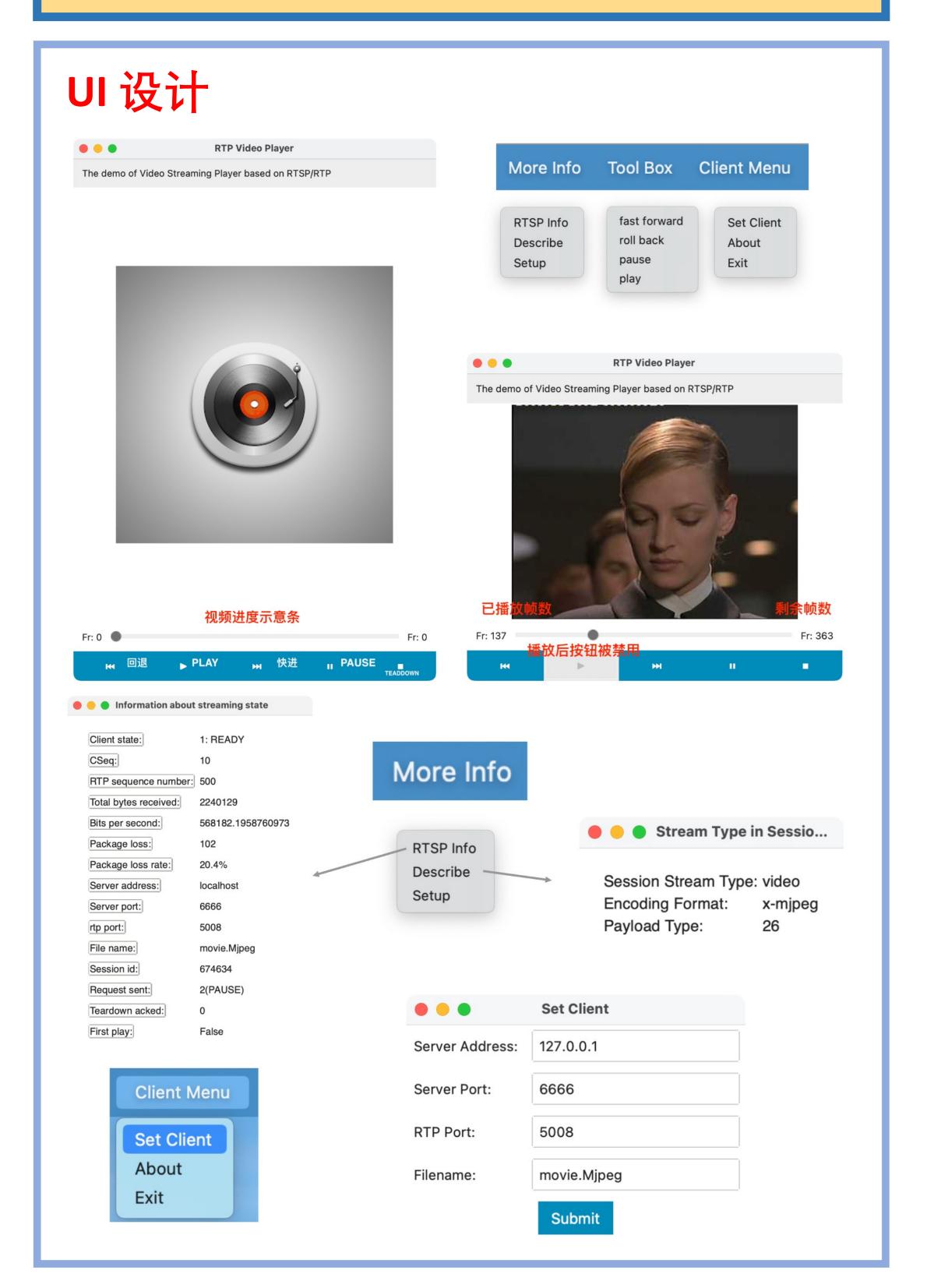
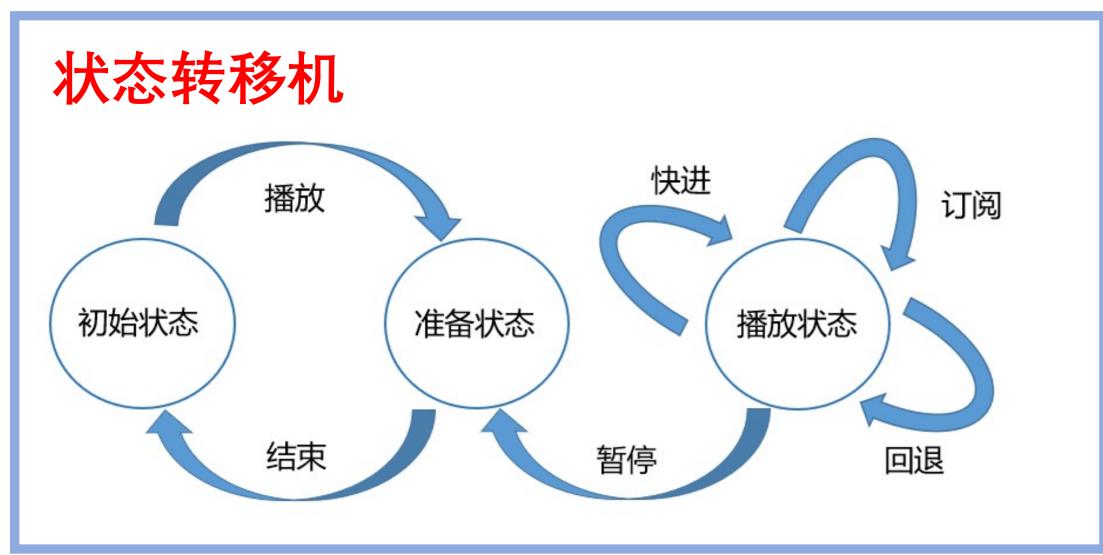
CS3611 VideoStreaming Player with RTSP/RTP

黄奔皓, 朱鹏翔、李佳鑫、周晟洋

工作总结

- ◆ 实现了RTSP/RTP流媒体传输的基本功能,并完成了PLAY合并SETUP、帧率、丢包率和DESCRIBE的附加功能.
- ◆对UI进行了美化,增加了流媒体信息展示界面,实现了对视频播放进度进行展示的功能;实现了Client端用户设置端口信息和文件名的交互界面.
- ◆ 实现了视频的快进和回退功能,并重新设计了播放与暂停按钮的交互逻辑.





特色功能流程示意图 快进/回退的实现 FORWARD request \REVERSE request move_point(100) \ rev_point(100) VideoStream FORWARD reply \REVERSE reply (200 OK) 在完成视频传输前,获取总帧数 SETUP request get_total_frames_num() VideoStream Client Server SETUP reply (200 OK) (random session ID) (Total FrameNbr) Total FrameNbr PLAY与SETUP功能的合并 Client: INIT Client: PLAYING PLAY get the total FrameNbr PLAY FIRST PAUSE Client: READY Client: Waiting have not get the total FrameNbr

def encode(self, version, padding, extension, cc, seqnum, marker, pt, ssrc, payload): """ Encode the RTP packet with header fields and payload. """ timestamp = int(time()) header = bytearray(HEADER_SIZE) header[0] = (version << 6) | (padding << 5) | (extension << 4) | cc header[1] = marker << 7 | pt header[2:4] = seqnum.to_bytes(2, byteorder='big') # big represents [big end] coding header[4:8] = timestamp.to_bytes(4, byteorder='big') header[8:12] = ssrc.to_bytes(4, byteorder='big') self.header = header # Get the payload from the argument self.payload = payload</pre>

RTP包编码过程代码展示

