



# 如何开发一款优雅的 **SDK**



个推：吕观祥

2016年9月24日

- 为企业和开发者提供消息推送技术服务

- 提供 Android 和 iOS 推送 SDK

- 支持 WEB 及服务器端推送 API 接入

- 使 APP 快速集成云推送功能，免去开发成本

- 有效提高产品活跃度与用户体验



## 实时推送



强大技术支撑  
推送实时高效触达

## 智能推送

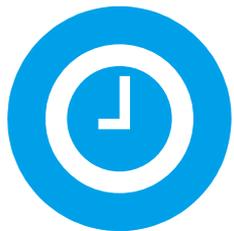


智能匹配  
数据驱动，高效运营

## 应景推送



利用 LBS 围栏技术  
实时触发消息通知



合适的时间

定时推送



合适的地点

地理围栏



合适的场景

智能场景  
标签



合适的内容

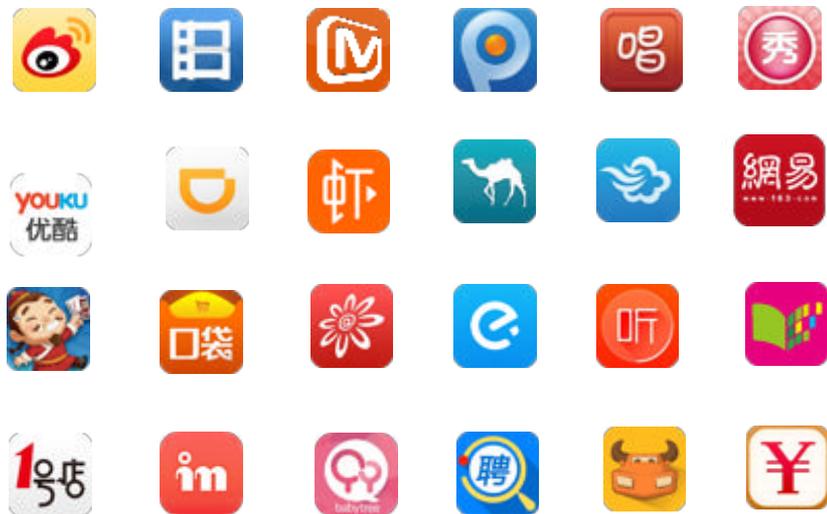
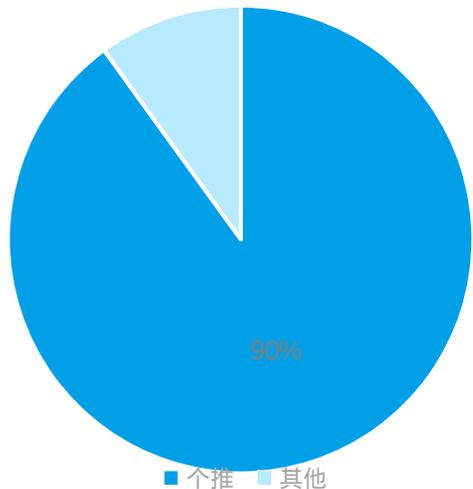
有价值信息



合适的人

智能画像  
标签

个推所占第三方推送市场份额



- SDK 累积安装覆盖量超过 108 亿，覆盖超过 10 亿终端（含海外），已接入应用 43 万，开发者 22 万，日活用户近 7.5 亿，同时在线 3.5 亿。

# 目录



开发



CI



发布



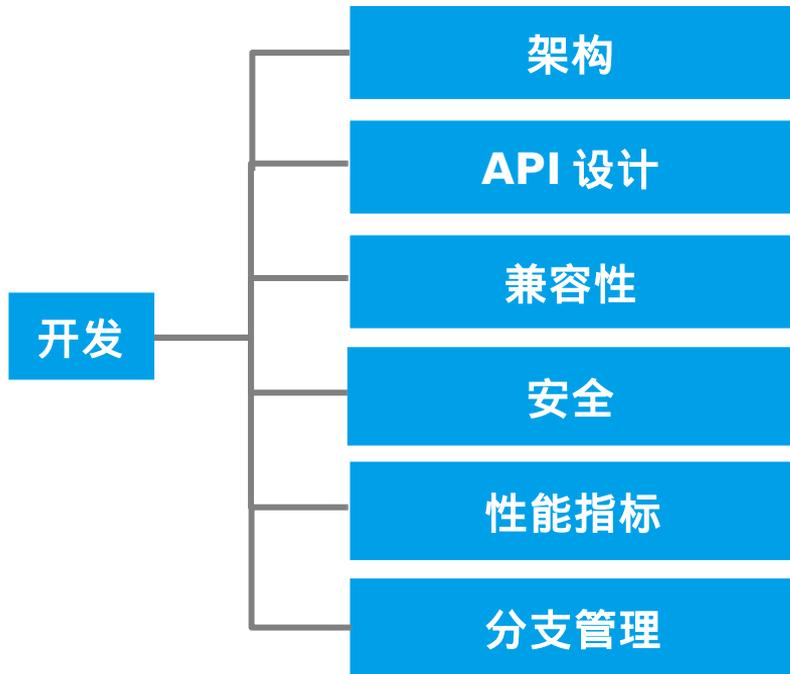
其他

开发

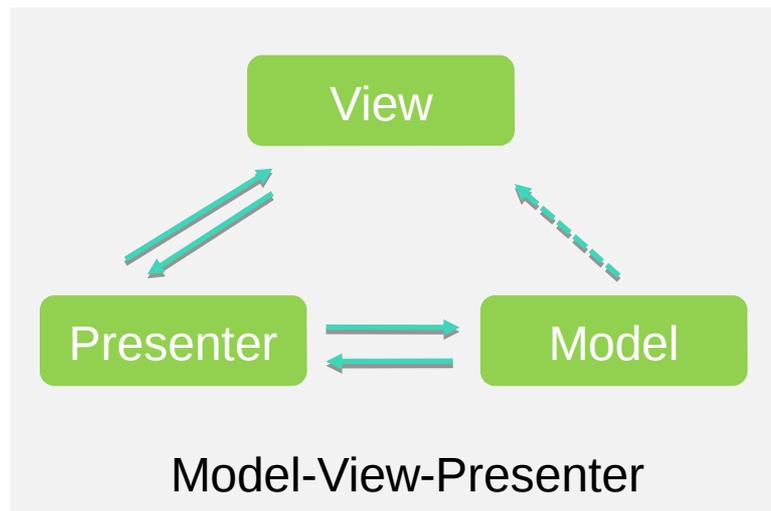
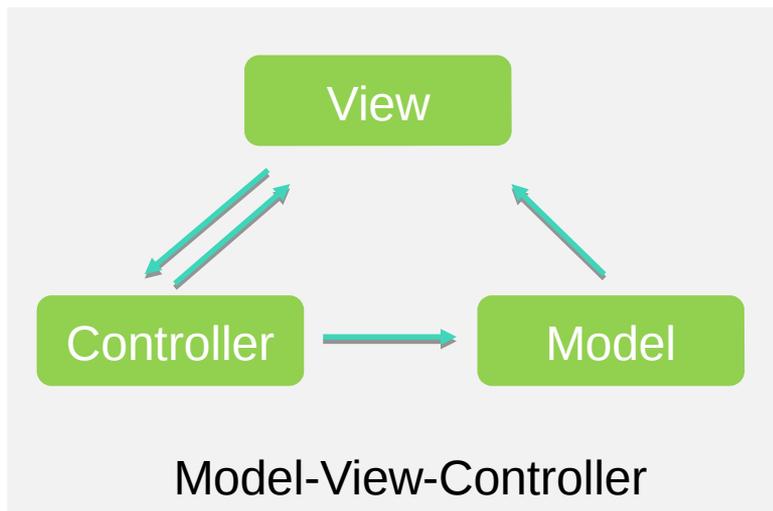
CI

发布

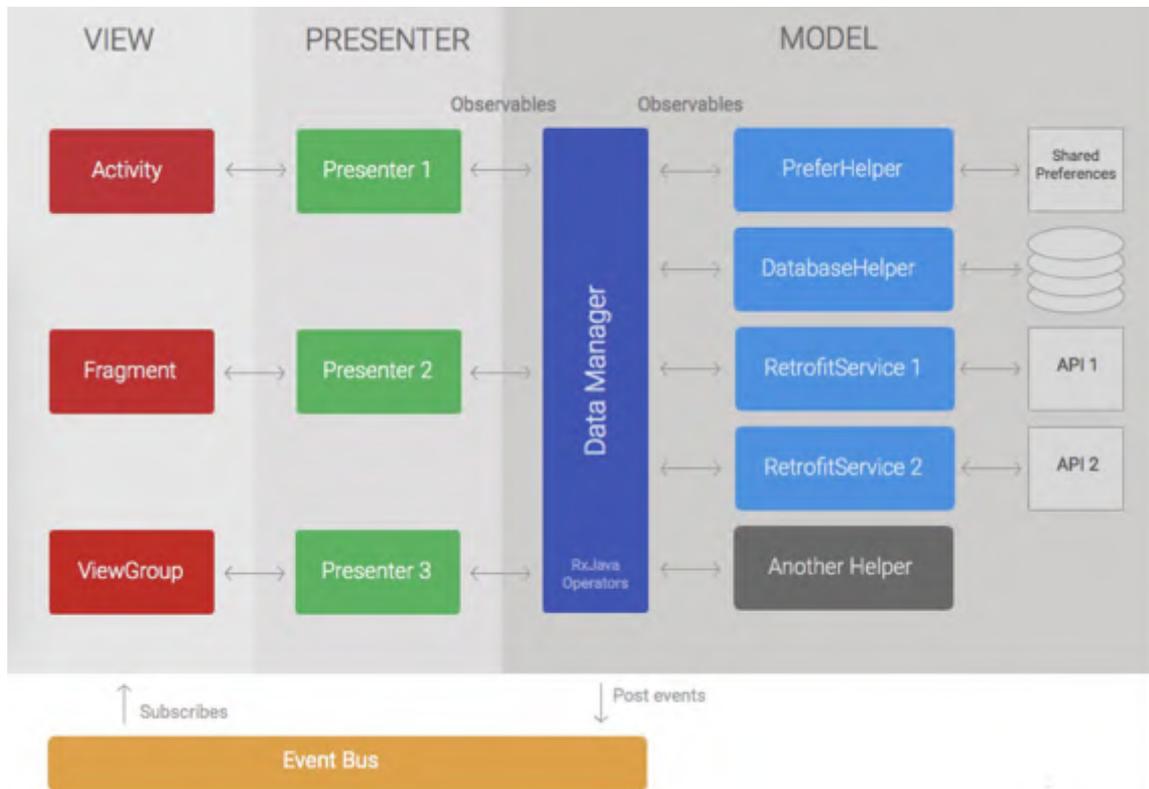
其他



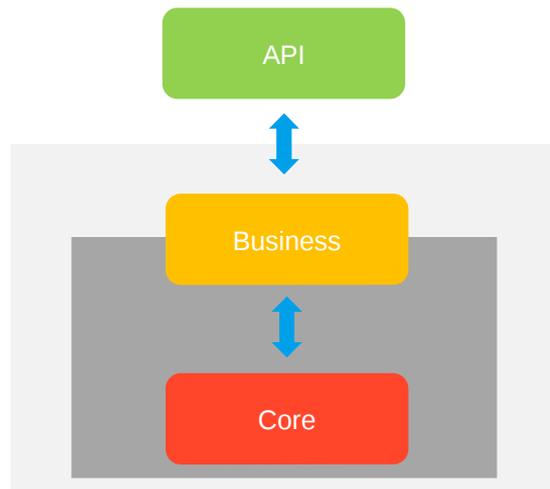
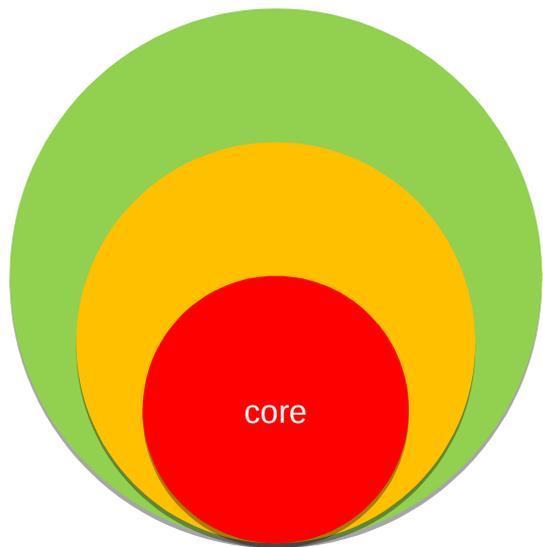
## 架构设计 -- APP



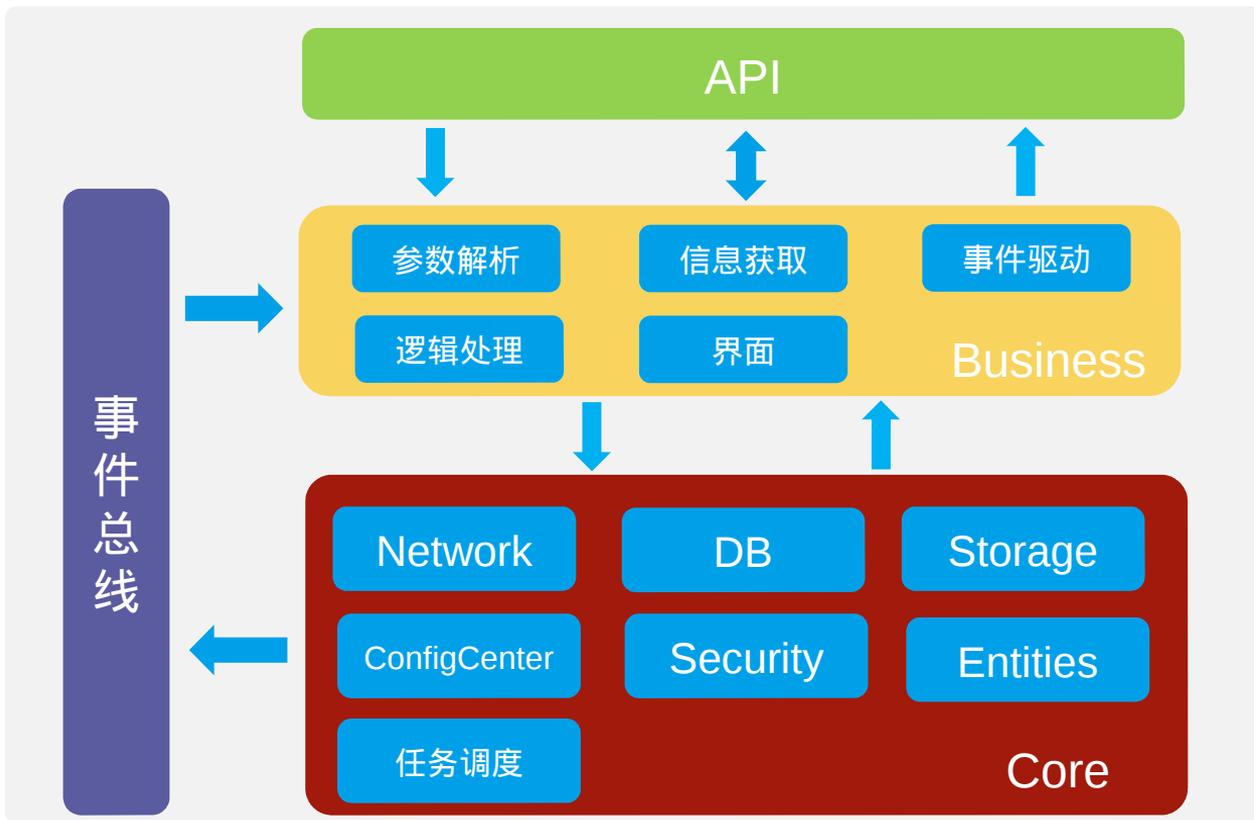
# 架构设计 -- APP



# 架构设计 -- SDK



## 架构设计 -- SDK



## 接口设计

### 易用性：

- 接入简单
- 可自解释
- 可定制
- 防止误用

```
LoggerFactory.init(this);
```

```
LoggerConfigurator.Builder builder = LoggerConfigurator.newBuilder();  
builder.setDefaultLoggerName("logger");  
builder.setDefaultMsgLayout("s, sendMessage, %s | g, getMessage, %s | r, result, %n | b, back, %n | c, call, %n");  
builder.setCaughtException(true);  
builder.setCaughtNativeCrash(true);  
builder.setCaughtANRError(true);  
  
LoggerFactory.init(this, builder.build());
```

## 接口设计

### 稳定性：

- 生命周期
  - 弃用 ( @Deprecated )
  - 扩展
- 异常处理

## 兼容性



## 安全性 -- 本地数据安全

范围	
1	storage
2	shared preference
3	db
4	resource
5	memory
6	...

### 实施：

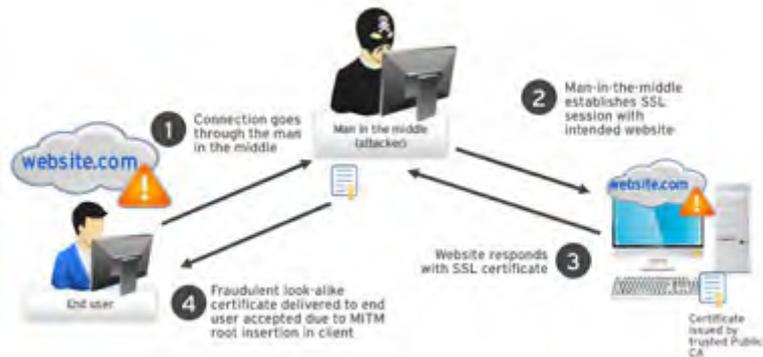
- JNI
- 加密
- 内存数据处理

# 安全性 -- 通信数据安全

明文

No.	Time	Source	Destination	Protocol	Length	Info
218	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
219	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
220	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
221	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
222	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
223	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
224	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
225	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
226	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
227	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
228	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
229	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
230	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
231	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
232	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
233	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
234	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
235	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
236	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0
237	0.000000	192.168.0.1	192.168.0.42	TCP	60	64849 → 80 [ACK] Seq=1000000000 Win=0 Len=0

中间人攻击



实施：

- HTTPS
- ECDHE-AES-GCM

## 性能指标

### 电量 & 流量

- 批处理请求
  - 优化协议
  - 压缩数据
- 心跳优化处理
- WakeLock & WifiLock

### 内存

- 多进程处理
- 优化图片使用

### 包大小

- 精简 resource / code
- raw/keep.xml



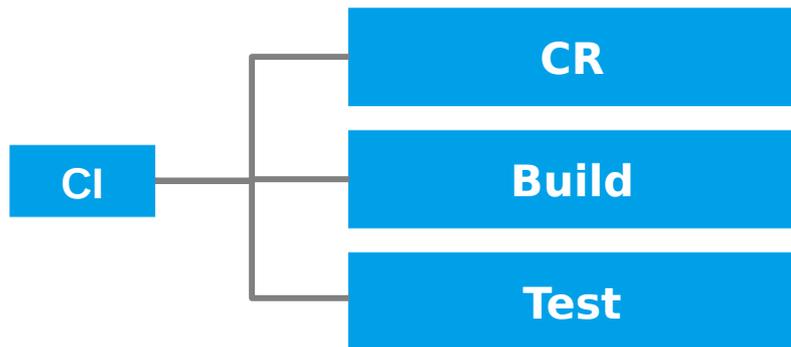


开发

CI

发布

其他



## Code Review

### 特点：

- 需求多分支兼容
- Bug Fix

### 工具：



sonarqube



## BUILD

### 特点：

- 多 demo 打包（包含不同配置的 demo 包）
- 提供 apk 和 jar ， aar 包供测试需求

## 测试

特点：

- 自动化测试
- 人工测试

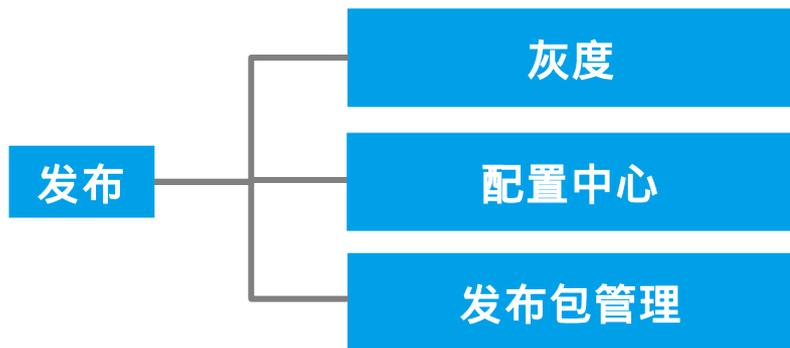


开发

CI

发布

其他



## 灰度发布

用户选择：

- 测试 app 选择
- 目标用户选择



## 灰度发布

### 工具：

- Logful
  - Android/iOS 支持
  - 日志加密存储和上传
  - 日志检索
  - 实时日志控制
  - Crash 捕获
  - 数据监控，报警



## 灰度发布

## 延时数据监控

### Alerts configuration for stream

You can define thresholds on any message field or message count of a stream and be alerted based on this definition.

[Learn more about alerts in the documentation.](#)

#### Add new alert condition

Message count condition [Configure new alert condition](#)

#### Configured alert conditions

**Field value condition**  
Alert is triggered when the field `col_gloose` has a higher mean value than 10000 in the last 5 minutes. Grace period: 1 minute. Including last 5 messages in alert notification.

#### Callbacks

The following callbacks will be performed when this stream triggers an alert.

Email Alert Callback [Add callback](#) [Find more callbacks](#)

**Email Alert Callback**  
Executed once per triggered alert condition.



## 报警通知

## 灰度发布

## Crash 趋势



error thread all thread log reason more

```

java.lang.OutOfMemoryError
Failed to allocate a 28 byte allocation with 4000 free bytes and 3KB until OOM

java.util.LinkedList$ListIterator(LinkedList.java:627)
java.util.AbstractSequentialList$Iterator(AbstractSequentialList.java:62)
rx.internal.util.s.a(s.java:122)
rx.internal.util.s.unsubscribe(s.java:113)
rx.internal.c.h.unsubscribe(h.java:80)
rx.internal.c.h.run(h.java:64)
  
```

## Crash 详情

## 配置中心

### 多维度配置



## 发布包管理

### Binary 打包：

- 提供 aar 包
- 多种构建工具的支持
  - Maven
  - Jcenter
  - 私有仓库

## 发布包管理

### 内容管理：

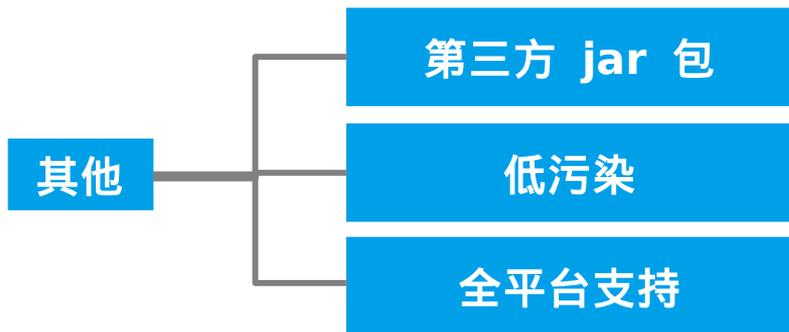
- 更新日志
- Sample 工程代码
- Demo APK
- 文档
- Jar ， aar 下载，接入方式
- 社区支持

开发

CI

发布

其他



## 其他

### 第三方 jar 包：

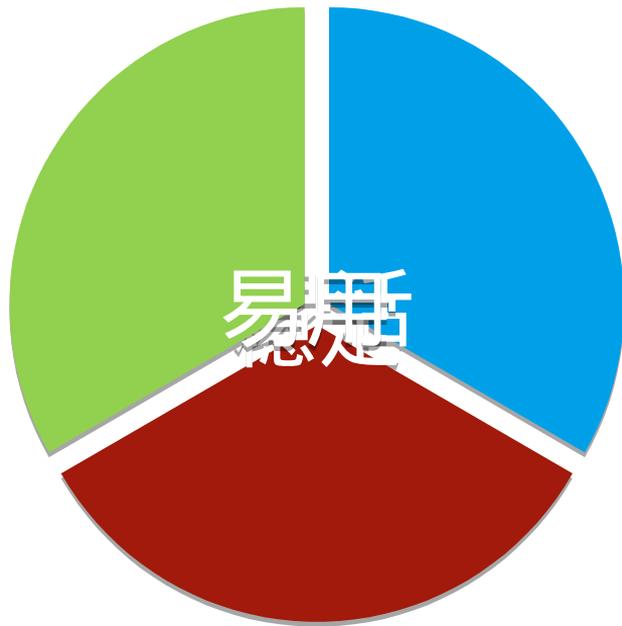
- 使用冲突问题
- Classpath 检测

### 低污染：

- 及时清理运行过程中创建的临时文件

### 全平台支持：

- Cocos2dx ， unity3d ， APICloud ， PhoneGap 等等



# Thanks !

@ 个推



@xiangzi-

