

1-播放器项目分析

1 为什么要学习播放器开发

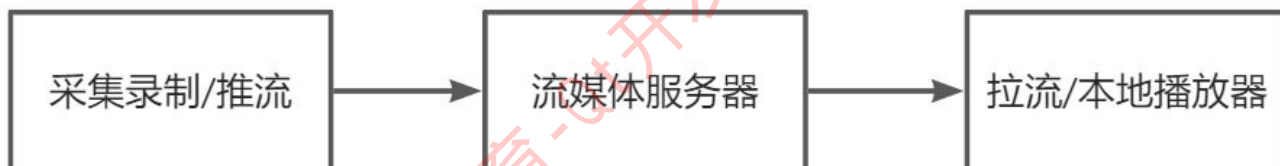
2 学习播放器开发的前置知识

3 我们该如何掌握一个商业的播放器

腾讯课堂《FFmpeg/WebRTC/RTMP音视频流媒体高级开发》<https://ke.qq.com/course/468797?tuin=137bb271>

1 为什么要学习播放器开发

对于音视频应用而言，主要的核心流程为：



即是

1. 采集录制/推流：比如采集麦克风+摄像头，进行编码后保存为MP4文件，或者通过RTMP协议推送到流媒体服务器；在后续的录制推流项目会详细讲解。
2. 流媒体服务器承载RTMP推拉流等，或者其他流媒体协议比如HLS/HTTP-FLV的转换；在srs项目详细讲解。
3. 拉流/本地播放。这个是我们当前项目要重点讲解的。

2 学习播放器开发的前置知识

学习本章节的播放器，需要先掌握ffplay.c。

本章播放器分为两个阶段：

1. 第一阶段，基于ffplay.c进行二次开发，并参考ijkplayer的逻辑流程。 **这属于核心章节，老师课堂上会精讲。**
2. 第二阶段，吸收QMPlay2的一些优秀设计。 **这属于选修章节，老师主要提供思路和技术支持。**

3 我们该如何掌握一个商业的播放器

我们为什么这个播放器是参考ffplay.c、ijkplayer和QMPlay2，是因为做一个商业的播放器，如果完全自己从0开始会走很多弯路，比如：

1. UI界面如何和播放器核心进行解耦，能百度到的简单demo播放器UI和播放器核心都是耦合在一起的，这个时候我们可以参考ijkplayer的消息机制
2. 再比如我们设计了消息机制进行UI界面和播放器核心的解耦，那如果自己完全写代码是否会想到每个状态转换之前是否会做状态能否转换的检测，比如 从IDE状态，能否直接转到START播放状态。
3. 还有我们反复得发暂停、恢复播放，消息队列里是否有存储多个暂停、播放指令的必要，而对于成熟的播放器消息队列，他会把消息队列里有冲突的消息指令删除，在发送新的消息指令，比如刚按了暂停，然后又要恢复播放，如果发送恢复播放的时候队列里有暂停指令，**那这个暂停指令就没有必要被继续执行，可以将其删除。**
4. 从上分析也可以得到UI界面和播放器核心直接的消息队列，是存在两个线程的竞争关系，此时消息队列是不是要设计为线程安全。
5. 以上等等分析，我们都可以借鉴成熟的商业播放器，站着前人的基础上提升。

对于参考ijkplayer，建议大家可以用vscode或者sourceinsight加载其源码，可以跟踪其源码。

在学习的过程中，一定要多思考有些逻辑为什么这么设计，课程上老师没有讲到的点，如果自己不理解，可以多在qq群或者找老师一对一沟通。