

# 深入理解Android



经典畅销书《深入理解Android：卷I》的姊妹篇，51CTO移动开发频道和开源中国社区一致鼎力推荐！

从系统设计的角度对Java Framework包含的重要模块和服务的源代码进行细致剖析，深刻揭示其实现原理和工作机制

移动开发



邓凡平◎著

Understanding Android Internals: Volume II

## 深入理解 Android 卷 II

机械工业出版社  
China Machine Press

[深入理解Android\\_下载链接1](#)

著者:邓凡平

出版者:机械工业出版社

出版时间:2011-9-5

装帧:平装

isbn:9787111357629

《深入理解Android：卷I》是一本以情景方式对Android的源代码进行深入分析的书。内容广泛，以对Framework层的分析为主，兼顾Native层和Application层；分析深入，每一部分源代码的分析都力求透彻；针对性强，注重实际应用开发需求，书中所涵盖的

知识点都是Android应用开发者和系统开发者需要重点掌握的。

全书共10章，第1章介绍了阅读本书所需要做的准备工作，主要包括对Android系统架构和源码阅读方法的介绍；第2章通过对Android系统中的MediaScanner进行分析，详细讲解了Android中十分重要的JNI技术；第3章分析了init进程，揭示了通过解析init.rc来启动Zygote以及属性服务的工作原理；第4章分析了Zygote、SystemServer等进程的工作机制，同时还讨论了Android的启动速度、虚拟机HeapSize的大小调整、Watchdog工作原理等问题；第5章讲解了Android系统中常用的类，包括sp、wp、RefBase、Thread等类，同步类，以及Java中的Handler类和Looper类，掌握这些类的知识后方能在后续的代码分析中做到游刃有余；第6章以MediaServer为切入点，对Android中极为重要的Binder进行了较为全面的分析，深刻揭示了其本质。第7章对Audio系统进行了深入的分析，尤其是AudioTrack、AudioFlinger和AudioPolicyService等的工作原理。第8章深入讲解了Surface系统的实现原理，分析了Surface与Activity之间以及Surface与SurfaceFlinger之间的关系、SurfaceFlinger的工作原理、Surface系统中的帧数据传输以及LayerBuffer的工作流程。第9章对Vold和Rild的原理和机制进行了深入的分析，同时还探讨了Phone设计优化的问题；第10章分析了多媒体系统中MediaScanner的工作原理。

本书适合有一定基础的Android应用开发工程师和系统工程师阅读。通过对本书的学习，大家将能更深刻地理解Android系统，从而自如应对实际开发中遇到的难题。

作者介绍:

邓凡平，资深Android开发工程师，热衷于Android源代码的研究，对Android的架构设计和实现原理有非常深刻的认识和理解，应用开发经验也十分丰富。目前就职于国内一家领先的Android企业，负责Framework的开发和维护。乐于分享，活跃于CSDN等专业社区，撰写的Android Framework源码的系列文章深受读者欢迎。此外，他对Linux内核、C/C++/Python相关的技术，以及高性能网络服务器和多核并行开发等也有一定的研究。

目录: 第1章 阅读前的准备工作 / 1
1.1 系统架构 / 2
1.1.1 Android系统架构 / 2
1.1.2 本书的架构 / 3
1.2 搭建开发环境 / 4
1.2.1 下载源码 / 4
1.2.2 编译源码 / 6
1.3 工具介绍 / 8
1.3.1 Source Insight介绍 / 8
1.3.3 Busybox的使用 / 11
1.4 本章小结 / 12
第2章 深入理解JNI / 13
2.1 JNI概述 / 14
2.2 学习JNI的实例：MediaScanner / 15
2.3 Java层的MediaScanner分析 / 16
2.3.1 加载JNI库 / 16
2.3.2 Java的native函数和总结 / 17
2.4 JNI层MediaScanner的分析 / 17
2.4.1 注册JNI函数 / 18
2.4.2 数据类型转换 / 22
2.4.3 JNIEnv介绍 / 24
2.4.4 通过JNIEnv操作jobject / 25
2.4.5 jstring介绍 / 27

2.4.6 JNI类型签名介绍 / 28
2.4.7 垃圾回收 / 29
2.4.8 JNI中的异常处理 / 32
2.5 本章小结 / 32
第3章 深入理解init / 33
3.1 概述 / 34
3.2 init分析 / 34
3.2.1 解析配置文件 / 38
3.2.2 解析service / 42
3.2.3 init控制service / 48
3.2.4 属性服务 / 52
3.3 本章小结 / 60
第4章 深入理解zygote / 61
4.1 概述 / 62
4.2 zygote分析 / 62
4.2.1 AppRuntime分析 / 63
4.2.2 Welcome to Java World / 68
4.2.3 关于zygote的总结 / 74
4.3 SystemServer分析 / 74
4.3.1 SystemServer的诞生 / 74
4.3.2 SystemServer的重要使命 / 77
4.3.3 关于 SystemServer的总结 / 83
4.4 zygote的分裂 / 84
4.4.1 ActivityManagerService发送请求 / 84
4.4.2 有求必应之响应请求 / 86
4.4.3 关于zygote分裂的总结 / 88
4.5 拓展思考 / 88
4.5.1 虚拟机heapsize的限制 / 88
4.5.2 开机速度优化 / 89
4.5.3 Watchdog分析 / 90
4.6 本章小结 / 93
第5章 深入理解常见类 / 95
5.1 概述 / 96
5.2 以“三板斧”揭秘RefBase、sp和wp / 96
5.2.1 第一板斧——初识影子对象 / 96
5.2.2 第二板斧——由弱生强 / 103
5.2.3 第三板斧——破解生死魔咒 / 106
5.2.4 轻量级的引用计数控制类LightRefBase / 108
5.2.5 题外话——三板斧的来历 / 109
5.3 Thread类及常用同步类分析 / 109
5.3.1 一个变量引发的思考 / 109
5.3.2 常用同步类 / 114
5.4 Looper和Handler类分析 / 121
5.4.1 Looper类分析 / 122
5.4.2 Handler分析 / 124
5.4.3 Looper和Handler的同步关系 / 127
5.4.4 HandlerThread介绍 / 129
5.5 本章小结 / 129
第6章 深入理解Binder / 130
6.1 概述 / 131
6.2 庖丁解MediaServer / 132
6.2.1 MediaServer的入口函数 / 132
6.2.2 独一无二的ProcessState / 133
6.2.3 时空穿越魔术——defaultServiceManager / 134

6.2.4 注册MediaPlayerService /	142
6.2.5 秋风扫落叶—StartThread Pool和Join Thread Pool分析 /	149
6.2.6 你彻底明白了吗 /	152
6.3 服务总管ServiceManager /	152
6.3.1 ServiceManager的原理 /	152
6.3.2 服务的注册 /	155
6.3.3 ServiceManager存在的意义 /	158
6.4 MediaPlayerService和它的Client /	158
6.4.1 查询ServiceManager /	158
6.4.2 子承父业 /	159
6.5 拓展思考 /	162
6.5.1 Binder和线程的关系 /	162
6.5.2 有人情味的诤告 /	163
6.5.3 匿名Service /	165
6.6 学以致用 /	166
6.6.1 纯Native的Service /	166
6.6.2 扶得起的“阿斗” (aidl) /	169
6.7 本章小结 /	172
第7章 深入理解Audio系统 /	173
7.1 概述 /	174
7.2 AudioTrack的破解 /	174
7.2.1 用例介绍 /	174
7.2.2 AudioTrack (Java空间) 分析 /	179
7.2.3 AudioTrack (Native空间) 分析 /	188
7.2.4 关于AudioTrack的总结 /	200
7.3 AudioFlinger的破解 /	200
7.3.1 AudioFlinger的诞生 /	200
7.3.2 通过流程分析AudioFlinger /	204
7.3.3 audio_track_cblk_t分析 /	230
7.3.4 关于AudioFlinger的总结 /	234
7.4 AudioPolicyService的破解 /	234
7.4.1 AudioPolicyService的创建 /	235
7.4.2 重回AudioTrack /	245
7.4.3 声音路由切换实例分析 /	251
7.4.4 关于AudioPolicy的总结 /	262
7.5 拓展思考 /	262
7.5.1 DuplicatingThread破解 /	262
7.5.2 题外话 /	270
7.6 本章小结 /	272
第8章 深入理解Surface系统 /	273
8.1 概述 /	275
8.2 一个Activity的显示 /	275
8.2.1 Activity的创建 /	275
8.2.2 Activity的UI绘制 /	294
8.2.3 关于Activity的总结 /	296
8.3 初识Surface /	297
8.3.1 和Surface有关的流程总结 /	297
8.3.2 Surface之乾坤大挪移 /	298
8.3.3 乾坤大挪移的JNI层分析 /	303
8.3.4 Surface和画图 /	307
8.3.5 初识Surface小结 /	309
8.4 深入分析Surface /	310
8.4.1 与Surface相关的基础知识介绍 /	310
8.4.2 SurfaceComposerClient分析 /	315

8.4.3 SurfaceControl分析 /	320
8.4.4 writeToParcel和Surface对象的创建 /	331
8.4.5 lockCanvas和unlockCanvasAndPost分析 /	335
8.4.6 GraphicBuffer介绍 /	344
8.4.7 深入分析Surface的总结 /	353
8.5 SurfaceFlinger分析 /	353
8.5.1 SurfaceFlinger的诞生 /	354
8.5.2 SF工作线程分析 /	359
8.5.3 Transaction分析 /	368
8.5.4 关于SurfaceFlinger的总结 /	376
8.6 拓展思考 /	377
8.6.1 Surface系统的CB对象分析 /	377
8.6.2 ViewRoot的你问我答 /	384
8.6.3 LayerBuffer分析 /	385
8.7 本章小结 /	394
第9章 深入理解Vold和Rild /	395
9.1 概述 /	396
9.2 Vold的原理与机制分析 /	396
9.2.1 Netlink和Uevent介绍 /	397
9.2.2 初识Vold /	399
9.2.3 NetlinkManager模块分析 /	400
9.2.4 VolumeManager模块分析 /	408
9.2.5 CommandListener模块分析 /	414
9.2.6 Vold实例分析 /	417
9.2.7 关于Vold的总结 /	428
9.3 Rild的原理与机制分析 /	428
9.3.1 初识Rild /	430
9.3.2 RIL_startEventLoop分析 /	432
9.3.3 RIL_Init分析 /	437
9.3.4 RIL_register分析 /	444
9.3.5 关于Rild main函数的总结 /	447
9.3.6 Rild实例分析 /	447
9.3.7 关于Rild的总结 /	459
9.4 拓展思考 /	459
9.4.1 嵌入式系统的存储知识介绍 /	459
9.4.2 Rild和Phone的改进探讨 /	462
9.5 本章小结 /	463
第10章 深入理解MediaScanner /	464
10.1 概述 /	465
10.2 android.process.media分析 /	465
10.2.1 MSR模块分析 /	466
10.2.2 MSS模块分析 /	467
10.2.3 android.process.media媒体扫描工作的流程总结 /	471
10.3 MediaScanner分析 /	472
10.3.1 Java层分析 /	472
10.3.2 JNI层分析 /	476
10.3.3 PVMediaScanner分析 /	479
10.3.4 关于MediaScanner的总结 /	485
10.4 拓展思考 /	486
10.4.1 MediaScannerConnection介绍 /	486
10.4.2 我问你答 /	487
10.5 本章小结 /	488
• • • • •	( <a href="#">收起</a> )

## 标签

Android

Andriod源码分析

android

计算机

编程

计算机科学

移动互联网

程序设计

## 评论

大致理解Android系统底层的架构,强烈推荐！IPC通信的Binder机制设计太巧妙了，还要继续 深入研究！

-----  
适合喜欢读Android源码的人

-----  
: TN929.53/7471#!

-----  
对audio分析的很好，可细读

-----  
对系统剖析很深入，很值得一读。

-----  
虽然有些过时，还是有些帮助

-----  
大体通读了一下这本书，整体上来说把Android的整个底层体系一些要点剖析得很完善，当然假如真要去好好研究一番的话，感觉还是需要去好好研究一下L/U的kernel。

-----  
我...  
嗯，好吧，其实没看明白，感觉很高深的样子。因为现在用不上，所以先把知识点混个脸熟... 嗯，好吧，脸熟都没有，就算见过了有点印象。

-----  
更多的是偏向底层的，不是应用开发

-----  
222

-----  
挺好，源码面前 了无秘密！

-----  
极客学院wiki.jikexueyuan.com20190226-20190508第一遍，了解了个大概20190509-第二遍

-----  
这本书讲得很细，就像被作者带着过了一遍android代码。但是，感觉分析不够，很多地方只是讲代码，很少讲为什么这样写代码。当然，这可能也跟本书的定位有关，本书之厚，充分证明android代码写得糟糕，理解困难。不推荐app开发人员花大量时间阅

读。

-----  
给个四星吧。。。前面几章简直凑字数，讲解parser的实现有什么用啊。。。

-----  
业界良心！

-----  
条分缕析 平易近人 难得的好书

-----  
都是源码分析，太细节了

-----  
比较靠谱的Android架构分析

-----  
值得一看：- 系统native到java层的初始化 - binder - surface  
如果从事音视频相关的工作，audio和mediascanner这两章对于这一类岗位较为有用。  
该书作者的文章绝大部分很多是水文，给Android中级工程师较为合适，高级工程师不建议看。而且很多文章都是来源于博客，没有经过再次打磨，所以对于高级工程师难免难以解困。

-----  
这本书很细致的讲了android的技术实现，从binder到mediaScan，从C到java，不像有的书，只讲架构关系而不关系代码实现、设计模式。

-----  
[深入理解Android\\_下载链接1](#)

## 书评

近来有不少读者向编辑反应： 1 本书很多内容从网上可以搜索到。 2



Binder一章从一个国外网站可以搜索到英文版 (http://unknownerror.net/2011-06/mechanism-of-binder-layman-android-5-30302) 这里要郑重声明一下： 1  
本书的来源：援引【这本书的策划编辑，关于整个误会...

-----  
z.cn上面，关于Android源码分析的有三本书： Android技术内幕:系统卷 - 杨丰盛  
Android内核剖析 - 柯元旦 还有这本书  
关于内核的部分三本书的内容其实都差不多，但坦白说会让你有“一口气读几章”的冲动的就是这一本。真正可以给你一种看到: Binder、驱动、D虚拟机、内核是怎...

-----  
对于linux不是非常了解的人来说，这是一本难得地由浅入深的书，通过代码展现android世界的宏大优美。  
作者对代码的解析非常清晰，加入一些作者自己的理解，然后将这种理解，将这种设计通过简单直白的方式娓娓道来，实在让人解馋!!!  
没有枯燥的概念介绍，没有罗列一堆乱七八...

-----  
怎么说呢，本来是服务端研发，校招进入阿里强行转成Android应用开发，进而又变成Rom适配开发。唉，过程很无奈，这本身也是在Rom适配过程中学习完的。总得来说，一般吧，偏重于对Android Framework和起机流程讲解，对应用开发同学基本没用。

-----  
现在市面很多android内核的书，但大部分都是从网上抄，要不就是翻译英文文档。本人粗读了四本，相较之下，《深》这本书质量还是很高的。大部分内容是作者自己做过实验的。不过下载android源码这部分确实现在已经不能用了，建议更新一下，或者作者自己试试。现在好像卷2也要出...

-----  
序 —— 一把打开Android神秘世界的钥匙  
近2年来，IT行业的最前沿焦点集中到了移动互联网上。PC时代，WINTEL联盟成就了英特尔和微软各自的霸业。移动互联网时代，谁将上演新的传奇？新生的Android用短短的1年多时间挤进了全球智能操作系统的三甲行列(Symbian和iOS分别占据前2)...

-----  
[深入理解Android\\_下载链接1](#)