**已完成进度：**

1. 视频流的传输

参考网上代码，手动实现RTSP协商、RTCP控制、RTP传输的功能

服务端包含可播放视频，需输入指定参数RTSP\_dest\_port

Java –jar server.jar 666

客户端指定服务器ip、RTSP\_server\_port、RTCP\_RCV\_PORT、RTP\_RCV\_PORT

Java –jar client.jar 10.108.166.14 666 2501 19001

首先，客户端和服务器端指定RTSP的端口，这样能够创建socket进行通信，接着，客户端点击setup按钮，客户端通过RTSP端口发送RTCP、RTP端口信息，服务器端接收并解析，从而得到与客户端进行控制、数据传输的端口号，接着，服务器端生成RTSPid（UUID保证为唯一标识符）并返回给客户端；这样，客户端与服务器端建立了RTSP、RTCP、RTP套接字进行通信，且客户端有唯一RTSPid来进行身份确认

1. 多播方式

重构了代码，源码为单线程方式，客户端与服务器端的RTSP、RTCP、RTP端口都是static静态申明，现改为手动输入，客服端与服务器端在正是通信前需相互通知端口号；服务器端改用多线程方式，每接收到客户端连接请求，就新建线程进行处理

1. 演示示例

服务器socket：<192.168.10.101，666>

客户端socket：<192.168.10.102，port>

中间通过一台SDN交换机（192.168.10.110），分别连接端口6、8，下发流表使得端口6和8相通

**待完成：**

1. 代码仍有改进空间，如可使用线程池的方式优化处理
2. 当开启多个客户端时，视频明显卡顿，应该是数据传输速率下降；分析原因：通道带宽为100M，不会发生丢包，应该是电脑多端口发包速率有限；在后序中可以详细分析这种情况，并列举解决方案
3. 融合发布订阅系统进行传输