# [小鱼音视频通话SDK接口文档](https://wiki.devlocal.zaijia.cn/pages/viewpage.action?pageId=34564149)

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 日期 | 修改内容 | 修改人 |
| V1.0 | 2019/5/15 | 初稿 | 刘俊华、郑家炜 |
| V2.0 | 2019/6/3 | 平台化支持 | 刘俊华 |

该文档主要针对开发人员与测试人员，描述了小鱼音视频通话SDK的详细接口说明，如后续接口有变更需要同步到此文档中。

## 一，小鱼音视频通话SDK概述

小鱼SDK支持android、ios、windows、linux等多种操作系统，为客户App提供简单、易用的接入方式。

小鱼收到客户的SDK使用权限申请后，为客户指定的App分配唯一的appId，同时生成appSecret，并提供token生成算法。

客户App后台使用appSecret、appid、uid、timestamp结合小鱼提供的token算法生成token(其中uid为客户App中设备/用户的唯一标识，相同的uid会被认定为同一实体)。

APP在初始化小鱼SDK时传入token，待小鱼音视频系统完成鉴权且返回结果为成功，即可发起和接收音视频通话。

### 1.1，平台支持：

支持iOS，Android，Windows，Linux，嵌入式ARM开发平台。

### 1.2，资源与性能：

Intel i7 -7系列移动版本双核处理器上

CPU占用：5%（音频），32%（音频+视频）

运行内存：20MB（音频），50MB（音频+视频），

### 1.3，SDK构成：

Android SDK由 rtc-core-xxx.aar和rtc-wrapper工程源码构造，rtc-core为音视频能力接口封装，rtc-wrapper为rtc-core能力的默认实现(示例代码，包含UI和简单的交互逻辑)。  
以上名字中“xxx”表示版本号，需根据实际情况来确定。

### 1.4，集成：

// 引入rtc-core

implementation files('$path/rtc-core-release-xxxx.aar')

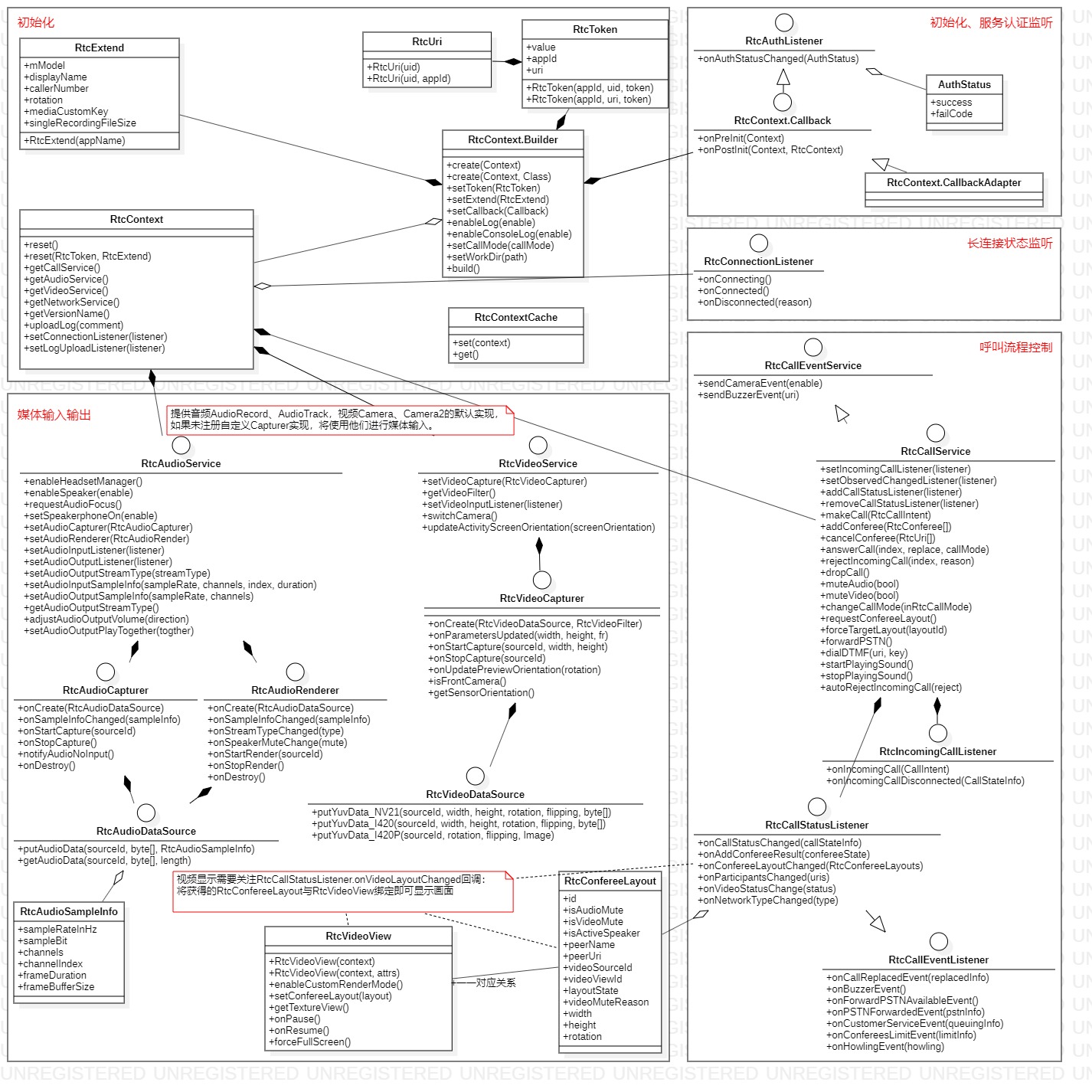
// 引入第三方依赖：

implementation 'com.google.code.gson:gson:x.x.x'

// 添加混淆规则

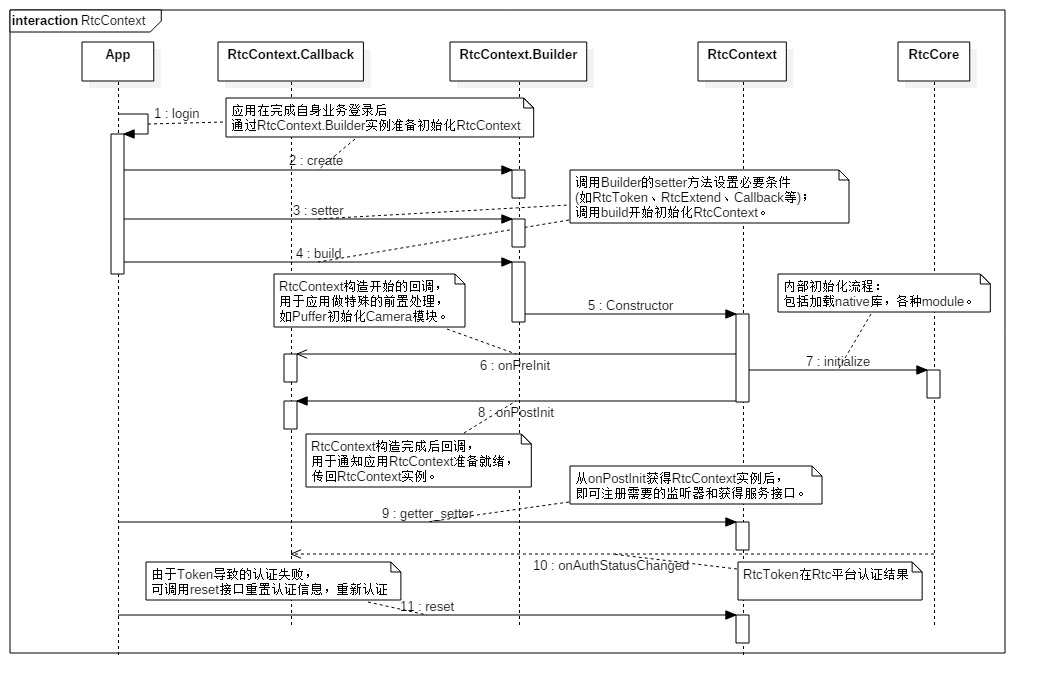
-keep class com.xiaoyu.\*\* { \*;}

## 二，小鱼音视频通话SDK主要接口描述



**图1** SDK接口com.xiaoyu.open包主要类图

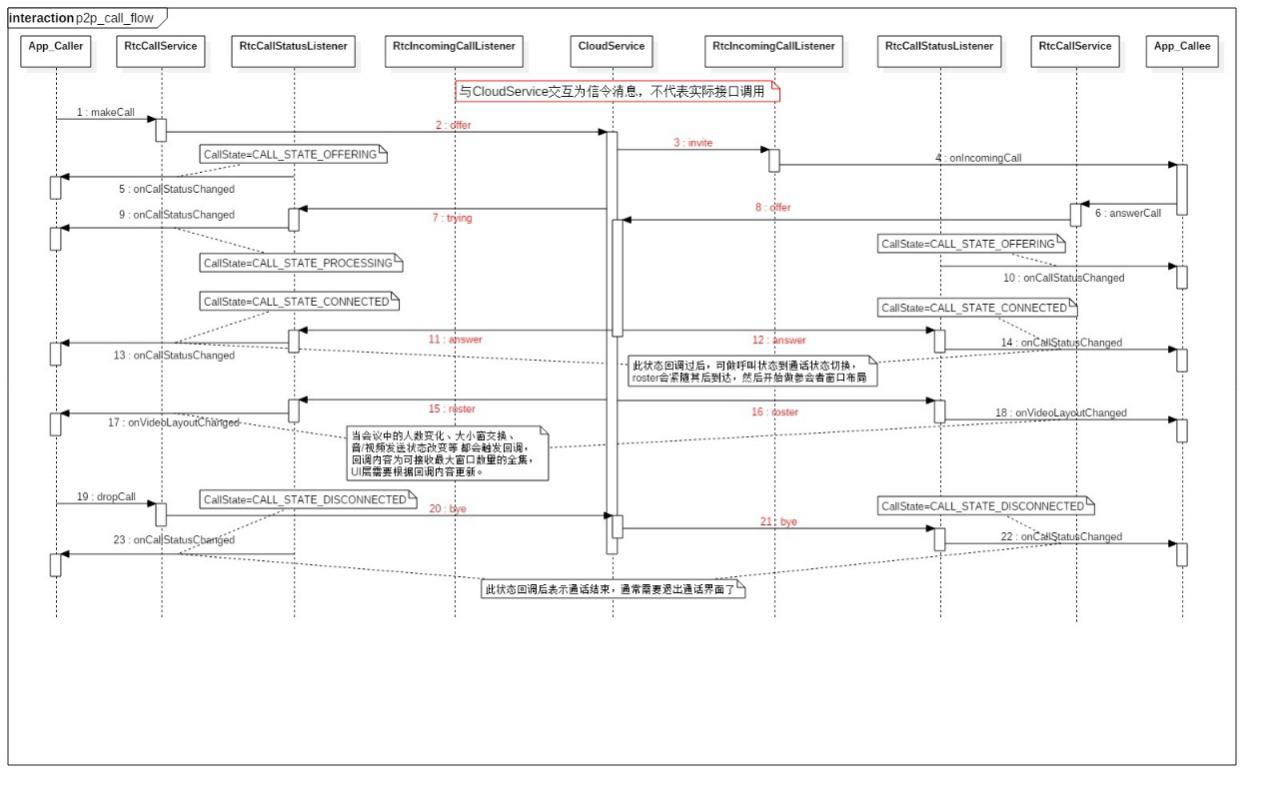
App通过SDK初始化[RtcSdkContext.Builder](#_2.1，SDK初始化)接口和相应的回调接口[RtcAuthListener](#_2.2，鉴权、初始化回调接口)、[RtcContext.Callback](#Callback)完成SDK的初始化工作，初始化流程的时序图如下。



**图2** SDK初始化时序图

App初始化SDK上下文[RtcSdkContext](#_2.2，SDK上下文接口)实例后，从中获取SDK通话服务接口[RtcCallService](#_2.4，SDK通话服务接口)，配合来电监听回调[RtcIncomingCallListener](#_2.5，来电监听回调接口)、通话状态回调[RtcCallStatusListener](#_2.6，通话状态回调接口)打通音视频通话。

典型的音视频通话的时序图如下。



**图3** SDK音视频通话时序图

### 2.1，SDK初始化

SDK初始化接口为RtcSdkContext.Builder，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **create** | |
| 接口 | static Builder create(Context context) | |
| 说明 | 工厂方法，创建RtcSdkContext.Builder实例 | |
| 参数 | context | 类型android.content.ContextApp上下文实例 |
| 返回 | RtcSdkContext.Builder实例 | |

s

|  |  |  |
| --- | --- | --- |
| **名称** | **create** | |
| 接口 | static Builder create(Context context, Class<? extends RtcContext> clz) | |
| 说明 | 工厂方法，创建RtcSdkContext.Builder实例 | |
| 参数 | context | 类型android.content.ContextApp上下文实例 |
| 参数 | clz | 继承RtcContext的子类类型 |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setToken** | |
| 接口 | Builder setToken(RtcToken token) | |
| 说明 | 设置app鉴权信息 | |
| 参数 | token | 用于SDK对App鉴权，详见[RtcToken](#_3.1，RtcToken) |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setExtend** | |
| 接口 | Builder setExtend(RtcExtend extend) | |
| 说明 | 设置app扩展信息 | |
| 参数 | extend | 扩展信息，用于SDK必要用户信息收集，详见[RtcExtend](#_3.2，RtcExtend) |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setCallback** | |
| 接口 | Builder setCallback(RtcContext.Callback callback) | |
| 说明 | 设置SDK初始化回调 | |
| 参数 | callback | 初始化回调实例，详见[RtcContext.Callback](#Callback) |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **enableLog** | |
| 接口 | Builder enableLog(Boolean enable) | |
| 说明 | 是否启用rtc服务日志，默认为true，关闭后无法定位通话问题。 | |
| 参数 | enable | 是否启用rtc服务日志 |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **enableConsoleLog** | |
| 接口 | Builder enableConsoleLog(Boolean enable) | |
| 说明 | 是否将rtc服务日志打印到控制台，默认为false，调用enableLog打开日志后此设置才有效。 | |
| 参数 | enable | 是否打印rtc服务日志到控制台 |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setCallMode** | |
| 接口 | Builder setCallMode(RtcCallMode mode) | |
| 说明 | 设置rtc服务初始化为仅支持指定通话模式的样子，默认为RtcCallMode. CallMode\_AudioVideo。 | |
| 参数 | mode | 通话模式，参见 [RtcCallMode](#_3.9，RtcCallMode) |
| 返回 | RtcSdkContext.Builder实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **build** | |
| 接口 | void build() | |
| 说明 | 应用所有设置构建RtcContext实例  构建开始前会调用[RtcContext.Callback](#Callback)的onPreInit回调  构建完成后会调用[RtcContext.Callback](#Callback)的onPostInit回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

### 2.2，鉴权、初始化回调接口

鉴权回调接口为RtcAuthListener，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **onAuthStatusChanged** | |
| 接口 | void onAuthStatusChanged(RtcAuthStatus status) | |
| 说明 | Rtc服务认证状态改变时回调，App需密切关注此消息，当认证失败时根据业务需求调用RtcContext.reset(RtcToken, RtcExtend)进行重新认证 | |
| 参数 | status | 认证状态，详见 [RtcAuthStatus](#_3.3，RtcAuthStatus) |
| 返回 | 无 | |

 初始化接口RtcContext.Callback为RtcAuthListener子类，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **onPreInit** | |
| 接口 | void onPreInit(android.content.Context context) | |
| 说明 | RtcContext初始化前调用 | |
| 参数 | context | RtcSdkContext.Builder create方法传入的app上下文 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onPostInit** | |
| 接口 | void onPostInit(android.content.Context context,  RtcContext rtcContext) | |
| 说明 | RtcContext初始化后调用 | |
| 参数 | context | RtcSdkContext.Builder create方法传入的app上下文 |
| rtcContext | RtcContext上下文实例，应用需要妥善保存 |
| 返回 | 无 | |

### 2.3，SDK上下文接口

SDK上下文为RtcSdkContext，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **getCallService** | |
| 接口 | RtcCallService getCallService() | |
| 说明 | 获得通话服务实例 | |
| 参数 | 无 |  |
| 返回 | RtcCallService实例，详见[RtcCallService](#_2.4，SDK通话服务接口) | |

|  |  |  |
| --- | --- | --- |
| **名称** | **getAudioService** | |
| 接口 | RtcAudioService getAudioService() | |
| 说明 | 获得音频服务实例 | |
| 参数 | 无 |  |
| 返回 | RtcAudioService 实例，详见 [RtcAudioService](#_2.7，音频服务接口) | |

|  |  |  |
| --- | --- | --- |
| **名称** | **getVideoService** | |
| 接口 | RtcVideoService getVideoService() | |
| 说明 | 获得视频服务实例 | |
| 参数 | 无 |  |
| 返回 | RtcVideoService 实例，详见[RtcVideoService](#_2.10，视频服务接口) | |

|  |  |  |
| --- | --- | --- |
| **名称** | **getNetWorkService** | |
| 接口 | RtcNetworkService getNetworkService() | |
| 说明 | 获得网络服务实例，提供网络测试相关接口 | |
| 参数 | 无 |  |
| 返回 | RtcNetworkService实例 ，参见 [RtcNetworkService](#_2.14，网络服务) | |

|  |  |  |
| --- | --- | --- |
| **名称** | **getVersionName** | |
| 接口 | String getVersionName() | |
| 说明 | 返回SDK版本信息 | |
| 参数 | 无 |  |
| 返回 | 版本字符串 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setConnectionListener** | |
| 接口 | void setConnectionListener(RtcConnectionListener listener) | |
| 说明 | 设置rtc服务连接状态监听器、可用做用户提示 | |
| 参数 | listener | rtc服务连接状态监听器实例 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setUploadLogListener** | |
| 接口 | void setLogUploadLogListener(RtcUploadLogListener listener) | |
| 说明 | 设置rtc日志上传监听器、可用做用户提示 | |
| 参数 | listener | rtc日志上传监听器实例 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **uploadLog** | |
| 接口 | long uploadLog(String comment) | |
| 说明 | 上传rtc日志，辅助定位问题 | |
| 参数 | comment | 备注信息，通常为问题描述内容 |
| 返回 | 上传操作唯一序号，RtcUploadLogListener回调会携带相同序号 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **reset** | |
| 接口 | void reset() | |
| 说明 | 重置Rtc服务设置，通常在应用业务登出后调用 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **reset** | |
| 接口 | void reset(RtcToken token, RtcExtend extend) | |
| 说明 | 重置Rtc服务设置  通常在应用业务更换更换账号信息或者更新认证信息后调用 | |
| 参数 | token | app认证信息 |
|  | extend | app扩展信息 |
| 返回 | 无 | |

### 2.4，SDK通话服务接口

通话服务接口为 RtcCallService，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **setIncomingCallListener** | |
| 接口 | void setIncomingCallListener(RtcIncomingCallListener listener) | |
| 说明 | 设置来电监听器  RtcIncomingCallListener需要是一个长生命周期实例，避免错过任何来电 | |
| 参数 | listener | 来电监听器实例，详见[RtcIncomingCallListener](#_2.5，来电监听接口) |
| 返回 | boolean | |

|  |  |  |
| --- | --- | --- |
| **名称** | **addCallStatusListener** | |
| 接口 | void addCallStatusListener(RtcCallStatusListener listener) | |
| 说明 | 注册通话状态监听器，  必须和removeCallStatusListener成对使用，避免内存泄漏。 | |
| 参数 | listener | 通话状态监听器实例，详见[RtcCallStatusListener](#_2.6，通话状态回调接口) |
| 返回 | boolean | |

|  |  |  |
| --- | --- | --- |
| **名称** | **removeCallStatusListener** | |
| 接口 | void removeCallStatusListener(RtcCallStatusListener listener) | |
| 说明 | 移除已注册通话状态监听器 | |
| 参数 | listener | 通话状态监听器实例，详见[RtcCallStatusListener](#_2.6，通话状态回调接口) |
| 返回 | boolean | |

|  |  |  |
| --- | --- | --- |
| **名称** | **makeCall** | |
| 接口 | RtcMakeCallResult makeCall(RtcCallIntent intent) | |
| 说明 | 发起呼叫，一个会议仅需呼出一次  通话中可使用addConferee来邀请其他人进入当前会议 | |
| 参数 | intent | 被叫端信息，详见[RtcCallIntent](#_3.4，RtcCallIntent)  此参数可借助RtcCallIntent.createForMakeCall接口构造 |
| 返回 | 呼出本地操作结果，详见[RtcMakeCallResult](#_3.5，MakeCallResult) | |

|  |  |  |
| --- | --- | --- |
| **名称** | **addConferee** | |
| 接口 | void addConferee(RtcConferee… conferees) | |
| 说明 | 邀请指定Conferee对象进入当前会议  通过[RtcCallStatusListener](#_2.6，通话状态回调接口)接口的onAddConfereeResult回调确认邀请结果 | |
| 参数 | conferees | 被邀请端信息 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **cancelConferee** | |
| 接口 | void cancelConferee(RtcUri... uris) | |
| 说明 | 取消邀请  通过[RtcCallStatusListener](#_2.6，通话状态回调接口)接口的onAddConfereeResult回调确认邀请结果 | |
| 参数 | uris | 需要取消邀请Conferee的uri，详见[RtcUri](#_3.6，CallUri) |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **answerCall** | |
| 接口 | void answerCall(int callIndex, boolean replaceCall, RtcCallMode callMode) | |
| 说明 | 通过该方法接听RtcIncomingCallListener收到的呼入请求 | |
| 参数 | callIndex | incomingCall序号 |
| replaceCall | 当RtcCallIntent.from为 From\_Normal时，此参数设置为true，退出当前通话，加入新通话，设置为false，则将主叫加入当前会议中。  当RtcCallIntent.from不为 From\_Normal时，此参数无论设置何值，都将退出当前会议，加入新会议。 |
| callMode | 接听模式，仅进入新会议有效，否则沿用当前会议模式 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **rejectIncomingCall** | |
| 接口 | void rejectIncomingCall(int callIndex, String userInfo) | |
| 说明 | 拒接incomingCall请求 | |
| 参数 | callIndex | incomingCall序号 |
|  | userInfo | 透传业务信息，此信息将出现在主叫侧收到的RtcCallStateInfo.userInfo上 |
| 返回 | 无 | |

|  |  |
| --- | --- |
| **名称** | **dropCall** |
| 接口 | void dropCall() |
| 说明 | 挂断当前正在进行的通话（包含正在发起的呼叫） |
| 返回 |  |

|  |  |  |
| --- | --- | --- |
| **名称** | **muteAudio** | |
| 接口 | void muteAudio(boolean mute) | |
| 说明 | 屏蔽本端音频发送 | |
| 参数 | mute | 屏蔽传入true，解除屏蔽传入false |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **muteVideo** | |
| 接口 | void muteVideo(boolean mute) | |
| 说明 | 屏蔽本端视频发送 | |
| 参数 | mute | 屏蔽传入true，解除屏蔽传入false |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **changeCallMode** | |
| 接口 | void changeCallMode(RtcCallMode callMode) | |
| 说明 | 修改本端通话模式，此操作同时影响音频/视频的收发，详见[RtcCallMode](#_3.6，RtcCallMode) | |
| 参数 | callMode | 设置新的通话模式 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **forceTargetLayout** | |
| 接口 | void forceTargetLayout(int layoutId) | |
| 说明 | 切换指定通话方画面在大窗口显示 | |
| 参数 | layoutId | 来源为  RtcCallStatusListener.onConfereeLayoutChanged(RtcConfereeLayouts) 回调中的RtcConfereeLayout.id，  设置为0时取消固定大窗口逻辑  详见[RtcConfereeLayout](#_3.8，RtcVideoLayout) |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **requestConfereeLayout** | |
| 接口 | void requestConfereeLayout() | |
| 说明 | 手动触发 RtcCallStatusListener.onConfereeLayoutChanged(RtcConfereeLayouts)回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **forwardPstn** | |
| 接口 | void forwardPstn() | |
| 说明 | makeCall呼出过程中，主动转呼被叫pstn手机号码 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **dialDTMF** | |
| 接口 | void dialDTMF(RtcUri uri, String key) | |
| 说明 | 想指定与会者发送按键指令，目前仅仅在PSTN通话中有效 | |
| 参数 | uri | 接收指令的与会者uri |
|  | key | 按键描述信息 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **startPlayingSound** | |
| 接口 | void startPlayingSound(String filePath) | |
| 说明 | 将指定的pcm资源文件中的音频数据混入本端通话输出声音中。  目前仅支持16k 16bit格式的数据 | |
| 参数 | filePath | 可访问的资源文件路径 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **stopPlayingSound** | |
| 接口 | void stopPlayingSound(String filePath) | |
| 说明 | 停止startPlayingSound接口正在播放的资源文件 | |
| 参数 | filePath | 可访问的资源文件路径 |
| 返回 | 无 | |

### 2.5，来电监听回调接口

来电监听接口为RtcIncomingCallListener，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **onIncommingCall** | |
| 接口 | void onIncomingCall(RtcCallIntent intent) | |
| 说明 | 收到来电时回调 | |
| 参数 | intent | 来电信息，详见[RtcCallIntent](#_3.4，RtcCallIntent) |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onIncomingCallDisconnected** | |
| 接口 | void onIncomingCallDisconnected(RtcCallStateInfo status) | |
| 说明 | 来电断开时的回调 | |
| 参数 | status | 来电断开描述信息 |
| 返回 | 无 | |

### 2.6，通话状态回调接口

通话状态监听接口为RtcCallStatusListener，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **onCallStatusChanged** | |
| 接口 | void onCallStatusChanged(RtcCallStateInfo status) | |
| 说明 | 通话状态改变回调 | |
| 参数 | status | 详见[RtcCallStateInfo](#_3.9，CallStateChanged) 和[RtcCallStateInfo.CallState](#_3.9，CallState) |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onAddConfereeResult** | |
| 接口 | void onAddConfereeResult (RtcConfereeState result) | |
| 说明 | 邀请其他人加入通话结果回调 | |
| 参数 | result | 邀请操作结果 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onConfereeLayoutChanged** | |
| 接口 | void onConfereeLayoutChanged (RtcConfereeLayouts layouts) | |
| 说明 | 本端可显示视频窗口信息描述，有如下触发条件：  与会者人数变化、与会者状态变化、与会者发言人变化、本地大小窗手动切换等 | |
| 参数 | layouts | 视频窗口描述集合(不会为null，但是size可能为0)  详见[RtcConfereeLayout](#_3.8，RtcVideoLayout) |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onParticipantsChanged** | |
| 接口 | void onParticipantsChanged (List<RtcUri> uris) | |
| 说明 | 会议中的成员清单，仅仅在与会者人数变化时回调 | |
| 参数 | uris | 会议中的成员清单 |
| 返回 | 无 | |

### 2.7，音频服务接口

音频服务主接口为RtcAudioService，用户可以通过该接口设置自有实现音频采集（SDK提供默认的音频采集，音频播放则为SDK内部调用audioTrack实现）。方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **setAudioCapturer** | |
| 接口 | boolean setAudioCapturer(RtcAudioCapturer capturer) | |
| 说明 | 注册音频采集实现，不注册的话则使用SDK提供的默认实现 | |
| 参数 | capturer | 自实现音频采集实例 |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **enableHeadsetManager** | |
| 接口 | boolean enableHeadsetManager () | |
| 说明 | 启用默认的耳机管理器，提供音频输出在有线耳机、蓝牙耳机、扬声器之间的自动切换。 注意：某些定制系统上可能存在不兼容问题 | |
| 参数 | 无 |  |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setSpeakerphoneOn** | |
| 接口 | boolean setSpeakerphoneOn (Boolean on) | |
| 说明 | 是否将通话音频切换为扬声器输出 | |
| 参数 | on | true表示扬声器输出、false表示听筒/耳机输出、null表示跟随系统默认行为（每次通话自动重置） |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **enableSpeaker** | |
| 接口 | boolean enableSpeaker (boolean enable) | |
| 说明 | 是否开启通话音频播放 | |
| 参数 | enable | true表示开启，false表示关闭，默认为true（每次通话自动重置） |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **requestAudioFocus** | |
| 接口 | boolean requestAudioFocus() | |
| 说明 | 手动为通话音频申请播放焦点，失去播放焦点通话声音会自动静音。 | |
| 参数 | 无 |  |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setAudioInputListener** | |
| 接口 | boolean setAudioInputListener(RtcAudioInputListener listener) | |
| 说明 | 设置音频输入监听器，主要用于接收rtc服务认为音频输入异常的回调 | |
| 参数 | listener | 音频输入监听器实例 |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setAudioOutputListener** | |
| 接口 | boolean setAudioOutputListener(RtcAudioOutputListener listener) | |
| 说明 | 设置音频输出监听器，主要用于监听音频输出通道的切换回调，  调用enableHeadsetManager()后此设置才有意义 | |
| 参数 | listener | 音频输出监听器实例 |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setAudioOutputStreamType** | |
| 接口 | boolean setAudioOutputStreamType(int type) | |
| 说明 | 设置音频输出流媒体类型 | |
| 参数 | type | 可选参数为AudioManager中的 STREAM\_VOICE\_CALL、STREAM\_MUSIC等属性， 默认为STREAM\_VOICE\_CALL |
| 返回 | true设置成功， false设置失败 | |

### 2.8，音频采集接口

音频采集接口为RtcAudioCapturer，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **onCreate** | |
| 接口 | void onCreate (RtcAudioDataSource dataSource) | |
| 说明 | 初始化后回调音频数据源  app需保存好dataSource对象，用于将采集到的音频数据送入SDK | |
| 参数 | dataSource | AudioDataSource实例 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onStartCapture** | |
| 接口 | void onStartCapture(String sourceId) | |
| 说明 | 当SDK需要输入音频数据时回调 | |
| 参数 | sourceId | SDK为音频输入分配的唯一标识，  调用RtcAudioDataSource.putAudioData(String sourceId, byte[] data, RtcAudioSampleInfo sampleInfo) 输入音频数据时传入，详见RtcAudioDataSource |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onStopCapture** | |
| 接口 | void onStopCapture() | |
| 说明 | 当SDK不再需要音频输入时回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **notifyNoInput** | |
| 接口 | void notifyNoInput () | |
| 说明 | 当SDK认为需要音频输入，实际未收到任何数据时周期（5S）性回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onDestroy** | |
| 接口 | void onDestroy() | |
| 说明 | 当重复调用RtcAudioService.setAudioCapturer(RtcAudioCapturer)时可能会回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

### 2.9，音频数据源接口

音频数据源接口为RtcAudioDataSource，用于将通过RtcAudioCapturer采集的数据传输到Rtc服务内部。方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **putAudioData** | |
| 接口 | boolean putAudioData(String sourceId, byte[] data,  RtcAudioSampleInfo info) | |
| 说明 | 将音频数据送入SDK | |
| 参数 | sourceId | SDK分配的唯一标识 |
|  | data | 音频数据 |
|  | info | 音频数据描述信息，详见 [RtcAudioSampleInfo](#_3.6，RtcAudioSampleInfo) |
| 返回 | 是否发送成功 | |

### 2.10，视频服务接口

视频服务主接口为RtcVideoService，用户可以通过该接口设置自有实现视频采集（SDK提供默认的视频采集，视频渲染当前为SDK内部调用openGL实现，调用方式参见demo）。方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **setVideoCapturer** | |
| 接口 | boolean setVideoCapture(RtcVideoCapturer capturer) | |
| 说明 | 注册视频采集实现 | |
| 参数 | capturer | 自实现视频采集实例 |
| 返回 | True设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **setVideoInputListener** | |
| 接口 | boolean setVideoInputListener(RtcVideoInputListener listener) | |
| 说明 | 设置视频输入监听器，主要用于接收rtc服务认为视频输入异常的回调 | |
| 参数 | listener | 视频输入监听器实例 |
| 返回 | true设置成功， false设置失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **requestCamera** | |
| 接口 | void requestCamera() | |
| 说明 | App主动打开摄像头  通常在呼叫还未接通，又希望提前显示本地预览画面时调用 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **releaseCamera** | |
| 接口 | void releaseCamera() | |
| 说明 | App释放摄像头 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **switchCamera** | |
| 接口 | void switchCamera() | |
| 说明 | App切换摄像头 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **updatePreviewOrientation** | |
| 接口 | void updatePreviewOrientation(int range) | |
| 说明 | 设置相机预览画面方向和通话Activity方向之间的顺时针夹角 | |
| 参数 | range | 本地画面和其他端收到的本端画面方向受此参数影响  可选值为0，90，180，270 |
| 返回 | 无 | |

### 2.11，视频采集接口

音频采集接口为RtcVideoCapturer，方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **onCreate** | |
| 接口 | void onCreate (RtcVideoDataSource dataSource) | |
| 说明 | 初始化后回调视频数据源  app需保存好dataSource对象，用于将采集到的视频数据送入SDK | |
| 参数 | dataSource | VideoDataSource实例 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onParametersUpdated** | |
| 接口 | void onParametersUpdated (int width, int height, int frameRate) | |
| 说明 | 回调SDK期望输入的视频最佳分辨率，用于辅助Camera的初始化，避免因参数变化导致Camera重启造成性能浪费 | |
| 参数 | width | 视频宽度 |
|  | height | 视频高度 |
|  | frameRate | 视频帧率 （值被放大了100倍） |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **notifyNoInput** | |
| 接口 | void notifyNoInput () | |
| 说明 | 当SDK认为需要视频输入，实际未收到任何数据时周期（5S）性回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onRequestCamera** | |
| 接口 | void onRequestCamera() | |
| 说明 | 响应RtcVideoService. requestCamera()调用 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onReleaseCamera** | |
| 接口 | void onReleaseCamera() | |
| 说明 | 响应RtcVideoService. releaseCamera ()调用 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onSwitchCamera** | |
| 接口 | void onSwitchCamera () | |
| 说明 | 响应RtcVideoService. switchCamera ()调用 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onUpdatePreviewOrientation** | |
| 接口 | void onUpdatePreviewOrientation (int range) | |
| 说明 | 响应RtcVideoService. updatePreviewOrientation ()调用 | |
| 参数 | range | 本地画面和其他端收到的本端画面方向受此参数影响  可选值为0，90，180，270 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onStartCapture** | |
| 接口 | void onStartCapture(String sourceId, int width, int height) | |
| 说明 | 当Rtc服务需要视频输入时回调 | |
| 参数 | sourceId | SDK为视频输入源分配的唯一标识 |
|  | width | 视频宽 |
|  | height | 视频高 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onStopCapture** | |
| 接口 | void onStopCapture (String sourceId) | |
| 说明 | 当Rtc服务需要视频输入时回调 | |
| 参数 | sourceId | SDK为视频输入源分配的唯一标识 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onDestroy** | |
| 接口 | void onDestroy() | |
| 说明 | 当重复调用RtcVideoService.setVideoCapture(RtcVideoCapturer)时可能会回调 | |
| 参数 | 无 |  |
| 返回 | 无 | |

### 2.12，视频数据源接口

视频数据源接口为RtcVideoDataSource，用于将通过RtcVideoCapturer采集的视频送入SDK。方法描述如下：

|  |  |  |
| --- | --- | --- |
| **名称** | **putYuvData\_NV21** | |
| 接口 | boolean putYuvData\_NV21(String sourceId,  int width,  int height,  int rotation,  boolean flipping,  byte[] data); | |
| 说明 | 将视频数据送入SDK  要求输入格式为YYYY...VUVU...，从android.hardware.Camera获得的数据是NV21格式 | |
| 参数 | sourceId | SDK分配的数据源唯一标识 |
| width | 音频宽度 |
| height | 音频高度 |
| rotation | 视频预览方向 |
| flipping | 采集图像是否为镜像 |
| data | 视频数据 |
| 返回 | 是否发送成功 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **putYuvData\_I420P** | |
| 接口 | boolean putYuvData\_I420P (String sourceId,  int rotation,  boolean flipping,  Image image); | |
| 说明 | 将视频数据送入SDK  要求输入格式为YYYY...VUVU...，从android.hardware.camera2.CameraDevice获得的数据是I420P格式 | |
| 参数 | sourceId | SDK分配的数据源唯一标识 |
| rotation | 视频输入方向 |
| flipping | 采集图像是否为镜像 |
| image | 视频数据载体，内含宽高信息 |
| 返回 | 是否发送成功 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **putYuvData\_I420** | |
| 接口 | boolean putYuvData\_I420 (String sourceId,  int width,  int height,  int rotation,  boolean flipping,  byte[] data); | |
| 说明 | 将视频数据送入SDK  要求输入格式为YYYY...UU...VV... | |
| 参数 | sourceId | SDK分配的数据源唯一标识 |
| width | 音频宽度 |
| height | 音频高度 |
| rotation | 视频预览方向 |
| flipping | 采集图像是否为镜像 |
| data | 视频数据 |
| 返回 | 是否发送成功 | |

### 2.13，视频显示组件

视频显示组件为RtcVideoView：

|  |  |  |
| --- | --- | --- |
| **名称** | **setConfereeLayout** | |
| 接口 | void setConfereeLayout (RtcConfereeLayout layout) | |
| 说明 | 绑定与会者布局信息，当VideoView变为可见时，将自动尝试显示与会者视频画面信息 | |
| 参数 | layout | 与会者布局信息 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **getTextureView** | |
| 接口 | TextureView getTextureView() | |
| 说明 | 调用setConfereeLayout接口后，可通过此接口获得TextureView 实例；  通常只有某些定制设备显示本端窗口时才会使用此接口 | |
| 参数 | 无 |  |
| 返回 | TextureView 实例 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onPause** | |
| 接口 | void onPause() | |
| 说明 | 暂停视频画面更新 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **onResume** | |
| 接口 | void onResume() | |
| 说明 | 恢复画面自动更新 | |
| 参数 | 无 |  |
| 返回 | 无 | |

### 2.14，网络服务

RtcNetworkService目前仅提供网络测试相关功能，接口定义如下

|  |  |  |
| --- | --- | --- |
| **名称** | **enableFlowSaving** | |
| 接口 | void enableFlowSaving(boolean enable) | |
| 说明 | 开/关流量节省模式，开启后会降低通话视频质量，请根据业务妥善使用 | |
| 参数 | enable | true开启省流量模式，false关闭 |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **stopTest** | |
| 接口 | void stopTest() | |
| 说明 | 停止当前正在进行的测试操作 | |
| 参数 | 无 |  |
| 返回 | 无 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **startIcmpTest** | |
| 接口 | void startIcmpTest(String targetIp, int interval, int duration,  String gateway, RtcPingTestCallback callbacks) | |
| 说明 | 向目标地址和网关发起ICMP 测试 | |
| 参数 | targetIp | 目标IP地址 |
|  | interval | Ping间隔时间(单位：ms) |
|  | duration | Ping持续时间(单位：ms) |
|  | gateways | 本地网关地址，可以设置为null |
|  | callback | 测试结果回调 |
| 返回 | True表示开始测试成功，等待callback回调，false表示失败 | |

|  |  |  |
| --- | --- | --- |
| **名称** | **startBandwidthTest** | |
| 接口 | void startBandwidthTest (NetTestServer server,  String gateway, RtcBandwidthTestCallback callbacks) | |
| 说明 | 向目标服务发起带宽测试（需要部署专用服务端配合） | |
| 参数 | server | 带宽测试服务信息，设置为null将使用内置服务 |
|  | gateways | 本地网关地址，可以设置为null |
|  | callback | 测试结果回调 |
| 返回 | True表示开始测试成功，等待callback回调，false表示失败 | |

## 三，相关类定义

### 3.1，RtcToken

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| appid | String | 为app分配的唯一标识 |
| uri | RtcUri | 详见[RtcUri](#_3.6，RtcCallUri) |
| value | String | Token字串值  应由app后台生成，避免appSecret泄露 |

### 3.2，RtcExtend

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| mModel | String | 必填，小鱼音视频后台统计appid下不同类型设备用 |
| displayName | String | 选填，设置内容原样传递到被叫端 |
| callerNumber | String | 选填，设置内容原样传递到被叫端 |
| rotation | Int | 选填，表示定制设备Camera画面预览方向 |
| mediaCustomKey | String | 选填，为应用定制媒体能力时使用 |

### 3.3，RtcAuthListener.AuthStatus

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| success | boolean | 是否认证成功  仅当结果为true，SDK才能正常工作 |
| failCode | AuthFailCode | 认证失败的错误信息 |

### 3.5，RtcUri

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| appid | String | 小鱼为app分配的唯一标识 |
| uid | String | App域的用户唯一标识 |

### 3.6，RtcConferee

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| peerUri | RtcUri | 对端uri，详见[RtcUri](#_3.6，CallUri) |
| userInfo | String | 业务自定义消息，从主叫透传到被叫 |
| autoPstn | boolean | 是否开启自动转呼pstn，仅在呼出时有效，默认为false |
| pstnNum | String | 转呼pstn时的被叫电话号码，仅在呼出时有效 |
| push | boolean | 是否开启被叫不在线时的push操作 |

### 3.7，RtcCallIntent

继承自RtcConferee

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| direction | Direction | 呼叫方向：INCOMING、OUTGOING、CONNECTED |
| from | IncomingFrom | 呼入来源：From\_Normal、From\_Conference、  From\_OtherConference |
| callIndex | int | SDK为呼叫分配的Index，贯穿一次呼叫所有环节 |
| callMode | RtcCallMode | 通话模式，详见[RtcCallMode](#_3.7，RtcCallMode) |
| peerName | String | 呼入时，供显示用的对端名称  来源于对端初始化SDK时设置[RtcExtend](#_3.2，RtcExtend)中的displayName属性 |
| peerNumber | String | 呼入时，供显示用的对端号码  来源于对端初始化SDK时设置[RtcExtend](#_3.2，RtcExtend)中的callNumber属性 |
| statId | String | 业务统计关联字段，用于数据分析 |
| force | boolean | 是否执行强制呼出，设置为true是将挂断所有正在进行的呼叫会话，执行新的呼出操作，默认为false |
| callType | int | 业务自定义通话类型，从主叫透传到被叫 |

### 3.8，RtcMakeCallResult

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| callIndex | int | SDK为呼叫分配的Index，贯穿一次呼叫所有环节 |
| success | boolean | 呼出、是否成功 |
| reason | RtcReason | 呼出、操作失败的系统原因 |
| callMode | RtcCallMode | 通话模式，详见[RtcCallMode](#_3.7，RtcCallMode) |
| peerUri | RtcUri | 对端uri，详见[RtcUri](#_3.6，CallUri) |

### 3.8，RtcConfereeState

继承至RtcMakeCallResult

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| userInfo | String | 邀请失败的用户原因（业务透传） |

### 3.9，RtcCallMode

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| CallMode\_AudioVideo | ENUM\_VALUE | 音视频模式  本端同时接收和发送音频、视频数据 |
| CallMode\_AudioOnly | ENUM\_VALUE | 语音模式  本端只接收和发送音频数据 |
| CallMode\_Observer | ENUM\_VALUE | 观察模式  本端只接收音频、视频数据，不发送音频、视频数据 |
| CallMode\_Observed | ENUM\_VALUE | 被观察模式  本端只发送音频、视频数据，不接收音频、视频数据 |
| CallMode\_Tel | ENUM\_VALUE | 发起/收到PSTN呼叫 |

### 3.10，RtcConfereeLayout

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| id | int | 小鱼后台分配通话与会者ID |
| isAudioMute | boolean | 音频是否Mute |
| isVideoMute | boolean | 视频是否Mute |
| isActiveSpeaker | boolean | 是否发言者，当前音量最大那个 |
| isAudioOnly | boolean | 是否纯音频模式 |
| remoteName | String | 远端名称 |
| remoteUri | String | 远端Uri，详见[RtcUri](#_3.6，CallUri) |
| videoSourceId | String | SDK分配的数据源唯一标识（软解）  用法参见 RtcTextureView |
| videoViewId | int | SDK分配的数据源唯一标识(硬解)  用法参见 demo代码 |
| layoutState | LayoutState | 视频流状态，详见[RtcConfereeLayout.LayoutState](#_3.10，RtcVideoLayout.VideoState) |
| videoMuteReason | VideoMuteReason | 视频Mute原因， 详见[RtcConfereeLayout.VideoMuteReason](#_3.11，RtcVideoLayout.VideoMuteReason) |

### 3.11，RtcConfereeLayout.LayoutState

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| kLayoutStateIdle | ENUM\_VALUE | 空闲  基本上感知不到，很快会转换到其他状态 |
| kLayoutStateObserving | ENUM\_VALUE | 观察中 |
| kLayoutStateMute | ENUM\_VALUE | 视频暂停  可从[VideoMuteReason](#_3.11，RtcConfereeLayout.VideoMuteRea)获知视频暂停原因 |
| kLayoutStateRequesting | ENUM\_VALUE | 视频请求中 |
| kLayoutStateNoBandwidth | ENUM\_VALUE | 本端下行带宽不足，无法接收对端视频 |
| kLayoutStateNoDecoder | ENUM\_VALUE | 本端解码资源不足，无法解码对端视频 |
| kLayoutStateReceived | ENUM\_VALUE | 视频接收并解码完成，可以正常显示s |
| kLayoutStateTelephone | ENUM\_VALUE | 对端正使用运营商电话接入 |
| kLayoutStateAudioOnly | ENUM\_VALUE | 对端正处于CallMode\_AudioOnly模式 |
| kLayoutStateReceivedAudioOnly | ENUM\_VALUE | 本端正处于CallMode\_AudioOnly模式 |
| kLayoutStateAddConferee | ENUM\_VALUE | 应用扩展状态，为开发手动设置 |

### 3.12，RtcConfereeLayout.VideoMuteReason

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| MuteByUser | ENUM\_VALUE | 主动Mute |
| MuteByBWLimit | ENUM\_VALUE | 上行带宽不足 |
| MuteByNoInput | ENUM\_VALUE | 无视频输入 |
| MuteByCameraDisabled | ENUM\_VALUE | 摄像头关闭 |

### 3.13，RtcCallStateInfo

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| callIndex | int | SDK为呼叫分配的Index，标识一次呼叫 |
| meetingId | String | 当前通话会议在小鱼后台的唯一标识 |
| state | CallState | 当前通话状态  详见[RtcCallStateInfo.CallState](#_3.13，RtcCallStateInfo.CallState) |
| reason | RtcReason | 通话断开系统原因 |
| userInfo | String | 通话断开业务原因（业务透传） |

### 3.14，RtcCallStateInfo.CallState

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| CALL\_STATE\_IDLE | ENUM\_VALUE | 空闲  基本上感知不到，很快会转换到其他状态 |
| CALL\_STATE\_PROCESSING | ENUM\_VALUE | SDK向小鱼后台发送offer且已收到后台回复trying时的状态 |
| CALL\_STATE\_OFFERING | ENUM\_VALUE | SDK向小鱼后台发送offer后的状态 |
| CALL\_STATE\_RINGBACK | ENUM\_VALUE | SDK被叫时收到new-call后的状态(振铃) |
| CALL\_STATE\_CONNECTED | ENUM\_VALUE | 呼叫建立完成 |
| CALL\_STATE\_DISCONNECTED | ENUM\_VALUE | 呼叫已断开，此时可通过RtcCallStateInfo.reason获取断开原因 |
| CALL\_STATE\_EARLY\_MEDIA | ENUM\_VALUE | 媒体就绪，表示有通话前置媒体需要播放，如PSTN回铃音 |

### 3.15，RtcAudioSampleInfo

|  |  |  |
| --- | --- | --- |
| **属性** | **类型** | **说明** |
| sampleRateInHz | int | 采样率，如16000、48000 |
| sampleBits | int | 采样位数，仅支持16bit |
| channels | int | 采样通道数，如1、2 |
| channelIndex | Int | 指定通话使用的采样通道，默认为0 |
| frameDuration | int | 采样帧时长，10ms，20ms(为10ms的整数倍数) |
| frameBufferSize | int | 采样信息，描述的pcm帧buffer长度 |

## 四，小鱼音视频通话SDK调用逻辑

### 4.1，逻辑示例（完整流程请参考Demo）

//1. 实现RtcContext.Callback回调接口并创建回调实例

AppCallback callback;

//2. 创建RtcContext

Builder builder = RtcContext.Builder.create(appCtxt);

builder.setToken(appToken);

//builder. setExtend(appExtend);

builder.setCallback(callback);

builder.build();

//2.Rtc鉴权通过，收到回调后

//获取RtcContext

RtcContext rtcContext = RtcContextGetter.getRtcContext();

//设置通话状态回调  
rtcContext.setCallStatusListener(this);  
  
//3. 获取callService  
CallService callService = rtcContext.getCallService();

//发起呼叫

callService.makeCall(calleeUri);

//停止呼叫

callService.dropCall();