

LiveVideoStackCon

# 播 放 器

2017年10月20日-21日

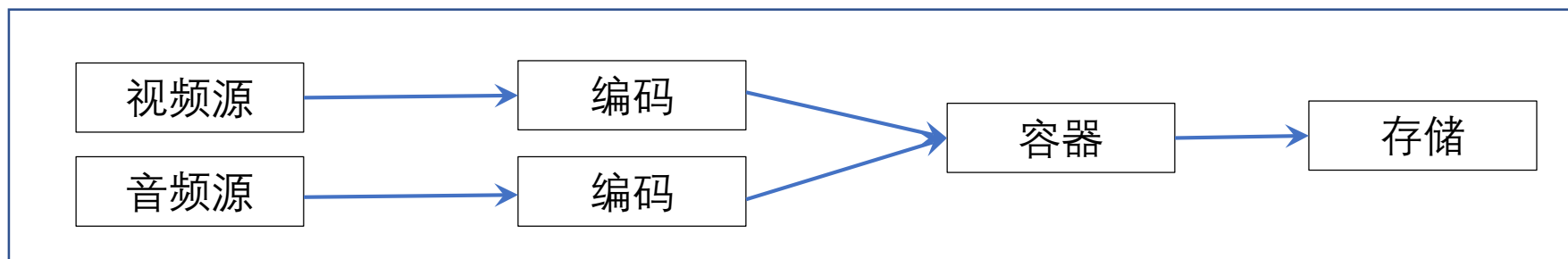
北京.丽亭华苑酒店

## 提 纲

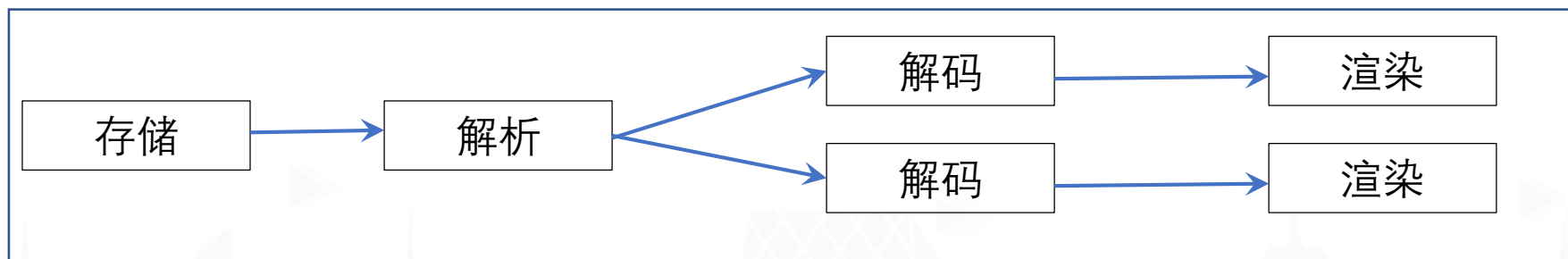
1. 播放器原理。
2. 音视频同步机制。
3. 模块化设计。
4. 分层设计。
5. 数据驱动。
6. 自我诊断。
7. 播放器实例。

## 播放器原理

视频录制



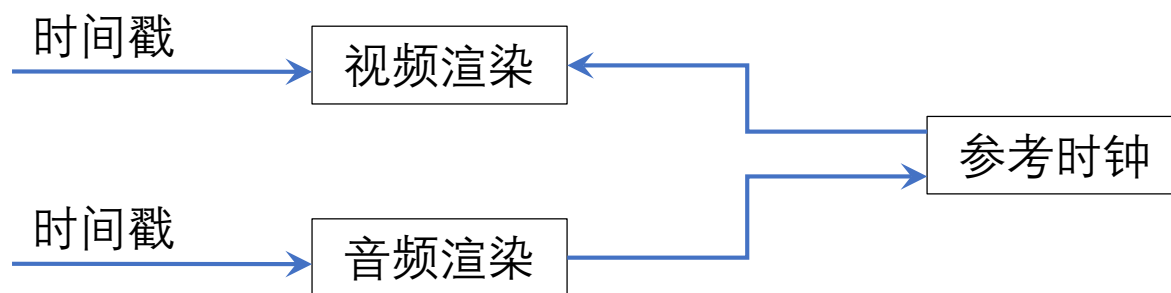
视频播放



所以播放器原理就是还原原来录制的内容。

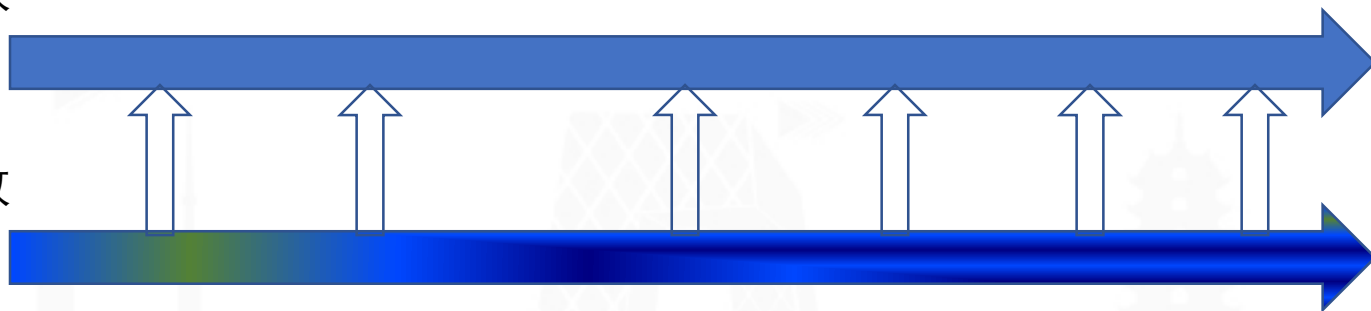
## 音视频同步机制

音视频同步

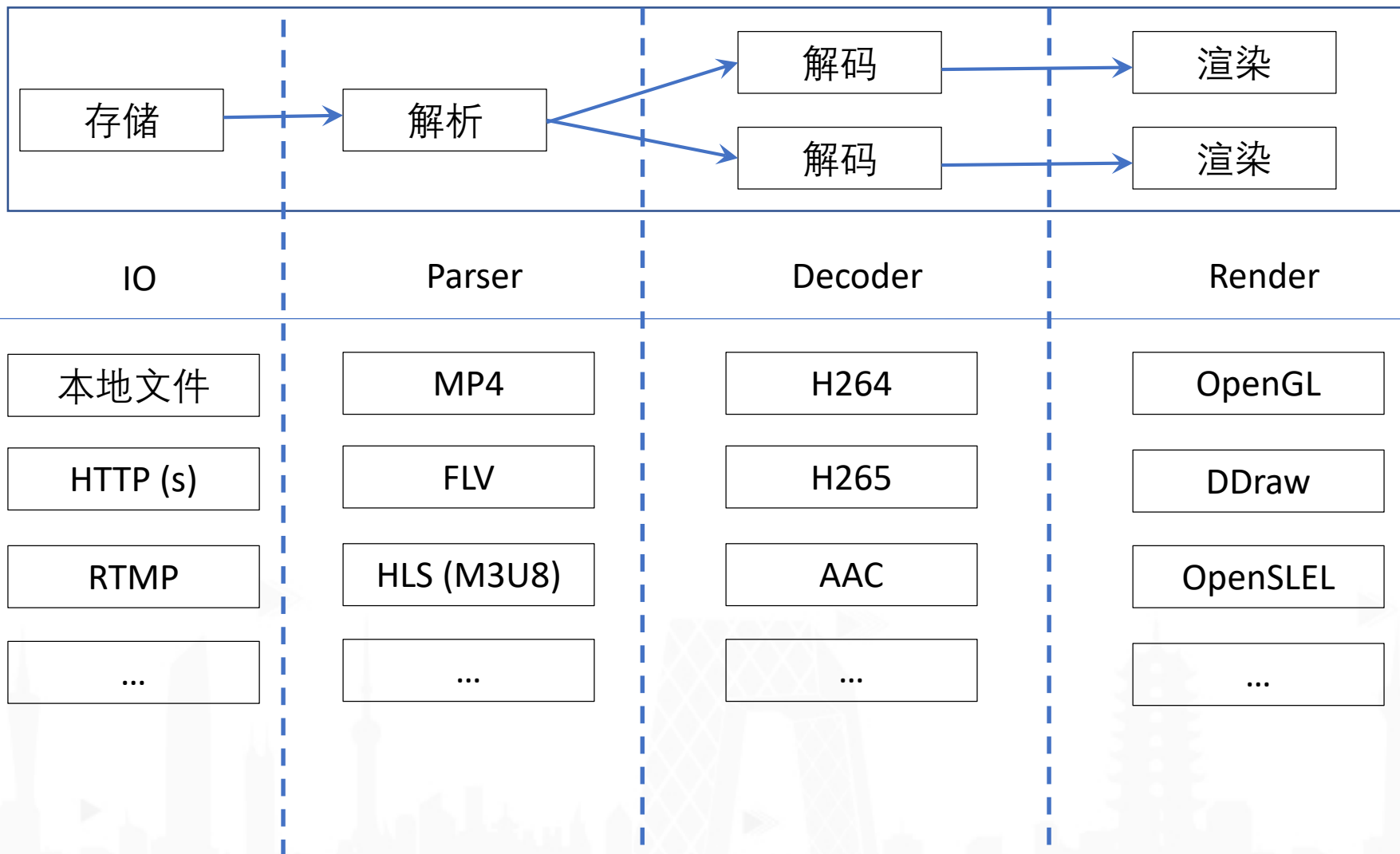


音频播放

视频播放



## 模块化设计

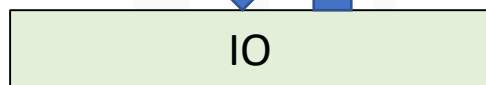
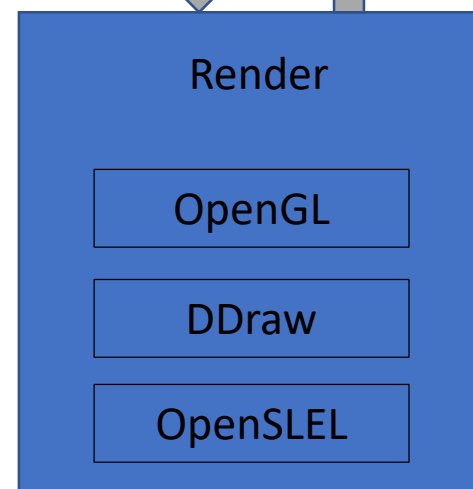
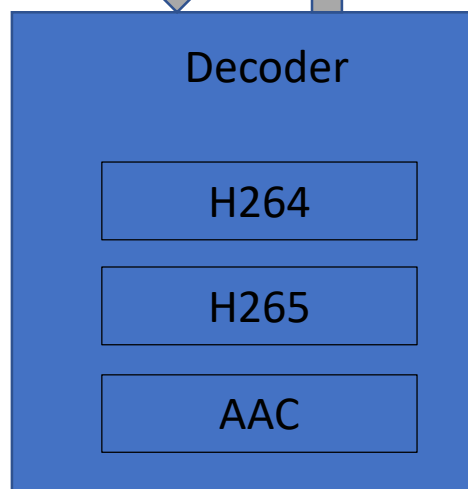
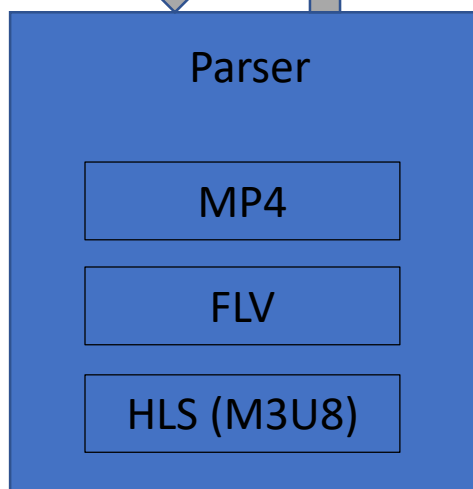


## 分层设计

控制层

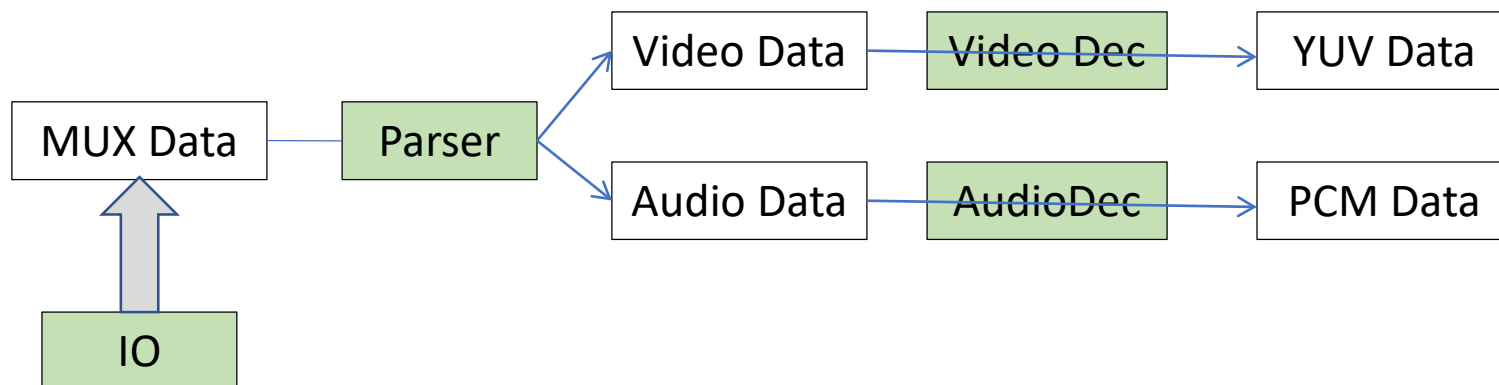


数据层



## 数据驱动

数据流动，变化过程



播放器播放过程其实就是一个把数据在不同的地方变成另外一种格式的数据，并加以控制的过程。所以我们可以通过数据来驱动各种模块，来获得自己需要的格式数据。

## 自我诊断

数据在每一个地方都需要有正确的格式。这样我们就可以在各个模块里面加入对数据进行检查，判断是否是正确有效的数据。如果有错误，输出信息，实时了解数据流动情况。

诊断有两个部分。一个是内容，一个是时间。

内容在模块里面进行检查。时间在控制层进行检查。自我诊断的好处在于可以很快判断出哪里出了问题。



## 播放器实例

### 简单

Audio Video Decoder : 2.0 M。 (可以用硬解)  
MFW, Parser, Render, IO : 0.3 M。

### 首开时间

#### 直播 :

RTMP : 200 毫秒。  
HTTP FLV : 200 毫秒。  
HLS : 500 毫秒。

#### 短视频 :

MP4 : 150 毫秒。

[Demo](#)

## 特色功能

### 一、变速不变调：

网络有时候是不稳定的，在用户没有感觉的情况下，对声音进行变速，进行缓冲或者加速，保持播放的平滑，也保证不会出现延迟。

### 二、快开模式：

在同一格式源，采用快开模式，加快首开速度。

### 三、动态管道：

支持视频，音频随时创建，格式变化。

### 四、文件缓存：

采用了文件分段保存，这样用户在播放的时候seek，也可以保存播放过的内容。为客户，用户节约流量。

### 五、双IO连接：

对MP4文件的信息部分在最后的，采用双IO连接，可以快速首开，不浪费流量。

## 总结

有一句话这样说的，大道理大家都懂，就是很难做的好。播放器也是这样，原理大家都了解，可是很难做到极致。所以很多地方就是靠从细节地方，一个一个的进行优化，这个时候经验就很有用，因为有了经验，你就知道怎么的优化是最有效率的。



# Thank You

金邦飞 : 18621962112

Jin\_bangfei@163.com



LiveVideoStackCon

聚音视 研修不止于形



关注LiveVideoStack公众号

回复 **金邦飞** 为讲师评分