

# Android群英传（套装共2册）

作者：徐宜生

## 总目录

[Android群英传](#)

[Android群英传：神兵利器](#)

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目（CIP）数据

Android群英传/徐宜生编著. —北京：电子工业出版社，2015.9

ISBN 978-7-121-26773-4

I. ①A... II. ①徐... 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169509号

责任编辑：徐津平

印 刷：

装 订：

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 印张：22.25 字数：540千字

版 次：2015年9月第1版

印 次：2015年9月第1次印刷

定 价：59.00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。

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

服务热线：（010）88258888。

# 目录

前言

第1章 Android体系与系统架构

- 1.1 Google生态系统
- 1.2 Android系统架构
  - 1.2.1 Linux
  - 1.2.2 Dalvik与ART
  - 1.2.3 Framework
  - 1.2.4 Standard libraries
  - 1.2.5 Application
- 1.3 Android App组件架构
  - 1.3.1 Android四大组件如何协同工作
  - 1.3.2 应用运行上下文对象
- 1.4 Android系统源代码目录与系统目录
  - 1.4.1 Android系统源代码目录
  - 1.4.2 Android系统目录
  - 1.4.3 Android App文件目录

第2章 Android开发工具新接触

- 2.1 Android开发IDE介绍
  - 2.1.1 Android Studio初体验
  - 2.1.2 Android Studio配置
- 2.2 Android Studio高级使用技巧
  - 2.2.1 更新SDK
  - 2.2.2 Android Studio常用界面
  - 2.2.3 导入Android Studio工程
- 2.3 ADB命令使用技巧
  - 2.3.1 ADB基础
  - 2.3.2 ADB常用命令
  - 2.3.3 ADB命令来源

2.4 模拟器使用与配置

第3章 Android控件架构与自定义控件详解

- 3.1 Android控件架构
- 3.2 View的测量
- 3.3 View的绘制
- 3.4 ViewGroup的测量
- 3.5 ViewGroup的绘制
- 3.6 自定义View
  - 3.6.1 对现有控件进行拓展
  - 3.6.2 创建复合控件
  - 3.6.3 重写View来实现全新的控件
- 3.7 自定义ViewGroup
- 3.8 事件拦截机制分析

第4章 ListView使用技巧

- 4.1 ListView常用优化技巧
  - 4.1.1 使用ViewHolder模式提高效率
  - 4.1.2 设置项目间分隔线
  - 4.1.3 隐藏ListView的滚动条
  - 4.1.4 取消ListView的Item点击效果
  - 4.1.5 设置ListView需要显示在第几项
  - 4.1.6 动态修改ListView
  - 4.1.7 遍历ListView中的所有Item
  - 4.1.8 处理空ListView
  - 4.1.9 ListView滑动监听
- 4.2 ListView常用拓展
  - 4.2.1 具有弹性的ListView
  - 4.2.2 自动显示、隐藏布局的ListView
  - 4.2.3 聊天ListView
  - 4.2.4 动态改变ListView布局

第5章 Android Scroll分析

- 5.1 滑动效果是如何产生的
  - 5.1.1 Android坐标系
  - 5.1.2 视图坐标系
  - 5.1.3 触控事件——MotionEvent
- 5.2 实现滑动的七种方法
  - 5.2.1 layout方法
  - 5.2.2 `offsetLeftAndRight()`与`offsetTopAndBottom()`
  - 5.2.3 LayoutParams
  - 5.2.4 `scrollTo`与`scrollBy`
  - 5.2.5 Scroller
  - 5.2.6 属性动画
  - 5.2.7 ViewDragHelper

第6章 Android绘图机制与处理技巧

- 6.1 屏幕的尺寸信息
  - 6.1.1 屏幕参数
  - 6.1.2 系统屏幕密度
  - 6.1.3 独立像素密度dp
  - 6.1.4 单位转换
- 6.2 2D绘图基础
- 6.3 Android XML绘图
  - 6.3.1 Bitmap
  - 6.3.2 Shape
  - 6.3.3 Layer
  - 6.3.4 Selector
- 6.4 Android绘图技巧
  - 6.4.1 Canvas
  - 6.4.2 Layer图层
- 6.5 Android图像处理之色彩特效处理
  - 6.5.1 色彩矩阵分析
  - 6.5.2 Android颜色矩阵——ColorMatrix
  - 6.5.3 常用图像颜色矩阵处理效果
  - 6.5.4 像素点分析
  - 6.5.5 常用图像像素点处理效果
- 6.6 Android图像处理之图形特效处理
  - 6.6.1 Android变形矩阵——Matrix
  - 6.6.2 像素块分析

- 6.7 Android图像处理之画笔特效处理
  - 6.7.1 PorterDuffXfermode
  - 6.7.2 Shader
  - 6.7.3 PathEffect
- 6.8 View之孪生兄弟——SurfaceView
  - 6.8.1 SurfaceView与View的区别
  - 6.8.2 SurfaceView的使用
  - 6.8.3 SurfaceView实例
- 第7章 Android动画机制与使用技巧
  - 7.1 Android View动画框架
    - 7.1.1 透明度动画
    - 7.1.2 旋转动画
    - 7.1.3 位移动画
    - 7.1.4 缩放动画
    - 7.1.5 动画集合
  - 7.2 Android属性动画分析
    - 7.2.1 ObjectAnimator
    - 7.2.2 PropertyValuesHolder
    - 7.2.3 ValueAnimator
    - 7.2.4 动画事件的监听
    - 7.2.5 AnimatorSet
    - 7.2.6 在XML中使用属性动画
    - 7.2.7 View的animate方法
  - 7.3 Android布局动画
  - 7.4 Interpolators (插值器)
  - 7.5 自定义动画
  - 7.6 Android 5.X SVG矢量动画机制
    - 7.6.1 <path>标签
    - 7.6.2 SVG常用指令
    - 7.6.3 SVG编辑器
    - 7.6.4 Android中使用SVG
    - 7.6.5 SVG动画实例
  - 7.7 Android动画特效
    - 7.7.1 灵动菜单
    - 7.7.2 计时器动画
    - 7.7.3 下拉展开动画
- 第8章 Activity与Activity调用栈分析
  - 8.1 Activity
    - 8.1.1 起源
    - 8.1.2 Activity形态
    - 8.1.3 生命周期
  - 8.2 Android任务栈简介
  - 8.3 AndroidManifest启动模式
    - 8.3.1 standard
    - 8.3.2 singleTop
    - 8.3.3 singleTask
    - 8.3.4 singleInstance
  - 8.4 Intent Flag启动模式
  - 8.5 清空任务栈
  - 8.6 Activity任务栈使用
- 第9章 Android系统信息与安全机制
  - 9.1 Android系统信息获取
    - 9.1.1 android.os.Build
    - 9.1.2 SystemProperty
    - 9.1.3 Android系统信息实例
  - 9.2 Android Apk应用信息获取之PackageManager
    - 9.2.1 PackageManager
    - 9.2.2 Android Apk应用信息获取之ActivityManager
  - 9.3 解析Packages.xml获取系统信息
  - 9.4 Android安全机制
    - 9.5.1 Android安全机制简介
    - 9.5.2 Android系统安全隐患
    - 9.5.3 Android Apk反编译
    - 9.5.4 Android Apk加密
- 第10章 Android性能优化
  - 10.1 布局优化
    - 10.1.1 Android UI渲染机制
    - 10.1.2 避免Overdraw
    - 10.1.3 优化布局层级
    - 10.1.4 避免嵌套过多无用布局
    - 10.1.5 Hierarchy Viewer
  - 10.2 内存优化
    - 10.2.1 什么是内存
    - 10.2.2 获取Android系统内存信息
    - 10.2.3 内存回收
    - 10.2.4 内存优化实例
  - 10.3 Lint工具
  - 10.4 使用Android Studio的Memory Monitor工具
  - 10.5 使用TraceView工具优化App性能
    - 10.5.1 生成TraceView日志的两种方法
    - 10.5.2 打开TraceView日志
    - 10.5.3 分析TraceView日志
  - 10.6 使用MAT工具分析App内存状态
    - 10.6.1 生成HPROF文件
    - 10.6.2 分析HPROF文件
  - 10.7 使用Dumpsys命令分析系统状态
- 第11章 搭建云端服务器
  - 11.1 移动后端服务介绍
  - 11.2 使用Bmob创建移动后端服务
    - 11.2.1 数据服务
    - 11.2.2 推送服务
- 第12章 Android 5.X新特性详解
  - 12.1 Android 5.X UI设计初步
    - 12.1.1 材料的形态模拟
    - 12.1.2 更加真实的动画
    - 12.1.3 大色块的使用
  - 12.2 Material Design主题

- [12.3 Palette](#)
- [12.4 视图与阴影](#)
- [12.5 Tinting和Clipping](#)
  - [12.5.1 Tinting \(着色\)](#)
  - [12.5.2 Clipping \(裁剪\)](#)
- [12.6 列表与卡片](#)
  - [12.6.1 RecyclerView](#)
  - [12.6.2 CardView](#)
- [12.7 Activity过渡动画](#)
- [12.8 Material Design动画效果](#)
  - [12.8.1 Ripple效果](#)
  - [12.8.2 Circular Reveal](#)
  - [12.8.3 View state changes Animation](#)
- [12.9 Toolbar](#)
- [12.10 Notification](#)
  - [12.10.1 基本的Notification](#)
  - [12.10.2 折叠式Notification](#)
  - [12.10.3 悬挂式Notification](#)
  - [12.10.4 显示等级的Notification](#)

[第13章 Android实例提高](#)

- [13.1 移动迷宫——拼图游戏](#)
  - [13.1.1 准备工作](#)
  - [13.1.2 初始界面](#)
  - [13.1.3 拼图界面](#)
  - [13.1.4 效果预览与功能进阶](#)
- [13.2 魔幻矩阵——2048](#)
  - [13.2.1 2048概述](#)
  - [13.2.2 2048游戏分析](#)
  - [13.2.3 2048初始化工作](#)
  - [13.2.4 小方块设计](#)
  - [13.2.5 全局设置](#)
  - [13.2.6 游戏面板设计](#)
  - [13.2.7 主程序设计](#)
  - [13.2.8 功能进阶](#)
  - [13.3 实战经验总结](#)

[返回总目录](#)

## 内容简介

本书针对具有一定Android开发基础的读者，以通俗易懂的语言介绍了Android开发的进阶技巧。全书共分为13章，讲解了Android体系与系统架构、Android开发工具新接触、Android控件架构与自定义控件详解、List View使用技巧、Android Scroll分析、Android绘图机制与处理技巧、Android动画机制与使用技巧、Activity与Activity调用栈分析、Android系统信息与安全机制、Android性能优化、搭建云端服务器、Android 5.X新特性详解、Android实例提高。本书最后通过实例来展示如何实际实践这些技巧，让读者更好地体会如何提高Android开发。

书中运用大量实际开发中的实例，并结合最新的Android开发SDK和IDE，让读者能够了解最前沿的开发技术。本书适合具有一定开发基础的开发者，帮助开发者快速提高开发能力。

# 前言

## 写书之前

当电子工业出版社的官杨女士邀请我写书之前，我一直在CSDN博客坚持发表Android相关的技术博客，并给在线教育网站——慕课网录制Android相关的教学视频。自大学接触Android以来，通过不断的学习，逐渐对Android开发有了越来越深入的了解。回想当初学习Android的时候，还是有很多感触的。由于对Java语言的熟悉，因此Android的入门就显得非常轻松，然而学习一段时间后，虽然对Android的基础内容已经掌握得比较熟练，但仍然无法做出比较满意的App。一方面想通过综合的App来提升自己学习的成就感，一方面又无法融会贯通知识点，创作过程中坎坷不断。其实这段过程与大多数人的高中时期是非常相像的，记得当时在高中的时候，很多人都会遇到一个学习的瓶颈期，在这段时间里我们对基础知识掌握得非常熟练，但对稍微拔高一点的内容就力不从心了，总是不能发挥出应有的能力。其实不管学习什么东西都是一样，都存在这样的瓶颈期，这时候你并不是没有能力去解决这些问题，而是没有找到解决问题的方法。我觉得克服瓶颈期的最好办法就是六个字——总结、归纳、演绎。

我们在学习某项知识的时候，不能只停留在最基础的层面上，你不仅仅是一个学习者，更应该是一个创造者。就拿学习Android来说，如果仅仅去学习如何使用Android提供的API，或者去网上寻找现成的解决方法，那么即使再熟练，也还是一个低级的层次。当基础的东西掌握熟练之后，就应该总结知识、归纳知识，并将其扩充到自己的认知、理解中。你需要记忆的不仅仅是API的使用方法，更应该了解为何Android要这样设计，API的参数不记得可以去查，但对它的理解，却是属于自己的。这就好像高中的物理公式，很多人觉得非常难记，的确如果单纯地看那些字母组成的公式确实非常难记，但实际上，那些复杂的物理公式都是由基本的公式一步步推导出来的，每个字母都有它背后的含义，结合这些东西去记，公式就像活了一样。同样，Android的学习也是如此，站在知识体系的高度上重新去理解那些基础知识，你才能真正地把它们融合起来，任何一个组件的设计都不是独立的，总结、归纳、演绎它们之间的联系才是记忆知识最好的方法。

我跟很多Android开发者一样，也经历了这样一个阶段，因此对于开发者，特别是自学开发者的这段时期的心情深有体会，所以在出版社的编辑邀请我写书的时候，我就萌生了写这样一本书的念头。期望从我的学习经历、开发经验中整理出正确的学习方法，让开发者少走弯路，帮助Android开发者早日度过瓶颈期。

## 本书内容

我在学习、研究Android的过程中，也读过很多前辈们写的Android书籍，这些书籍大多分为三种类型：基础型、实例型和底层型。

基础型是市面上最多的一类书籍，这些书籍基本上有一个共性，就是按照Android组件类型、资源类型、网络、传感器等方面进行非常细致地讲解。在学习的初期，这样的书确实是非常有必要的，它可以帮你快速地了解整个Android的内容，但里面的很多东西，其实用的并不多，全部罗列在一起，反而增加了记忆的难度。实例型的书籍是最受开发者喜爱的一类，大家都希望通过看这些实例型的书籍来快速提高创造的成就感，但是很多时候，在看完这类书后却发现，除了书中讲的实例之外，还是无法独立完成一个App。而底层型的书籍相对于应用层的书籍来说是最难读懂的，很多应用开发者都希望通过了解底层的实现细节来更好地掌握Android系统，这当然是非常正确的，不过这些底层型的书籍的作者，大多都是经验非常丰富的高手，书中包含了大量的底层实现细节，初学者在阅读时，由于功力不够会导致越读越没有信心。

所以我在写这本书的时候，就在想如何才能避免上面的类似情况发生。记得还是在读高中的时候，市面上的参考书，基本上也是这样一个情况，但是有一本书却给我留下了很不错的印象——重难点手册。这本书不像其他参考书那样把书本上所有的知识点都罗列出来，也不会通过题海战术让学生背题，而是通过知识点的关系图理清知识点之间的联系，再通过分析重点、难点的知识点，来提高学生对知识体系的认识与理解。我期望我写的这本书能达到这样的效果，那么所有的辛苦也就值得了。

而这本书之所以取名为《Android群英传》，是因为我想通过比较轻松有趣的名字来提高读者的学习兴趣。Android中的一个知识点，并非是枯燥乏味的，通过一些神话、传说故事来类比这些知识点，可以让读者在阅读过程中，能体会到学习的乐趣。

创世纪（第1~2章）：讲解Android背景知识与Android架构、IDE相关的基本知识点。讲解Android体系架构，好比上帝的创世纪，告诉读者如何理解Android的架构；讲解Android开发的工具，好比上古四大神器，让读者欲善其事，必先利其器。

群英会（第3~12章）：针对Android开发中的重点、难点，讲解开发中重要的经验和技巧，并总结、归纳其使用方法。这几章讲解了Android中一些比较重要的组件和设计方法，笔者分别为它们著书立传，让每个知识点都好比一个英雄的传记。例如讲解Android控件架构，好比神笔马良的故事，让读者知道如何去设计控件。

地下城（第13章）：通过实例开发过程的讲解，综合运用知识点。本篇带领读者来到地下城的挑战，综合前面的知识点，好比读者在玩游戏的时候，在升级过程中，去挑战一些比较有难度的地下城，获得更多的经验。



## 本书读者对象

本书适用于各个层次的Android开发者。对于Android初学者来说，可以通过前两部分的内容，快速建立Android开发的知识体系；对于有经验的Android开发者来说，可以通过对这本书的学习快速提高自己的开发经验；对于处于瓶颈期的开发者来说，更是可以非常有效地提高开发能力。

## 本书特色

- 本书各个章节之间并没有严格的递进关系，读者可以随时挑选自己感兴趣的章节开始读起。
- 本书在讲解时，尽量使用通俗易懂的讲解方式，不讲过多深入的理论和不常用的知识点，而是将这些深入分析的过程留给读者，培养读者分析、理解的能力。
- 本书所使用的代码都是基于Google最新的Android Studio进行开发的，所有的实例都在Android Studio 1.1正式版上通过测试，对读者使用Android Studio具有很好的帮助。同时，书中包含了大量Android 5.X的新功能实例，读者可以第一时间掌握Android 5.X的新特性，并运用到自己的App中。
- 书不仅介绍了Android组件之间的联系与设计思想，同时与很多Framework层的知识点联系起来，让读者知其然，并知其所以然。
- 本书中不引用大量的底层实现代码，而是先让读者对系统的实现有一个认识，等到读者遇到相关的问题时，再去查看源代码中的实现。这样就不会没有方向、摸不着头脑，而是有目的地去验证。
- 笔者作为Android的一线开发者，所写内容具有很强的实战性，而且书中很多实例都来源于平时网友、群友的问题，因此应用性强。

欢迎访问：电子书学习和下载网站 (<https://www.shgis.cn>)  
文档名称：《Android群英传（套装共2册）》徐宜生 著.epub  
请登录 <https://shgis.cn/post/267.html> 下载完整文档。  
手机端请扫码查看：

