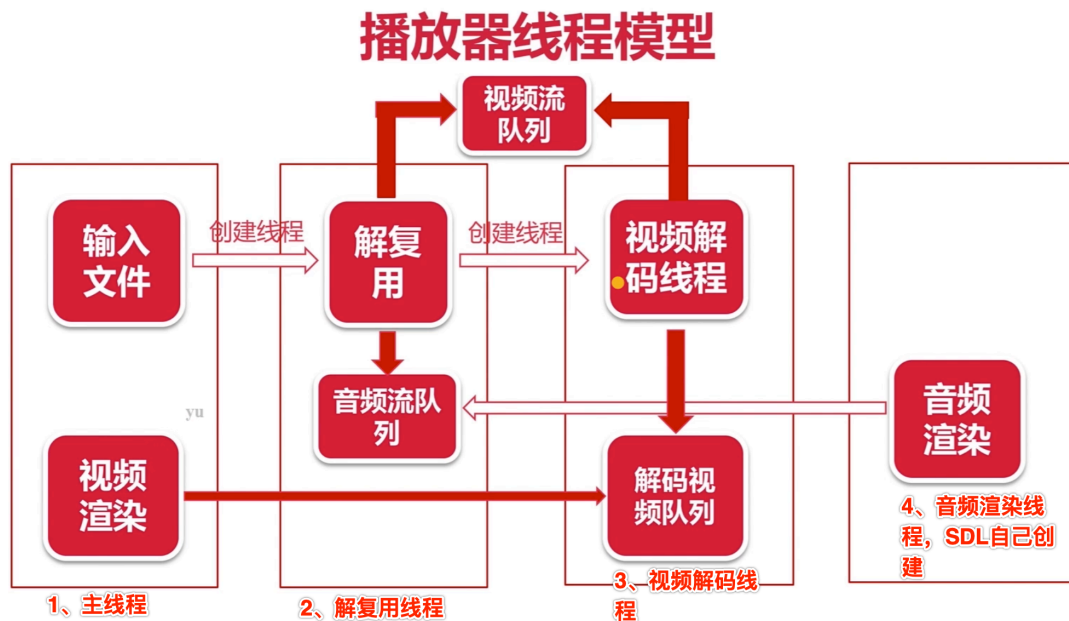


FFmpeg 复习第三天

```
superleadeMacBook-Pro:sd1 superlea$ clang -g -o player2 player2.c `pkg-config --cflags --libs sdl2 libavformat libavutil libswscale` clang命令
```

真课网



线程的退出机制

- ◆ 主线程接收到退出事件
- ◆ 解复用线程在循环分流时对 quit 进行判断
- ◆ 视频解码线程从视频流队列中取包时对 quit 进行判断

线程的退出机制

- ◆ 音视解码从音频流队列中取包时对 quit 进行判断
- ◆ 音视循环解码时对 quit 进行判断
- ◆ 在收到信号变量消息时对 quit 进行判断

时间戳

- ◆ PTS : Presentation timestamp
- ◆ DTS : Decoding timestamp
- ◆ I (intra) / B (bidirectional) / P (predicted) 帧

时间戳顺序

实际帧顺序 : I B B P

存放帧顺序 : I P B B

解码时间戳 : 1 4 2 3

展示时间戳 : 1 2 3 4

从哪儿获得PTS

- ◆ AVPacket 中的 PTS
- ◆ AVFrame 中的 PTS
- ◆ `av_frame_get_best_effort_timestamp()`

时间基

- ◆ tbr: 帧率
- ◆ tbn: time base of stream
- ◆ tbc: time base of codec

计算当前帧的PTS

- ◆ $PTS = PTS * av_q2d (video_stream->time_base)$
- ◆ `av_q2d(AVRational a)`
{ return a.num / (double)a.den; }

计算下一帧的PTS

- ◆ **video_clock**: 预测的下一帧视频的 PTS
- ◆ **frame_delay**: $1/\text{tbr}$
- ◆ **audio_clock**: 音频当前播放的时间戳

音视频同步方式

- ◆ 视频同步到音频
- ◆ 音频同步到视频
- ◆ 音频和视频都同步到系统时钟

视频播放的基本思路

一般的做法，展示第一帧视频帧后，获得要显示的下一个视频帧的 PTS，然后设置一个定时器，当定时器超时后，刷新新的视频帧，如此反复操作。