

LiveVideoStackCon

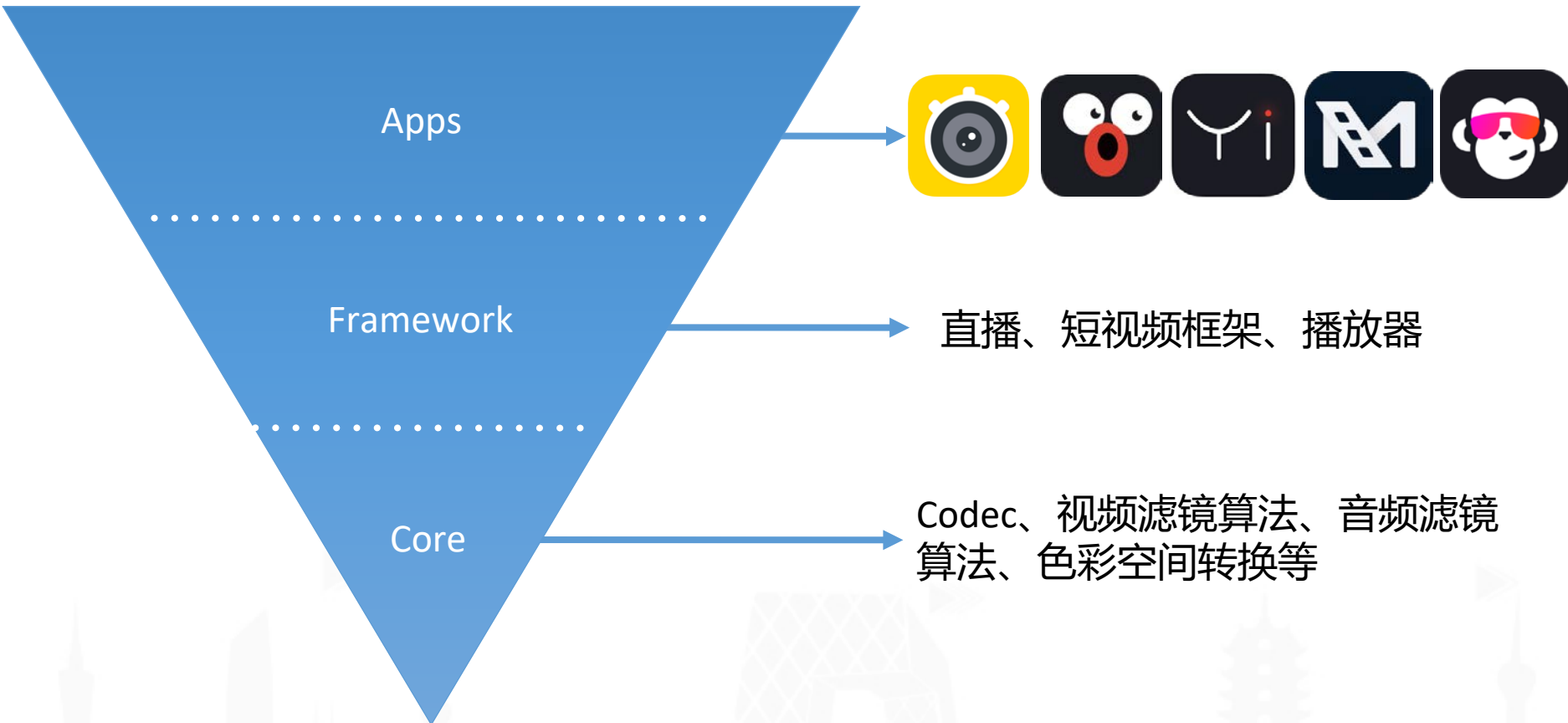
短视频框架设计和关键性技术实践

2017年10月20日-21日

北京·丽亭华苑酒店

- 短视频框架
- 关键性技术应用实践





硬件编解码应用

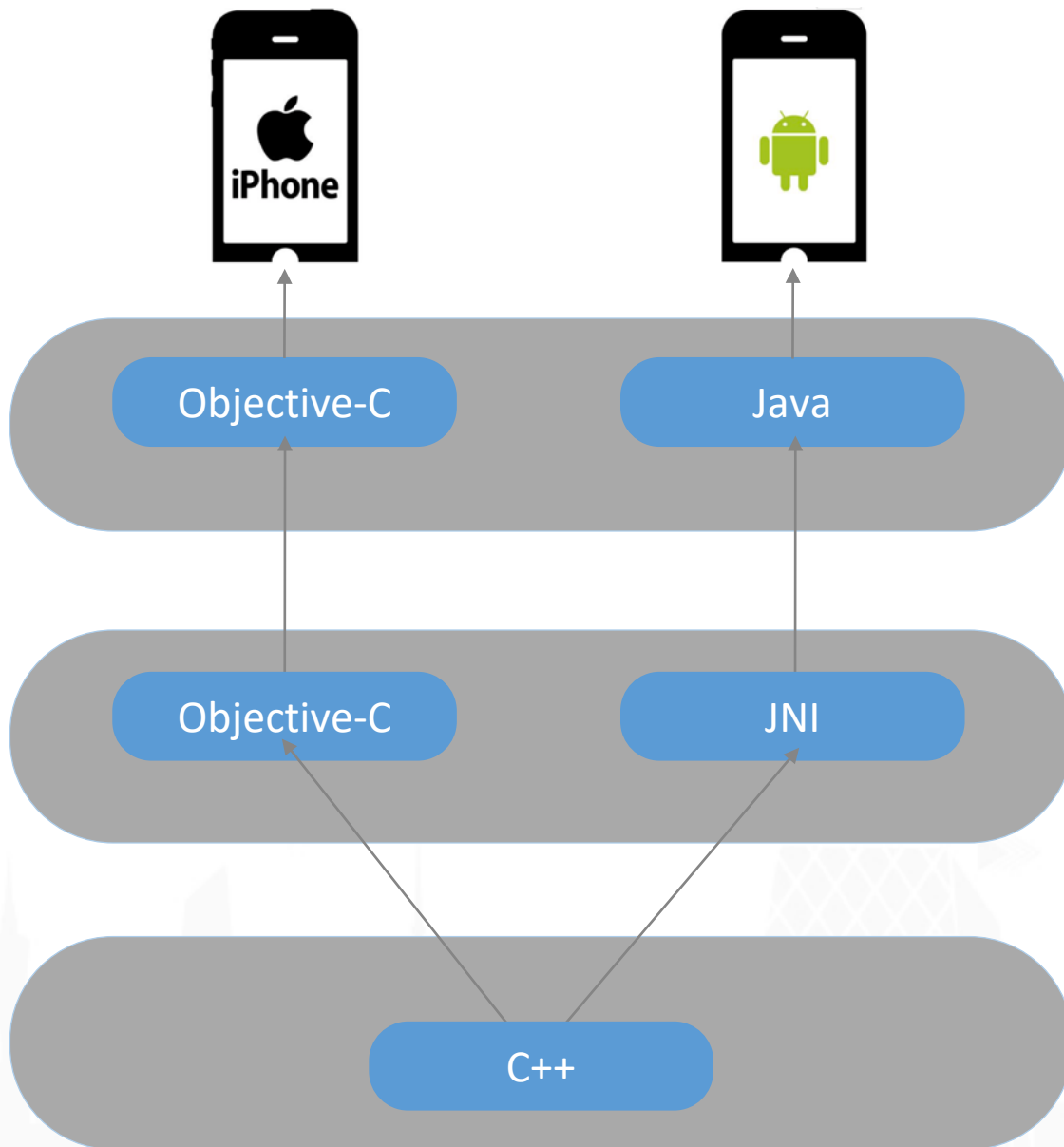
通用性滤镜框架设计

Android低端机处理

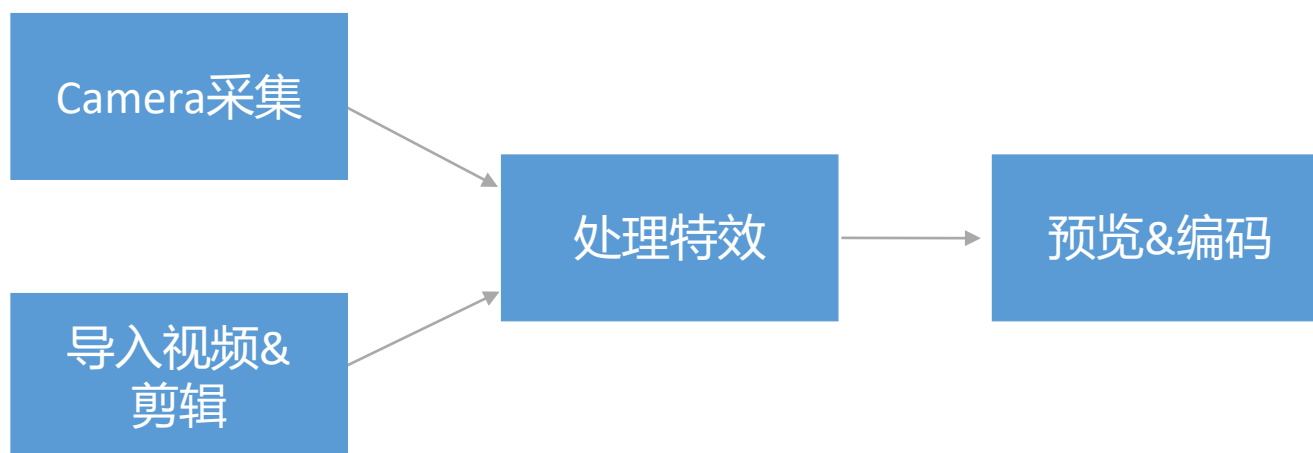
▶ 1.1 短视频框架：项目背景



1.2 短视频框架：跨平台考虑

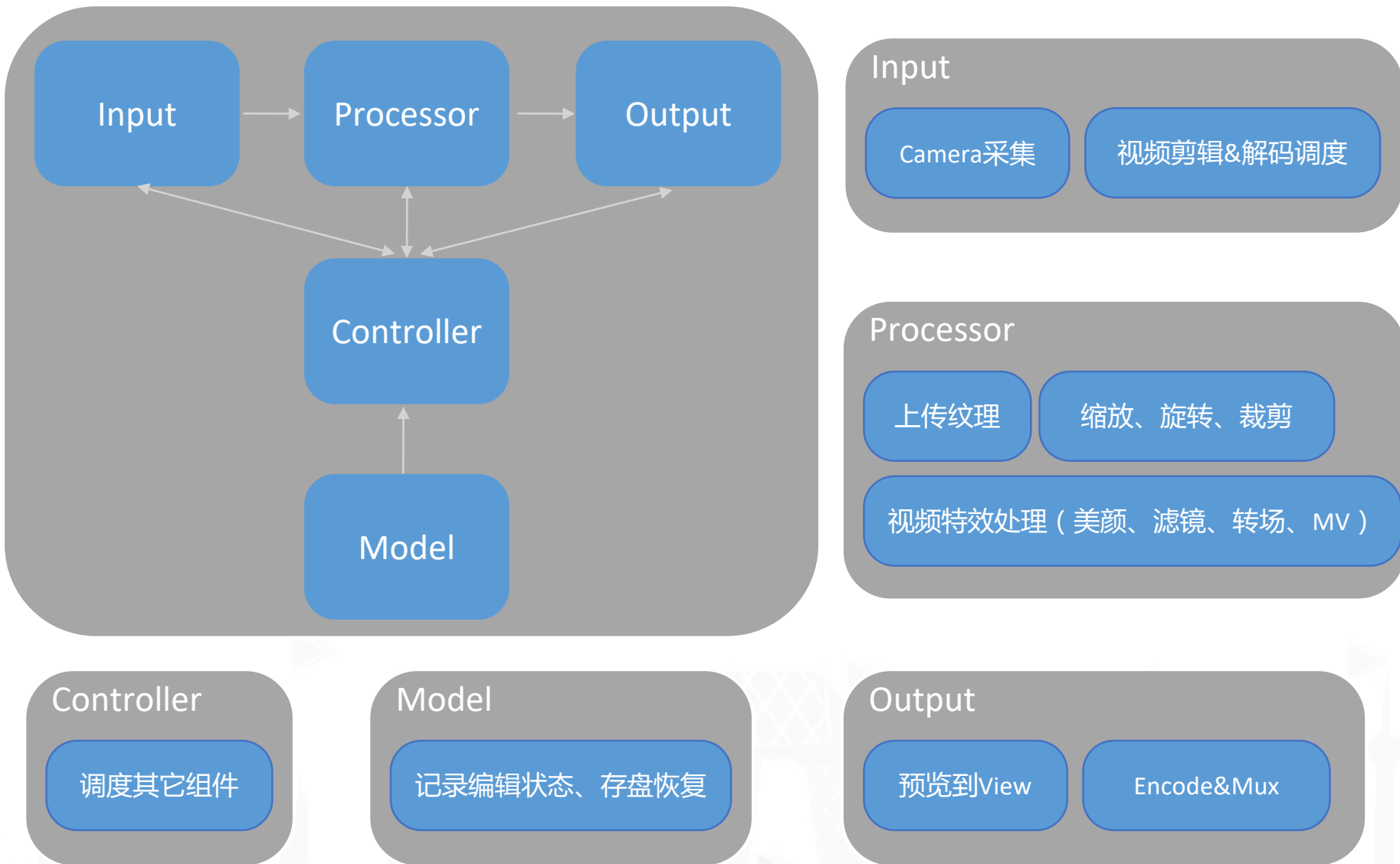


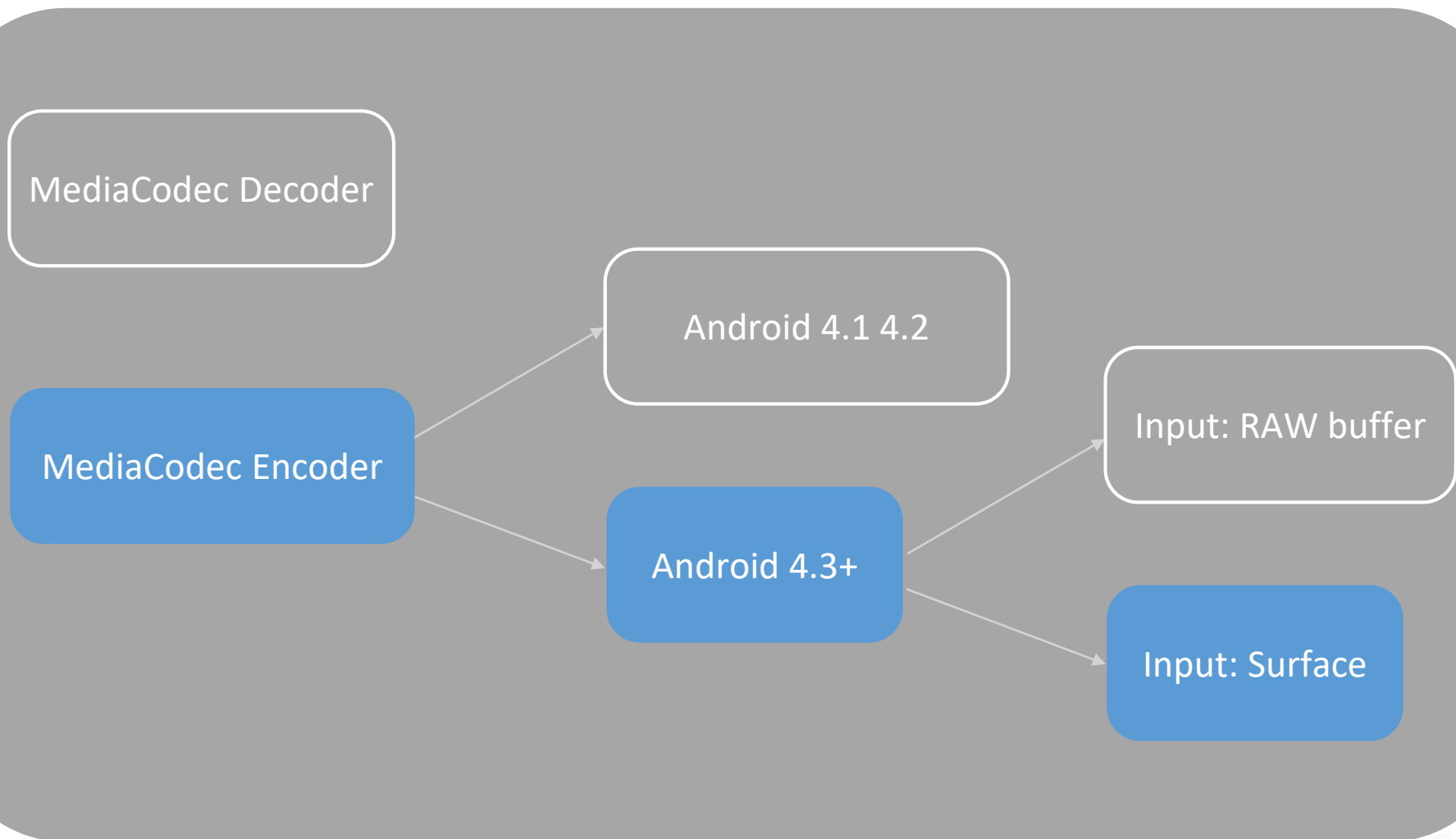
- C++部分是跨平台代码，包括：
- 剪辑状态描述
- 引擎调度
- 软件编解码
- 视频特技，OpenGL代码
- 音频特技

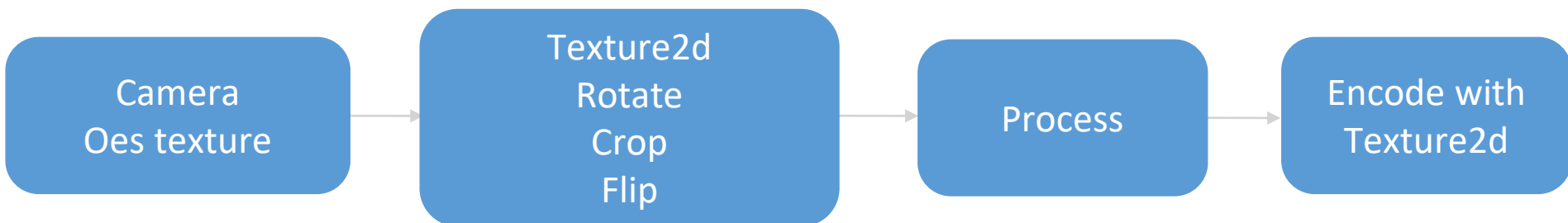


音频也是同样的处理流程

1.4 短视频框架：视频workflow组件化设计







- iOS8以上，VideoToolBox
- Android4.3以上，MediaCodec

编码器对比：

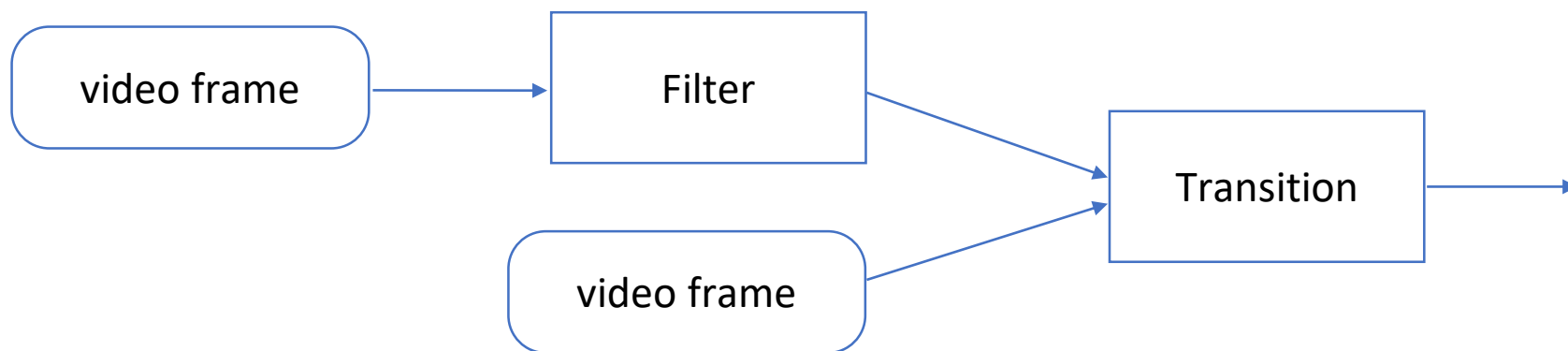
	输入	输出	Muxer	优点
FFmpeg	RAW buffer	H264	FFmpeg mp4 muxer	稳定性好，能保证编码画面质量
MediaCodec	Surface	H264	H264->AVPacket FFmpeg mp4 muxer	编码速度快，CPU占用低

解码器对比：

	Demuxer	输入	输出	优点
FFmpeg	FFmpeg mp4 demuxer	H264	RAW buffer	稳定性好，首帧解码时间快，能开多实例
MediaCodec	FFmpeg mp4 demuxer AVPacket->H264	H264	Surface	高清情况下解码速度快，CPU占用低

- 编码器：只能编码Baseline，所以压缩比和画面损失比较大。但是编码速度提升比较大，推荐使用。
- 解码器：首帧解码时间不稳定，并且有的设备最多只能开2个实例。除非有好的预加载策略或者考虑高清或者H265，否则不推荐。
- 编码器稳定性控制：黑名单、线上软硬件编码切换开关

Graph处理流程



- Android手机性能差距很大，为了在不同用户的手机上都能保证处理的流畅度，我们做了一些分档规则：
 - 1 不同应用场景对手机的性能要求不同，基本上就是CPU核数、频率、内存、GPU型号。
 - 2 分档原则来源于两方面，一方面是我们测试和用户反馈的结果；另一方面是关键数据上报出BI报表进行分析得出。

Thank You

解晶：18510632546

xiejing@yixia.com



LiveVideoStackCon

聚音视 研修不止于形



关注LiveVideoStack公众号

回复 **解晶** 为讲师评分