

非集群 DEMO 使用说明

www.mediapro.cc

CS 架构篇

一、服务器部署

服务器简介:

SRTP-Server 是在 RTP-FEC-QOS 传输层基础上建立了一套通用型 RTP 直播流媒体转发服务器,可用于一对多、多对多等场合的音视频实时互动,播发端支持全平台包括 PC 客户端、浏览器、Android、IOS、微信。它具备以下特点:

- ※ 超低延时,采用 RTP (UDP) 作为传输层解决方案,能获得低于 300ms 的系统延时,适用于实时互动等应用场景。
- ※ 抗丢包能力强,采用 RTP-FEC-QOS 技术,保证上行线路和下线线路在弱网情况下依旧有良好表现。

※ 基于虚拟的房间(教室)模式,支持在房间内广播音视频(RTP)和信令(TCP)。

系统不限房间数量,不限单个房间内客户端的数量。

※ 集群模式或单机模式可选,集群扩展简便,规模可大可小,灵活配置,避免单点故障。

※ 轻量级,代码精简高效,可部署于嵌入式设备之中。

※ 高并发、7*24 高稳定性,支持开源 NYX 监控。

※ 负载均衡,将流量(客户端)合理的分配到集群的各台服务器上。自带 CDN 加速功能,为每个客户端优选最佳的服务器资源,保障用户体验。

※ 支持在同一服务器上部署多套完全独立的集群,多套集群并发工作,可以实现不同业务单位的软件隔离,共享硬件和网络资源。亦可用于主从备份。

※ 支持同步推送 RTMP 流到第三方流媒体服务器,可以与 SRS、FMS、Nginx-rtmp 等结合实现 RTP、RTMP、HLS 流的同步播出,从而实现全平台的直播。

※ 客户端掉线自动重连功能,客户端因网络等异常掉线时,服务器将短时间内等待客户端的重连,获得更好的用户体验。

※ 支持同一个房间内多达 32 路音视频

※ 日志系统完善,支持远程 telnet 命令查询状态,管理和维护更加便捷

系统使用传统 C++ 实现,代码精简高效,同时支持 Win 和 Linux。注重编码规范,注释完善,代码可读性强。剥离了业务相关,通用性强,可以快速加入自身业务。

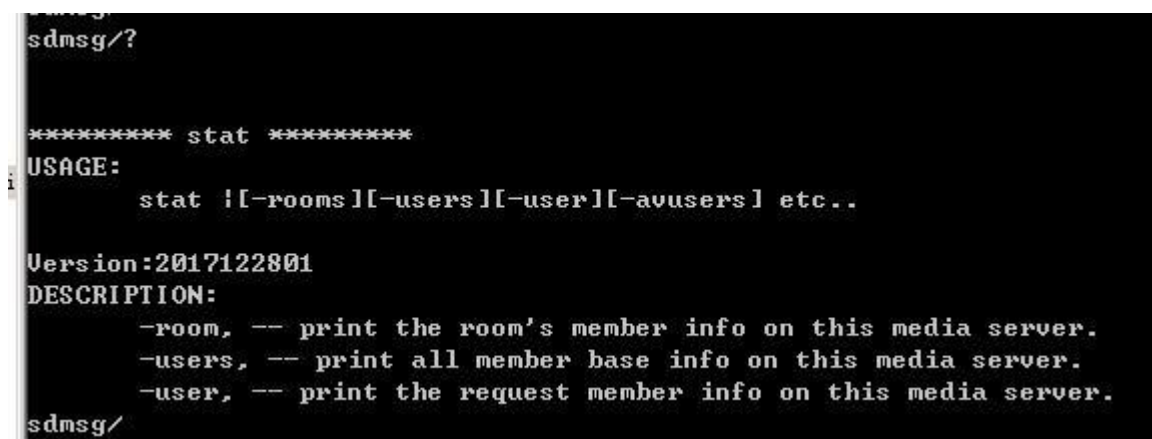
本 DEMO 提供 SRTP-Server 服务器单机版,包括 Win64 版本和 Linux 64 版本。集群版请联系技术支持另行获取。

Linux 版本服务器启动:

请参考文档《媒体服务器安装指南.pdf》

Window 版本服务器启动命令:

- (1) 双击启动 MediaServer.exe
- (2) 启动后即进入命令解析模式, 可以输入?查询服务器支持的命令。



```
sdmsg/?  
  
***** stat *****  
USAGE:  
    stat ![-rooms][-users][-user][-avusers] etc..  
  
Version:2017122801  
DESCRIPTION:  
    -room, -- print the room's member info on this media server.  
    -users, -- print all member base info on this media server.  
    -user, -- print the request member info on this media server.  
sdmsg/
```

图 1 windows 查询 Media 服务器命令支持

二、客户端 DEMO 使用

(1) 1 个 PC 发布者, N 个观看者

启动客户端<RTP_CS 架构_发端_AV_DEMO_WIN>, 设置好服务器 IP 地址、本客户端的用户 ID、需要发布的房间 ID, 即可开始发布。发布后, 将在左边窗口看到本地视频, 右边窗口看到经过服务器下发的流。

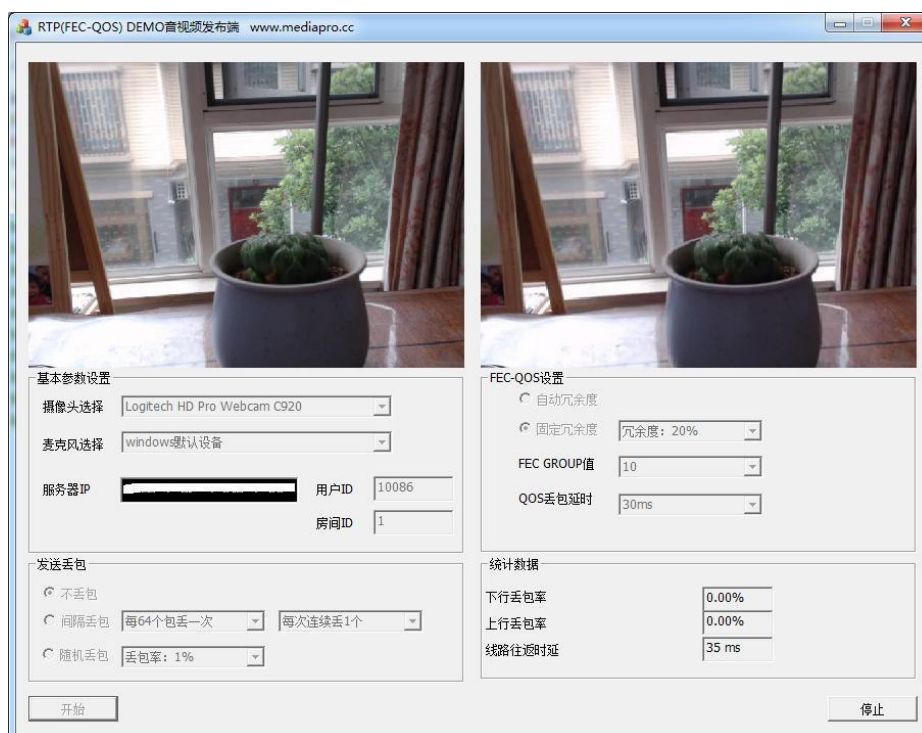


图 2 发端界面 (Win)

当需要更改发布的音视频分辨率、码率时，可以编辑 RTP_CS 架构_发端_AV_DEMO_WIN 文件夹下的 AVClient.ini 文件，文件内容如下：

```
VideoTransWidth=704
VideoTransHeight=576
VideoBitrate=350
UseSingleWindowForDisplay=0
UseFreezeFrameWhenLost=1
AvUpIndex=0
AvDownIndex=0
```

其中 VideoTransWidth、VideoTransHeight、VideoBitrate 分别用于指定编码发送的视频宽高和码率(单位 kbps)，最大支持到 1080P 分辨率，码率 8000kbps，最低 50kbps。

UseSingleWindowForDisplay 用于指定是否使用单独的窗口显示远端画面（画面更大），还是图 1 那样内嵌。

UseFreezeFrameWhenLost 控制是否启用丢帧冻结机制，为 1 开启，为 0 关闭。开启时，若有 FEC 无法恢复的丢包，将冻结画面，直至下一个完整的关键帧到来，这样可以避免用户看到花屏，在一些偏静态的场景下，画面短暂冻结可能较花屏体验更佳。

AvUpIndex 表示本客户端上行音视频到哪一个 Position 位置，位置的概念请参考服务器设计文档，当前 DEMO 限制为 6 个位置即 0~5。

AvDownIndex 表示本客户端接收哪一个 Position 位置的音视频。

【注意】 本 DEMO 带宽受服务器带宽所限，请根据自身服务器带宽情况而定，避免超出服务器带宽限制而丢包。

【注意】 界面上的接收缓存延时用于设置接收缓存大小(单位毫秒)，适当的接收缓存可以缓解因网络带来的抖动，降低画面卡顿率，但也意味着更大的延时。当设置为 0ms 时关闭接收 JitterBuff 功能。

【注意】 修改配置文件后，需要重启程序方能生效。

【注意】 当两个客户端（发端客户端或者播放器）使用相同的 ID 登录时，后登陆的客户端会将之前登录的客户端顶下去。

要观看音视频流可以选择客户端<RTP_CS 架构_单画面播放器_AV_DEMO_WIN>、<RTP_CS 架构_多画面播放器_AV_DEMO_WIN>、<RTP_CS 架构_单画面播放器_AV_DEMO_Android>



图 3 单画面播放器

(2) 1 个 Android 发布者，N 个观看者

Android 客户端<RTP_CS 架构_发端_AV_DEMO_Android>支持前后摄像头热切换、软硬编码热切换、较多的视频美颜滤镜、MP4 文件录制，可在设置界面选择传输参数。

【注意】 APP 并未强制指定 Position 位置，而是从服务器申请一个当前空闲的位置。如果已经运行了一个<RTP_CS 架构_发端_AV_DEMO_WIN>并指定了位置 0，则 APP 将向 1 号位置发送音视频（服务器从 0 开始分配）。

【注意】 需要保证播放器和 APP 在同一房间；域 ID 默认为 1，无需修改（与服务器对应）。



图 4 Android 发送端设置界面

(3) 多个发布者 (WIN or Android), N 个观看者

可以使用多个<RTP_CS 架构_发端_AV_DEMO_WIN> 或者 <RTP_CS 架构_发端_AV_DEMO_Android> 作为发布者, 将音视频发布到指定房间。本服务器在同一个房间内允许同时 6 路音视频发布 (默认 6 路, 可增大到 32 路), 对应发布位置分别为[0~5]。使用下面<RTP_CS 架构_多画面播放器_AV_DEMO_WIN>即可收看多画面。

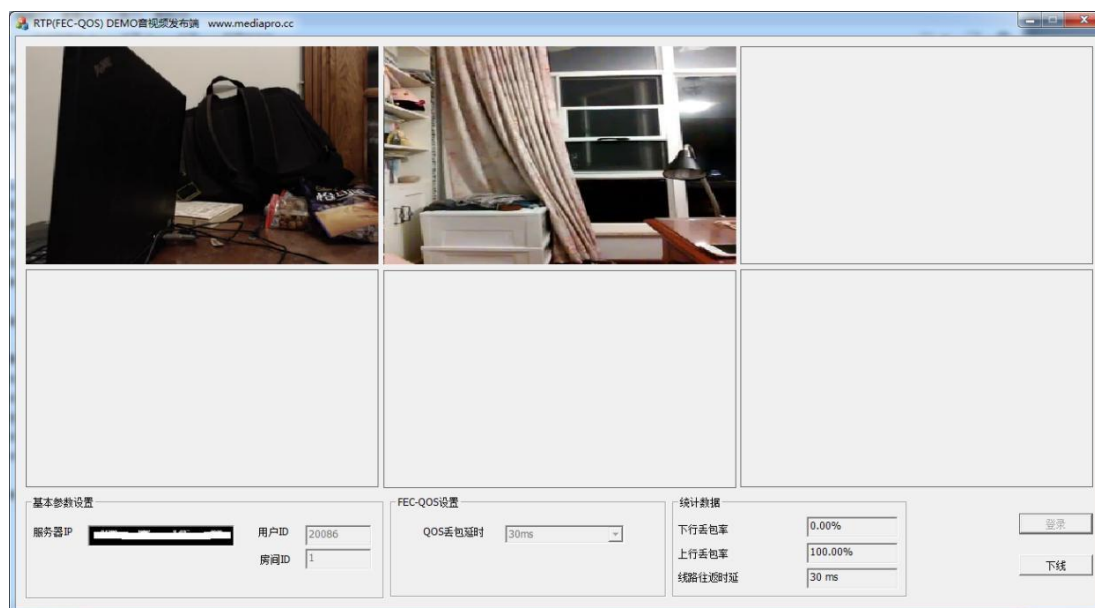


图 5 多画面播放器

点对点篇

一、<RTP_点对点_AV_DEMO_WIN>

本 DEMO 用于局域网内点对点互通的测试，支持摄像头、MIC 采集音视频数据，编码后 RTP 传输到指定客户端 DEMO。双方要求在同一网段内。默认情况下收发 IP 地址为本机 IP，这样相当于本地回环模式，测试更加简便。本地回环模式下，音视频数据依然经历采集、编码、发送、接收、解码、渲染的完整过程，方便用户了解全流程工作机制。

本 DEMO 支持 NVI 硬编码、Qsv 硬编码、X264 软编码，当所在的机器支持相关硬编码时将优先使用硬编码，否则使用 X264 软编码。

其配置文件除了上面讲过的宽高、码率、单独窗口显示、丢包冻结机制外，还包括以下几个：

工作模式选择：0-RecvOnly 1-SendOnly 2-Send&Recv，一般情况下使用默认的 2 即可。前 0 表示只接收流不发送流，1 表示只发送流不接收流，适用于不同的测试需求。

录制接收的流到 TS 文件：CaptureDownStream 为 0 则关闭，否则开启。录制文件将存放于 ts 目录下。

设置编码质量：EncodeQualityLevel0to7，取值范围 0~7（对应于 X264 的 preset，比如 fast、veryfast、slow 等），值越大同等码率下图像质量越高，但计算开销也越大。X264 软编码时，若设置得过高将导致编码丢失实时性，一般使用默认设置 1 即可。