



2015 移动开发者大会
Mobile Developer Conference China 2015

Android应用开发浅谈

——技术架构视角

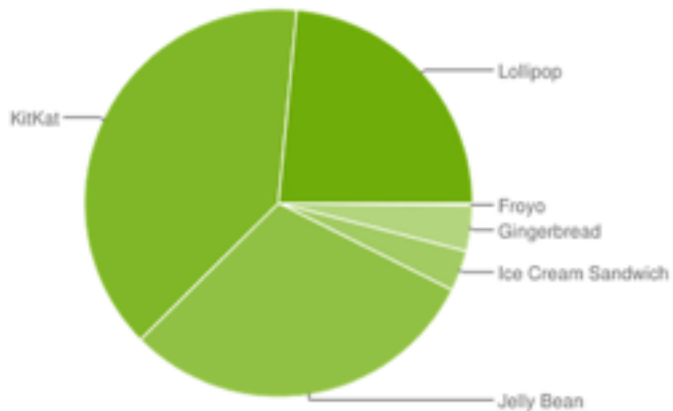
- 关于
 - 涂勇策，百度资深研发工程师，百度手机卫士Android客户端技术负责人。
 - Email: tuyongce@gmail.com
 - Github: <https://github.com/yongce>
- Android开发经历
 - 2010年11月加入创新工场点心OS团队，走上Android开发之路。
 - 主要负责过点心省电、百度手机卫士（原安卓优化大师）等产品的开发工作，在Android应用架构、技术演进、性能优化等方面有不少经验积累。

- **Android发展历程**
- 主要问题及技术方案
- 新产品开发
- 技术演进

- 大事记
- 版本分布

Version	Release date	Notes
1.0 (API Level 1)	2008.09	<ul style="list-style-type: none">• 正式发布的第一个版• HTC G1上市
1.6 (API Level 4)	2009.09	<ul style="list-style-type: none">• 早期开发者开始支持的版本• Support库支持的最低版本
2.1 (API Level 7)	2010.01	<ul style="list-style-type: none">• Nexus One发布• 开发者广泛支持的版本
4.0 (API Level 14)	2011.10	<ul style="list-style-type: none">• 全新设计UI• 同年, Android Support库发布
5.0 (API Level 21)	2014.11	<ul style="list-style-type: none">• Material design• ART代替Dalvik• 支持64位CPU
6.0 (API Level 23)	2015.09	<ul style="list-style-type: none">• 运行时权限• 指纹认证

Version	Codename	API	Distribution
2.2	Froyo	8	0.2%
2.3.3 - 2.3.7	Gingerbread	10	3.8%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	3.4%
4.1.x	Jelly Bean	16	11.4%
4.2.x		17	14.5%
4.3		18	4.3%
4.4	KitKat	19	38.9%
5.0	Lollipop	21	15.6%
5.1		22	7.9%



- ▶ 4.0+, 96%
- ▶ 4.4+, 62%
- ▶ 5.0+, 23.5%

Data collected during a 7-day period ending on October 5, 2015.
Any versions with less than 0.1% distribution are not shown.

- Android发展历程
- 主要问题及技术方案
- 新产品开发
- 技术演进

- 主要问题
 - 性能
 - 产品质量
 - 产品迭代
 - 多进程架构
 - 其它典型问题

问题	技术方案
运行速度	<ul style="list-style-type: none">• 性能分析 (log & Traceview)• 优化UI布局 (Hierarchy Viewer)• 优化算法和数据结构• 业务逻辑调整
内存占用	<ul style="list-style-type: none">• 防止内存泄露• 优化数据结构• 优化图片使用• 多进程架构: 常驻进程与非常驻进程
流量消耗	<ul style="list-style-type: none">• TrafficStats#setThreadStatsTag()• "/proc/net/xt_qtaguid/stats"
电量消耗	<ul style="list-style-type: none">• WakeLock & Wakeup (dumpsys power & batterystats)• 定量分析

问题	技术方案
静态代码检查	<ul style="list-style-type: none">• Android Lint（质量、性能、安全等）• 自定义lint规则• Android Annotations
运行时检查	<ul style="list-style-type: none">• StrictMode（确保仅debug版本）
单元测试	<ul style="list-style-type: none">• 便于回归测试• 便于适配测试
Code Review	<ul style="list-style-type: none">• 代码入库前Code Review• 团队定期集中式Code Review

问题	技术方案
UI改版	<ul style="list-style-type: none">• 制定和实施统一的UI规范• 建立公共UI库
产品策略变更	<ul style="list-style-type: none">• 封装Wrapper类（例如，Toast, Notification）

问题	技术方案
多进程设计	<ul style="list-style-type: none">• 常驻进程• UI进程• 特殊任务进程• IPC
SharedPreferences	<ul style="list-style-type: none">• 实现多进程版“SharedPreferences”• 多进程访问转单进程访问（例如，通过Provider）
数据库	<ul style="list-style-type: none">• ContentProvider• Service
单例对象	<ul style="list-style-type: none">• 允许多个单例对象• 仅一个单例对象，通过Service访问

问题	技术方案
APK包大小	<ul style="list-style-type: none">• 删除冗余资源（Android Lint or "shrinkResources"）• 功能删减• 插件化，按需下载功能模块
方法数	<ul style="list-style-type: none">• 代码精减• multidex• 插件化
文件句柄	<ul style="list-style-type: none">• WebView：独立进程，按需创建• 小心调用Zip相关API
数据库操作	<ul style="list-style-type: none">• 单进程：引用计数方式打开/关闭数据库• 多进程：ContentProvider
系统API封装	<ul style="list-style-type: none">• 系统兼容性（例如，HTTP请求）• 安全问题（例如，广播发送/接收，Intent extra参数提取）

- Android发展历程
- 主要问题及技术方案
- 新产品开发
- 技术演进

- 研发流程管理
- 技术架构与选型

问题	技术方案
源码管理	<ul style="list-style-type: none">• 版本控制（例如，git）• Code Review系统（例如，gerrit、gitlab/PR）• 源码版本Tag管理
开发环境与构建系统	<ul style="list-style-type: none">• 操作系统（Mac > Linux > Windows）• IDE & 编码规范（例如，Android Studio）• 构建系统 & CI（例如，Gradle & Jenkins）
产品运营支持	<ul style="list-style-type: none">• 数据统计（包括Crash统计）• 应用版本号管理• 渠道号管理• 应用签名与打包系统
其它	<ul style="list-style-type: none">• Bug跟踪系统• 测试用例管理系统• 需求管理系统

问题	技术方案
Android版本	<ul style="list-style-type: none">• 确定minSdkVersion和targetSdkVersion
反逆向分析	<ul style="list-style-type: none">• 代码混淆• 资源混淆• 代码加壳
基础代码库	<ul style="list-style-type: none">• 工具类• 基础能力封装• Wrapper封装，便于维护和演进
公共UI库	<ul style="list-style-type: none">• 制定和实施统一的UI规范• UI改版• 附属产品的UI统一和重用

- Android发展历程
- 主要问题及技术方案
- 新产品开发
- 技术演进

- 基本原则
- 技术跟踪

- 中长期技术规划
 - 产品发展规划和技术特点
 - 技术跟踪和趋势判研
- 多做小重构
 - 维护架构的可扩展性
 - 为技术架构变迁提前准备
- 不做大重构

关注点	目标	建议
Android版本及SDK	新特性和技术的价值挖掘； 及时发现并解决产品的兼容性问题。	<ul style="list-style-type: none"> • 新的系统特性和API • 系统行为和API的改变
Android Support库	了解重要API及使用场景	<ul style="list-style-type: none"> • v4/v7 Support Library • Annotations Support Library • Design Support Library
Android Studio	关注新特性和内置工具，提升开发效率和质量。	<ul style="list-style-type: none"> • http://tools.android.com/recent
Android Plugin for Gradle	关注新特性，适时引入项目，优化构建流程。	<ul style="list-style-type: none"> • http://tools.android.com/tech-docs/new-build-system
性能优化工具	优化产品的各性能指标（运行速度、内存占用、流量消耗、电量消耗）。	<ul style="list-style-type: none"> • http://developer.android.com/tools/performance/index.html • 开发性能监控和分析工具
质量工具	引入适当的质量工具，提高代码质量。	<ul style="list-style-type: none"> • Android Lint • Testing Support Library

问题	参考资料
Google Play统计的Android版本分布	<ul style="list-style-type: none">• http://developer.android.com/about/dashboards/index.html
文件句柄用尽问题研究	<ul style="list-style-type: none">• https://github.com/dxopt/OpenFilesLeakTest
自定义lint规则	<ul style="list-style-type: none">• https://github.com/yongce/AndroidDevNotes/blob/master/notes/knowledge/0005-custom-lint.asc
Android Annotations	<ul style="list-style-type: none">• https://github.com/yongce/AndroidDevNotes/blob/master/notes/knowledge/0001-support-annotations.asc
数据库Wrapper	<ul style="list-style-type: none">• https://github.com/yongce/AndroidLib/tree/master/lib/src/main/java/me/ycdev/android/lib/common/dbmgr



2015 移动开发者大会

Mobile Developer Conference China 2015

谢谢！

mdcc.csdn.net