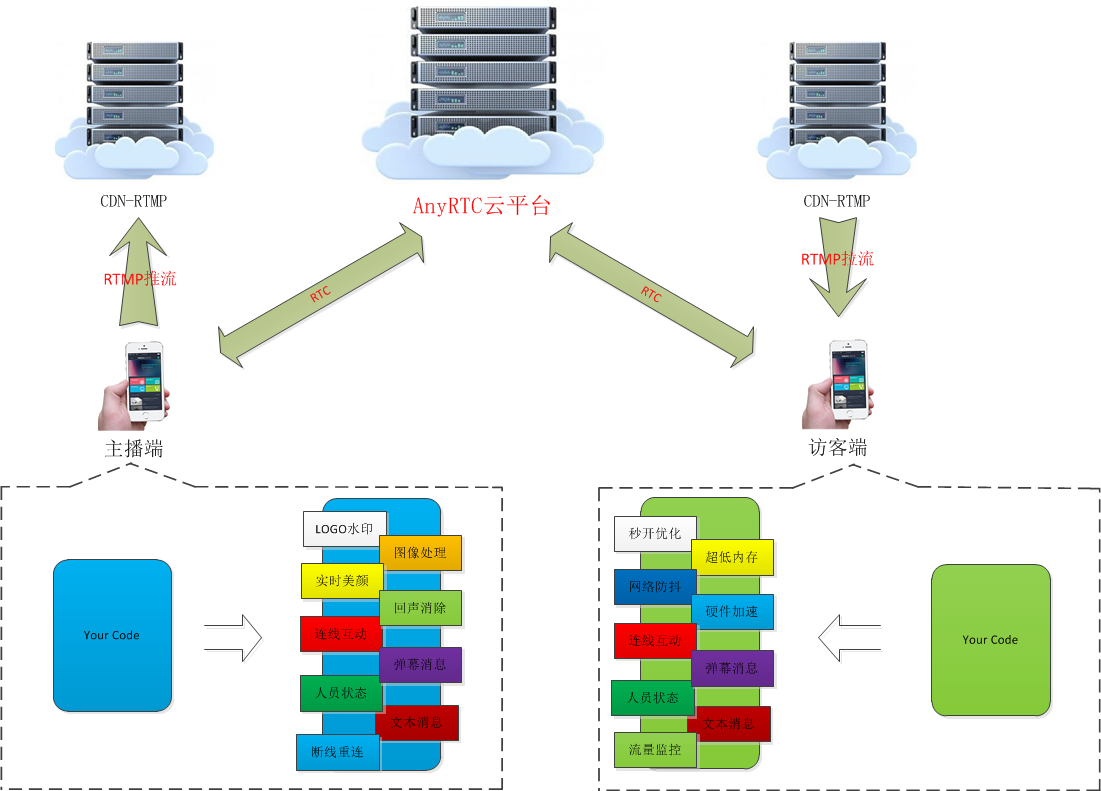
**Android视频连麦直播RTMPC Hybird Engine SDK 集成**

# 简介

AnyRTC云通讯RTMPC Hybird Engine SDK（简称RTMPC）是一套基于RTMP协议可以进行视频连线的混合引擎，由两部分构成：主播端+访客端，本文将主要介绍RTMPC的相关信息。

    首先该SDK遵循标准RTMP视频推流拉流协议，完全兼容市场上主流的标准RTMP视频直播服务器及云平台。与此同时，AnyRTC云通讯创新性实现了RTMP+RTC的技术融合，使用RTC技术实现基于标准的RTMP协议的互动连线视频直播，当然SDK内部囊括了AnyRTC音视频团队多年的技术积累，在视频处理、硬件加速、美颜滤镜、音频降噪、码率控制、实时通讯、移动端性能等方面都做了很多的优化处理。

如果您是一位需要在产品中需要加入互动连线直播的合作伙伴，您只需要几行代码即可完成对对接流程，无论您是新开发的或者已经有产品上线的需求，RTMPC都能为您定制出最符合需求的方案，简单的说RTMPC不会对您的产品系统架构进行大刀阔斧的修改，完全基于RTMP的方案实现互动连线，绝非市面上的什么基于私有协议、UDP等互动直播方案。



                            AnyRTC-RTMPC系统结构图

    如图所示：主播端和访客端通过RTMP协议继续推流拉流，如果需要进行互动连线直接使用SDK中的RTC功能即可。

    RTMPC主要的功能包括如下几个主要部分：

    1）RTMP推流拉流

    RTMPC本身是RTMP+RTC的混合引擎，支持RTMP推流和拉流功能；

    2）RTC视频连线

    视频连线功能分为主播端和访客端，连线的控制权限在主播端即接受、拒绝等权限由主播进行控制，访客可以申请与主播进行连线；

    3）消息弹幕功能

    支持主播/访客发送文本和弹幕消息，支持自定义消息内容；

    4）人员状态动态更新

    支持当前直播的人员上下线实时动态更新，支持用户自定义人员信息；

    6）LOGO水印

    在主播端的直播视频上添加自定义水印LOGO。

    RTMPC最多支持4人互动，但是不会额外增加带宽消耗及其他的开销；RTMPC之所以称之为混合引擎，因为SDK中同时装配了两套引擎，让这两套引擎同时发挥各自的优势，扬长避短才是RTMPC的真正优势。

# 下载Demo

前往<https://github.com/AnyRTC/RTMPCHybirdEngine-Android> 进行下载。下载完成后可直接使用Android Studio 开发工具直接打开，进行编译运行。

# 获取开发者信息

1）前往<https://www.anyrtc.io/> 进行注册开发，在官网首页选择登录或注册，新用户先注册，老用户直接登录。



2）注册或登录完成后，会返回至首页，点击Apps 选项后进行创建一个App。

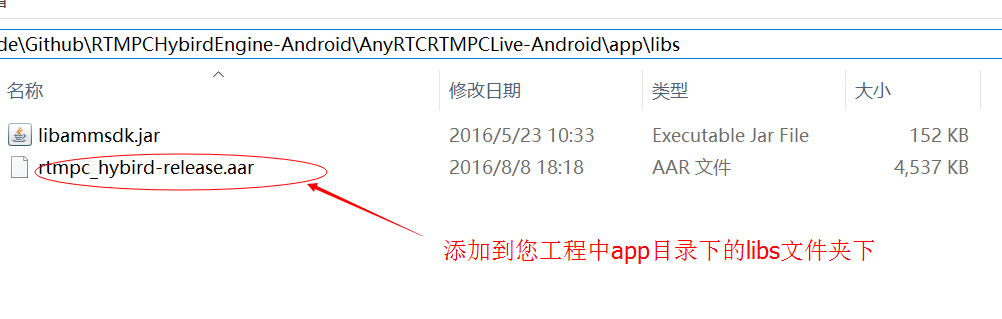




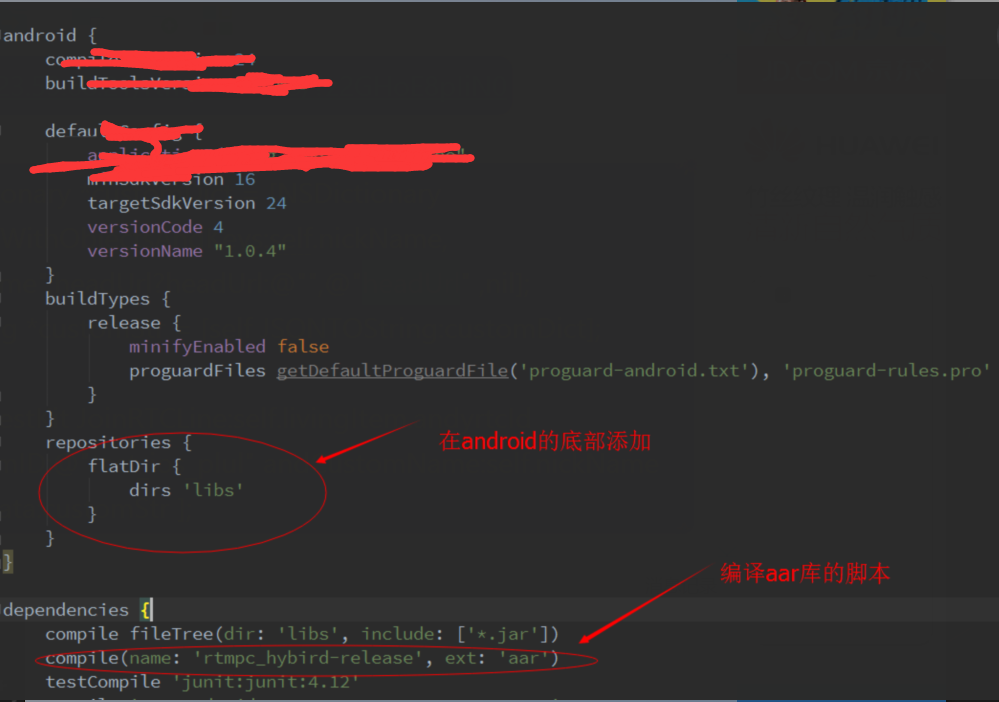
# 快速集成

1. 使用Android Studio 新建一个Android应用；
2. 拷贝下载的Demo中的app文件夹下的libs目录中的rtmpc\_hybird-release.aar 文件，

粘贴到您的工程中；



1. 编辑Android工程中app的build.gradle文件，添加编译aar文件的脚本；



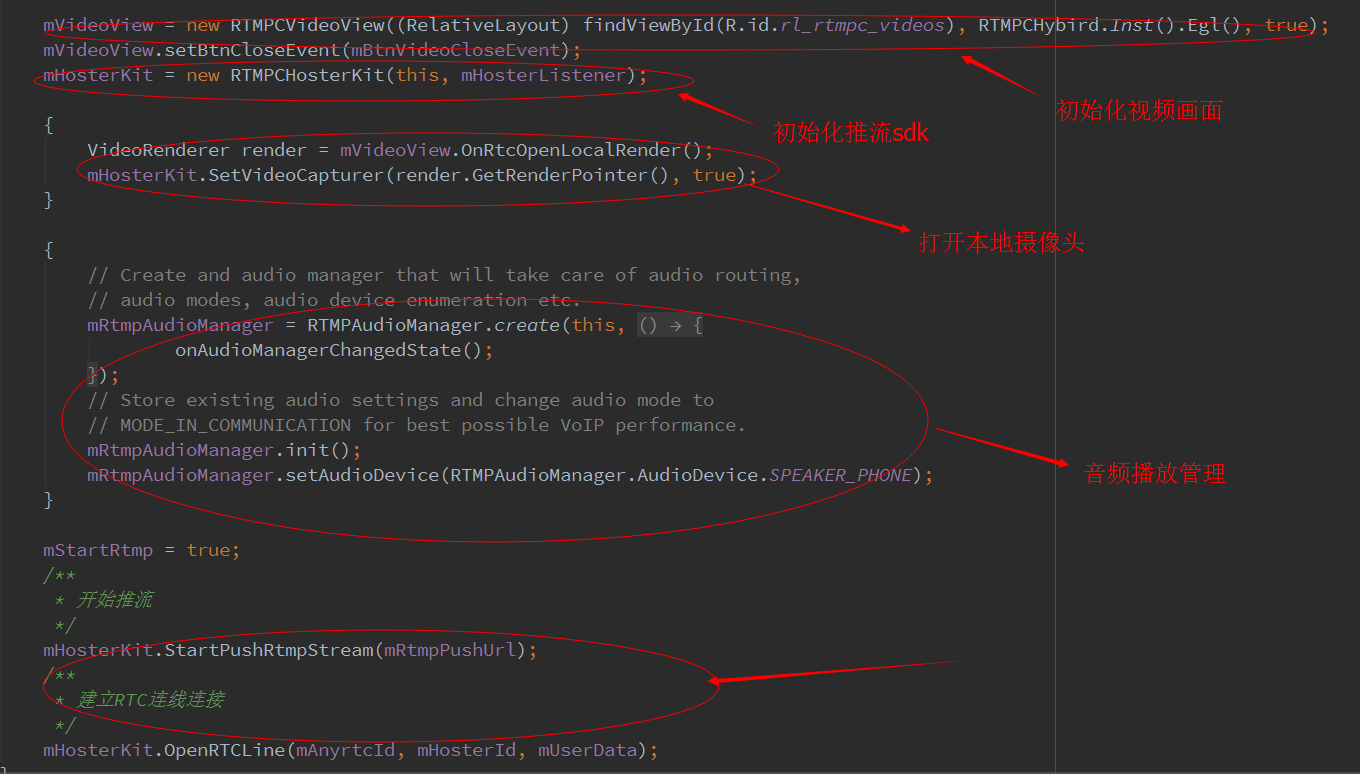
1. 增加AndroidManifest.xml中相机、麦克风等使用权限

<uses-feature android:name="android.hardware.camera" />  
<uses-feature android:name="android.hardware.camera.autofocus" />  
<uses-feature android:name="android.hardware.camera.flash" />  
<uses-feature  
 android:glEsVersion="0x00020000"  
 android:required="true" />  
  
<uses-permission android:name="android.permission.CAMERA" />  
<uses-permission android:name="android.permission.MODIFY\_AUDIO\_SETTINGS" />  
<uses-permission android:name="android.permission.RECORD\_AUDIO" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />  
<uses-permission android:name="android.permission.WAKE\_LOCK" />  
<uses-permission android:name="android.permission.WRITE\_SETTINGS" />  
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />  
<uses-permission android:name="android.permission.CHANGE\_NETWORK\_STATE" />  
<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE"/>  
<uses-permission android:name="android.permission.READ\_PHONE\_STATE"/>

1. 使用AnyRTC云平台上面申请的应用的appid，developerId（即云平台的用户id），appkey和appToken初始化直播引擎（参考Demo中的MainActivity）：



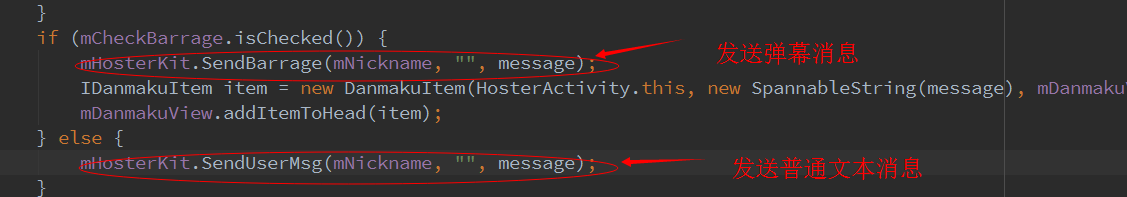
1. 主播端（推流端）集成
2. 初始化视频画面及推流sdk



1. 简单弹幕显示：



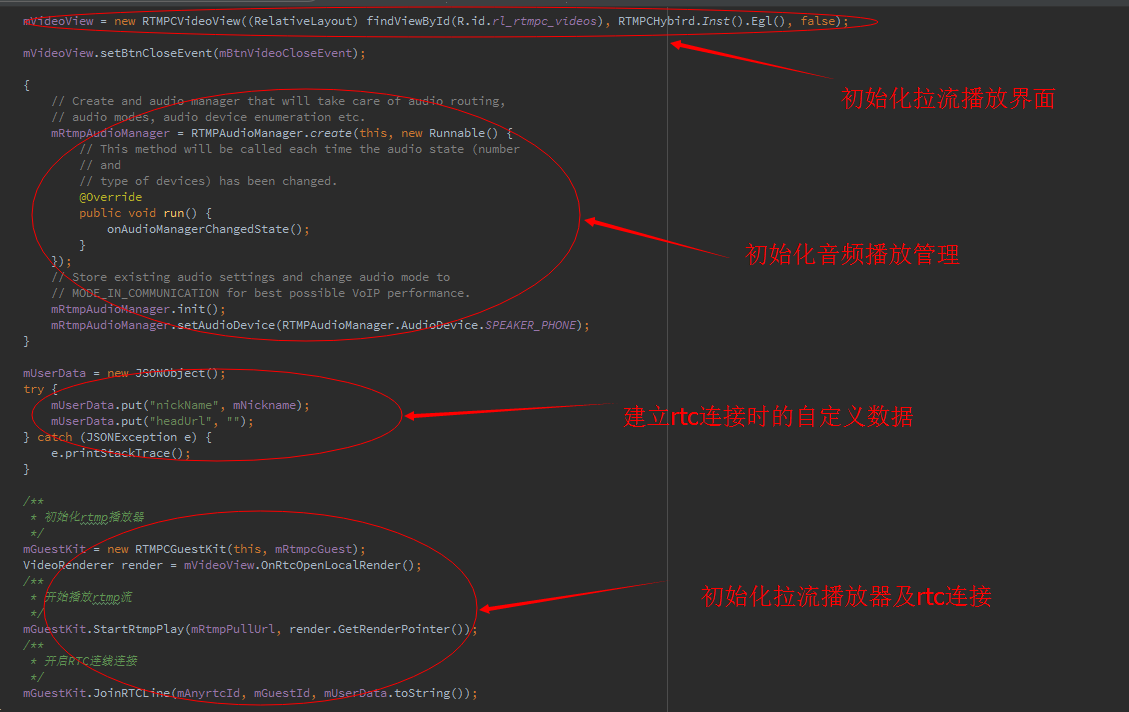
1. 消息和弹幕集成



1. 推流回调事件使用集成

具体参考Demo中HosterActivity中的mHosterListener对象中的注释。

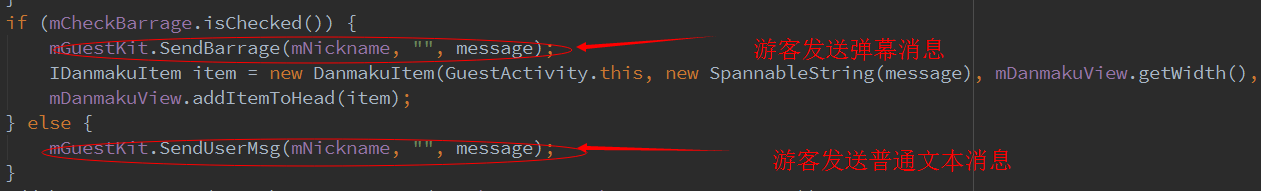
1. 游客端（拉流端）集成
2. 初始化视频播放界面及拉流sdk



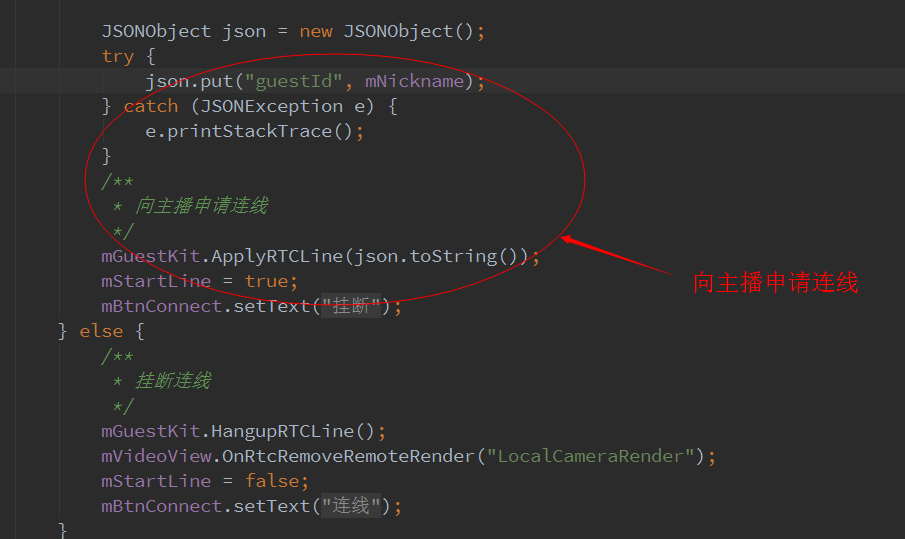
1. 简单弹幕显示



1. 消息及弹幕集成



1. 游客申请连线



1. 拉流回调事件集成

具体参考Demo中GuestActivity中的mGuestListener对象中的注释。

# 获取大厅直播列表

参考Demo中的RTMPCHttpSDK类

使用AnyRTC云平台中申请的应用的appid，developerId，appToken获取大厅的直播列表；