适合 Android 初、中、高级工程师阅读。本书的定位是学习 Android 的第二本书，其阅读前提是要有一定的 Android 基础。

致谢

感谢本书的策划编辑付睿，她在 CSDN 博客中发现了我，并积极推动本书的出版进度，这才使得本书得以及时出版。感谢本书的责任编辑李云静，她审稿时很细致，这使得书中的一些错误能被提早发现并改正。感谢我的父母在写书过程中对我的不断鼓励，这样我才得以全力以赴地投入编写工作。感谢所有关注我的朋友们，你们的鼓励和认可为我写博客以及写书带来了不可或缺的动力。

勘误与互动

本人虽已竭尽全力，但书中难免会有错误，欢迎大家向我反馈，我也会在独立博客和 CSDN 博客中定期发布本书的勘误信息。

本书互动地址

独立博客：http://liuwangshu.cn

CSDN 博客：http://blog.csdn.net/itachi85

GitHub：https://github.com/henrymorgen

微信公众号：刘望舒

QQ 交流群：499174415

源码下载

https://github.com/henrymorgen/android-advanced-light

http://www.broadview.com.cn/31530

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- **下载资源**：本书如提供示例代码及资源文件，均可在下载资源处下载。
- **提交勘误**：您对书中内容的修改意见可在提交勘误处提交，若被采纳，将获赠博文视点

社区积分（在您购买电子书时，积分可用来抵扣相应金额）。

- **交流互动**：在页面下方读者评论处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：http://www.broadview.com.cn/31530

目录

第 1 章
Android 新特性

作为本书的第 1 章，肯定是全书比较简单的内容，而本书面向中高级的开发者，所以基础知识讲得较少。在本章笔者会介绍 Android 5.0、Android 6.0 以及 Android 7.0 的新特性，并会详细介绍 Android 5.0 和 Android 6.0 的新特性。虽然 Android 7.0 已经发布了，但是目前 Android 5.0 和 Android 6.0 的技术仍旧没有普及。笔者希望通过本章的学习，读者能够了解各版本有什么新的功能、带给用户怎样的新体验，且能够掌握 Android 的新特性并尝试运用到项目中。

1.1 Android 5.0 新特性

Android 5.0 Lollipop 是 Google 于 2014 年 10 月 15 日发布的 Android 操作系统。北京时间 2014 年 6 月 26 日，Google I/O 2014 开发者大会在旧金山正式召开，发布了 Android 5.0 的开发者预览版。下面我们先来看看 Android 5.0 给我们带来了什么。

1.1.1 Android 5.0 主要新特性概述

作为一个 Android 开发者，我们需要了解最近的 Android 版本带来了什么特性，这样更有利于开发。谷歌（Google）在 Android 5.0 中带给我们很多惊喜。

1. 全新的 Material Design 设计风格

Material Design 是一种大胆的平面化创新（见图 1-1）。换句话说，谷歌希望能够让 Material Design 给用户带来纸张化的体验。这种新的视觉语言，在基本元素的处理上，借鉴了传统的印刷设计，以及字体版式、网格系统、空间、比例、配色和图像使用等这些基础的平面设计规范。另外，Material Design 还推崇实体隐喻理念，利用实体的表面与边缘的质感打造出视觉线索，

让用户感受到真实性。熟悉的触感让用户可以快速地理解并认知。在设计中可以在符合物理规律的基础上灵活地运用物质，打造出不同的使用体验。为了吸引用户的注意力，Material Design 还带来了有意义而且更合理的动态效果，以及维持整个系统的连续性体验。需要注意的是 Material Design 虽然是在 Android 5.0 时被提出来的，但是它也是在不断更新的，所以关于 Material Design 的内容会在第 2 章专门介绍。

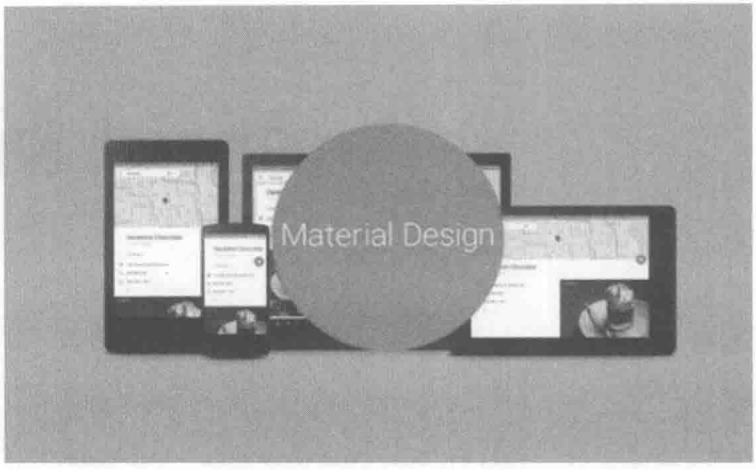


图 1-1　Material Design

2. 支持多种设备

Android 系统的身影早已出现在多种设备中，比如：智能手机、平板电脑、笔记本电脑、智能电视、汽车、智能手表甚至是各种家用电子产品等。

3. 全新的通知中心设计

谷歌在 Android 5.0 中加入了全新风格的通知系统。改进后的通知系统会优先显示对用户来说比较重要的信息，而将不太紧急的内容隐藏起来。用户只需要向下滑动就可以查看全部的通知内容，如图 1-2 所示。

4. 支持 64 位 ART 虚拟机

Android 5.0 内部的性能上也提升了不少，它放弃了之前一直使用的 Dalvik 虚拟机，改用了 ART 虚拟机，实现了真正的跨平台编译，在 ARM、X86、MIPS 等无处不在。

5. Overview

多任务视窗现在有了一个新的名字，Overview。在界面中，每一个 App 都是一张独立的卡片，拥有立体式的层叠效果，用户可以设定“最近应用程序”，通过滑动来快速切换 App，如图 1-3 所示。

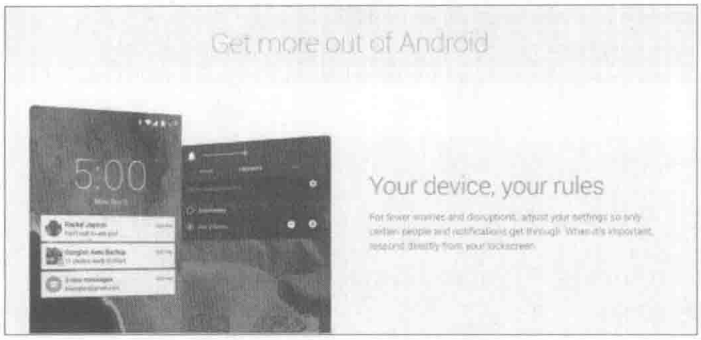


图 1-2　全新的通知中心设计

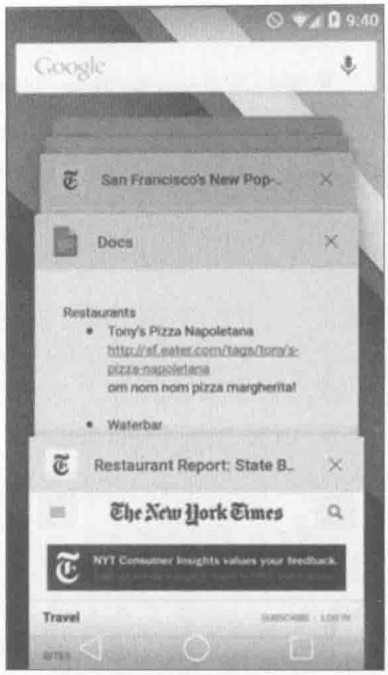


图 1-3　全新的“最近应用程序”

6. 设备识别解锁

现在个人识别解锁已经被普遍使用，比如当特定的智能手表出现在 Android 设备的附近时，就会直接绕过锁屏界面进行操作。而 Android 5.0 也增加了这种针对特定设备识别解锁的模式。换句话说，当设备没有检测到附近有可用的信任设备时，就会启动安全模式以防止未授权访问。

7. Ok Google 语音指令

当手机处于待机状态时，对你的手机轻轻说声“Ok Google”，手机即刻被唤醒，只需说出简单的语言指令，如播放音乐、查询地点、拨打电话和设定闹钟等，一切只需“说说”而已。

8. Face unlock 面部解锁

在 Android 5.0 中，Google 花费大力气优化了面部解锁功能。当用户拿起手机处理锁屏界面上的消息通知时，面部解锁功能便自动被激活。随意浏览几条消息之后，手机已经默默地完成了面部识别。

1.1.2　替换 ListView 和 GridView 的 RecyclerView

有了 ListView、GridView，为什么还需要 RecyclerView 这样的控件呢？从整体上看，RecyclerView 架构提供了一种插拔式的体验，它具有高度的解耦、异常的灵活性和更高的效率，通过设置它提供的不同 LayoutManager、ItemDecoration、ItemAnimator 可实现更加丰富多样的效果。但是 RecyclerView 也有缺点和让人头疼的地方：设置列表的分割线时需要自定义，另外列表的点击事件需要自己去实现。

1. 配置 build.gradle

要想使用 RecyclerView，我们首先要导入 support-v7 包。因为我用的是 Android Studio（本书的所有例子均基于 Android Studio），所以在此需要在 build.gradle 中加入如下代码以自动导入 support-v7 包，记得配置完再重新 Build 一下工程。

```
dependencies {
...
    compile 'com.android.support:appcompat-v7:22.2.0'
    compile 'com.android.support:recyclerview-v7:22.1.0'
}
```

2. 使用 RecyclerView

```
RecyclerView mRecyclerView= (RecyclerView) this.findViewById(R.id.id_
recyclerview);
        //设置布局管理器
        mRecyclerView.setLayoutManager(new LinearLayoutManager(this));
        // 设置 item 增加和删除时的动画
        mRecyclerView.setItemAnimator(new DefaultItemAnimator());
        mHomeAdaper=new HomeAdapter(this, mList);
        mRecyclerView.setAdapter(mHomeAdaper);
```

与 ListView 不同的一点就是，需要设置布局管理器用于设置条目的排列样式，可以是垂直排列或者水平排列。这里我们设置 setLayoutManager(new LinearLayoutManager(this))表示条目是线性排列的（默认是垂直排列的）。

```
public LinearLayoutManager(Context context) {
        this(context, VERTICAL, false);
    }
```

如果想要设置为水平排列，可以按如下代码所示编写：

```
LinearLayoutManager linearLayoutManager=new LinearLayoutManager(this);
linearLayoutManager.setOrientation(LinearLayoutManager.HORIZONTAL);
mRecyclerView.setLayoutManager(linearLayoutManager);
```

此外，RecyclerView 比 ListView 的设置要复杂一些，主要是它需要自己去自定义分割线，设置动画和布局管理器，等等。布局文件 activity_recycler_view.xml 如下：

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent" >
```