

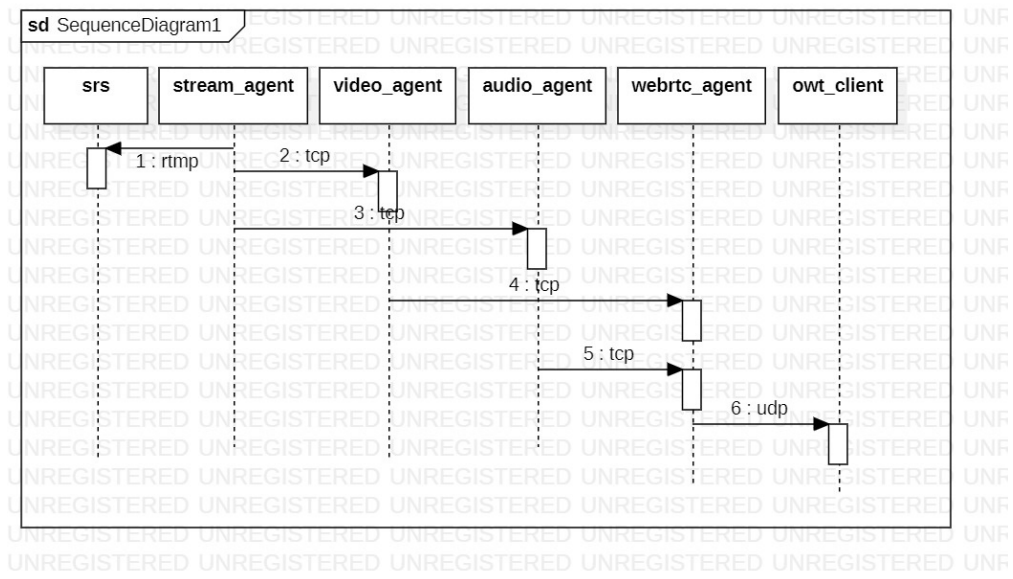
有关WIKI的需求烦请在<https://servicedesk.iflytek.com/> 内提交，选择：质效平台运营 下分类即可

页面 / ... / Open Visual Cloud-Immersive-Video-Sample介绍

owt-server rtmp的码流数据流程

由 陈明军创建, 最后修改于十二月 01, 2022

owt-server内部数据流程图



几个主要的rpc方法以及作用：

createInternalConnection(in):创建一个IO(in)通道，接受外部agent发送过来的码流数据。会带connectionId，一般模式为tcp。

publish:创建一个pub session，和IO(in)通道连接起来，接受IO通道的数据。

createInternalConnection(out):创建一个IO(out)通道，将处理完的数据发送给其他agent。

subscribe:创建一个sub session，和IO(out)通道连接起来，将处理完的数据发给IO(out)通道。

generate:根据agent需要可能创建， pub和sub之间的中转。

linkup:将pub和sub之间的数据连接起来，连通数据通道。

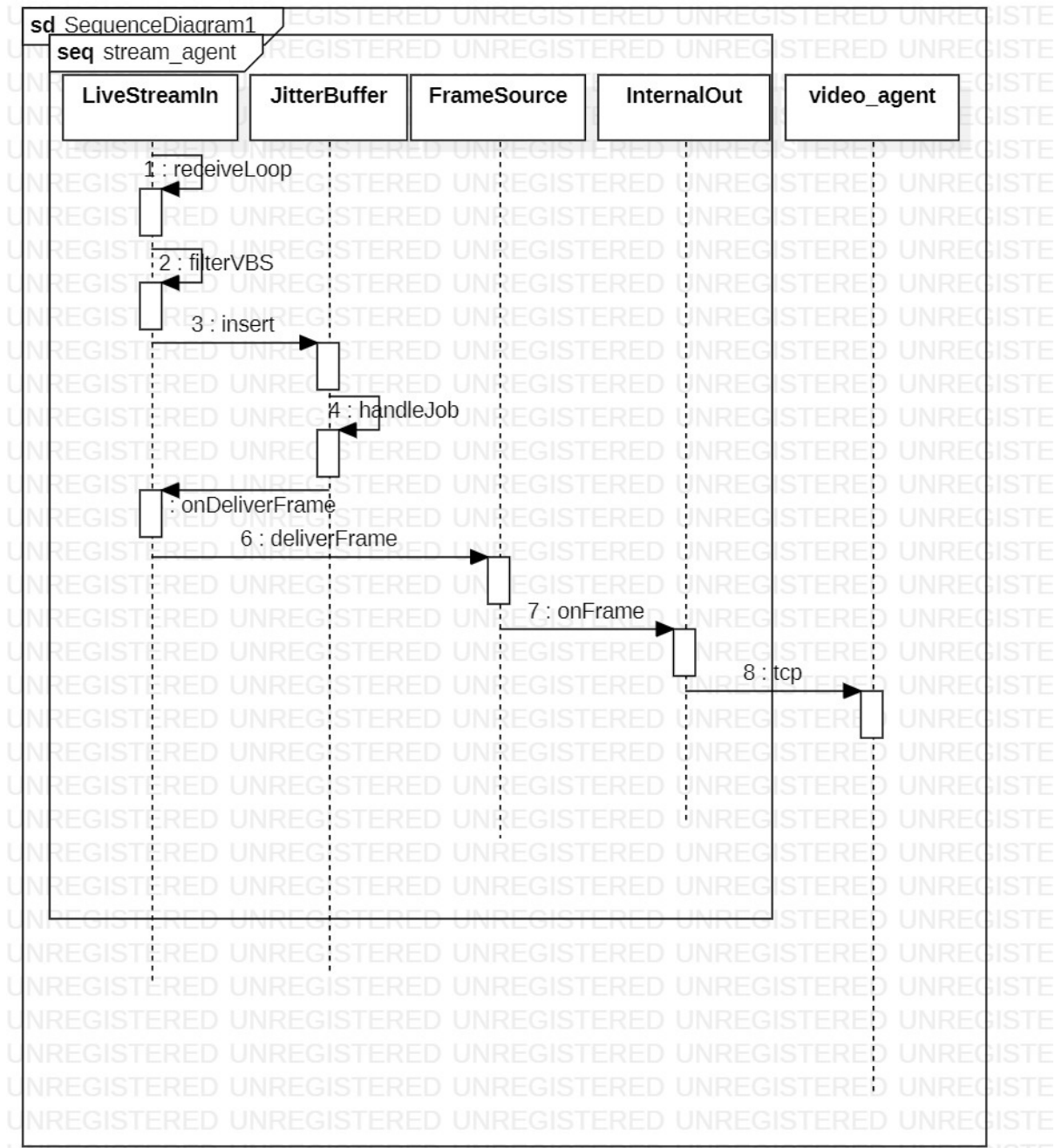
一. stream_agent

- LiveStreamIn、LiveStreamOut等C++类回调给NodeJs的线程通信是用的libuv封装的async（猜的）
- LiveStreamIn的Video和Audio都有Jitterbuffer，从ffmpeg拉取的rtmp数据会丢入这两个队列。jittbuffer最大缓冲1s。
如果队首队尾之间的dts差大于1s，比如1300ms，则将队列中前1300-500=700ms立刻取出，进行处理。
否则每隔两帧dts差的时间就处理一帧。
- LiveStreamIn对格式做了判断，av_bsf_get_by_name从mp4文件读取h264数据，av_bsf_send_packet从flv流读取h264数
- 连接LiveStreamIn和InternalOut是stream_agent(js)做的，调用linkupConnection，FrameSource作为中间类，有In和Out的对应关系。

InternalConnectionFactory.js创建InternalOut类，传递给LiveStreamin作为destination。rtmp中InternalOut类的dest其实就是video_agent和audio_agent，

InternalOut类保存了对端的ip和端口，直接将拉取的流通过udp传输给video_agent和audio_agent。

stream_agent的数据流程图：



二.video_agent

1.udp数据监听的端口是在配置文件中配的，有min_port和max_port，端口是在接受到publish的rpc请求后把端口返回给conference_agent，conference_agent再告诉stream_agent。

2.video_agent要发送的对端的地址是conference_agent在sub时候调用createInternalConnection通知的，video_agent再收到udp数据后知道应该发送给谁。有connectionId作为引用。

3.video_agent默认有mcu模式和SFU模式，决定该模式应该是conference_agent调用init，demo默认mcu模式。在mcu模式中，在conference_agent调用generate接口时，到VideoMixer类，会进行编码工作。hevc的4k和8k会用SVTHEVCMCTSEncoder这个编码器，会编码成大分辨率的和分辨率的图像。在sfu模式中，hevc会用SVTHEVCEncoder这个编码器。

大小分辨率切法为：

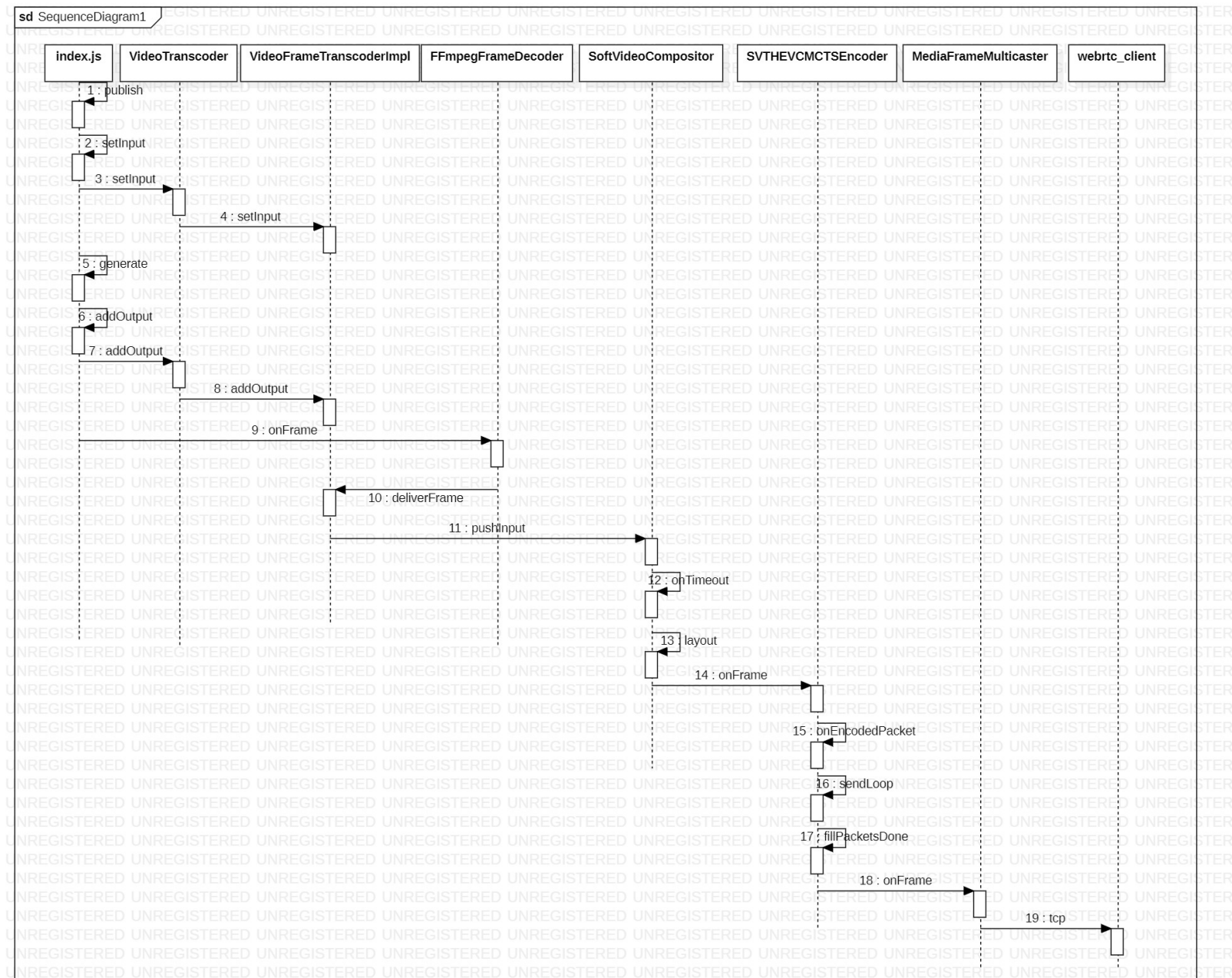
```
if (width == 7680 && height == 3840) {
    m_width_hi = 7680;
    m_height_hi = 3840;
    bitrate_hi = 108000 * 30 / frameRate;
    tiles_w_hi = 12;
    tiles_h_hi = 6;

    m_width_low = 512;
    m_height_low = 1280;
    bitrate_low = calcBitrate(m_width_low, m_height_low, frameRate);
    tiles_w_low = 2;
    tiles_h_low = 2;
} else if (width == 3840 && height == 2048) {
    m_width_hi = 3840;
    m_height_hi = 2048;
    bitrate_hi = 25000 * 30 / frameRate;
    tiles_w_hi = 8;
    tiles_h_hi = 4;

    m_width_low = 1280;
    m_height_low = 768;
    bitrate_low = calcBitrate(m_width_low, m_height_low, frameRate);
    tiles_w_low = 5;
    tiles_h_low = 3;
}
```

编码器将原始原始图像通过SVTHEVCMCTSEncoder，分辨编码出来大分辨率h265和小分辨率h265，两个数据在buffer中累加后发送给webrtc_agent。

2.1 video_agent内部数据流程图（MCU）



mcu比sfu模式 会多出拼接的流程，以及编码器分出大分辨率和小分辨率的功能。


```

sequenceDiagram
    participant Index as Index.js
    participant VideoTranscoder
    participant VideoFrameTranscoderImpl as VideoFrameTranscoderImpl
    participant FFMpegFrameDecoder
    participant FrameProcessor
    participant SYNTHEVCEncoder
    participant MediaFrameMulticaster
    participant webrtc_client

    Index->>VideoTranscoder: 1: publish
    activate VideoTranscoder
    VideoTranscoder->>VideoFrameTranscoderImpl: 2: setOutput
    activate VideoFrameTranscoderImpl
    VideoFrameTranscoderImpl->>VideoTranscoder: 3: setOutput
    deactivate VideoFrameTranscoderImpl
    VideoTranscoder->>VideoFrameTranscoderImpl: 4: setOutput
    activate VideoFrameTranscoderImpl
    VideoFrameTranscoderImpl->>VideoTranscoder: 5: generate
    deactivate VideoFrameTranscoderImpl
    VideoTranscoder->>VideoFrameTranscoderImpl: 6: addOutput
    activate VideoFrameTranscoderImpl
    VideoFrameTranscoderImpl->>VideoTranscoder: 7: addOutput
    deactivate VideoFrameTranscoderImpl
    VideoTranscoder->>VideoFrameTranscoderImpl: 8: addOutput
    activate VideoFrameTranscoderImpl
    VideoFrameTranscoderImpl->>VideoTranscoder: 9: onFrame
    deactivate VideoFrameTranscoderImpl
    VideoTranscoder->>FFMpegFrameDecoder: 10: onFrame
    activate FFMpegFrameDecoder
    FFMpegFrameDecoder->>FrameProcessor: 11: onFrame
    activate FrameProcessor
    FrameProcessor->>SYNTHEVCEncoder: 12: onFrame
    activate SYNTHEVCEncoder
    SYNTHEVCEncoder->>MediaFrameMulticaster: 13: deliverFrame
    activate MediaFrameMulticaster
    MediaFrameMulticaster->>webrtc_client: 14: sendLoop
    deactivate MediaFrameMulticaster
    SYNTHEVCEncoder->>MediaFrameMulticaster: 15: deliverFrame
    deactivate SYNTHEVCEncoder
    MediaFrameMulticaster->>webrtc_client: 16: tp
    deactivate MediaFrameMulticaster
    deactivate Index
    deactivate VideoTranscoder
    deactivate VideoFrameTranscoderImpl
    deactivate FFMpegFrameDecoder
    deactivate FrameProcessor
    deactivate SYNTHEVCEncoder
    deactivate MediaFrameMulticaster
    deactivate webrtc_client
  
```

- ### ≡.webrtc_agent

- ## 内部流程图