

一，相关接口说明：

1.Bool initialize(String appid)

初始化SDK。

参数一：Appid：应用id，由三体云后台分配，区别不同应用的唯一标识

输出：true 表示初始化成功，false 表示失败

2、String getVersion()

获取接口版本号

输入参数：无

输出：返回SDK版本信息

3、Bool destroy（）

注销SDK。

输入：无

输出：true 表示成功，false表示失败

4、Int joinChannel(string channelkey, string channelname, Number userId)

加入房间

参数一：channelkey，频道key

参数二：channelname，频道名称，用户加入同一个频道的唯一标识

参数三：userId，用户ID，唯一标识；默认为0，由系统自动分配

输出：0表示成功，其他表示失败

5、Bool leaveChannel（）

离开当前频道

输入：无

输出：true表示成功。

6、Int setChannelProfile（RoomMode_TTT profile）

设置频道模式

参数一：profile，频道模式，具体取值参考如下数据结构：

```
typedef enum RoomModeTTT
```

```
{
```

```
ROOMMODECOMMUNICATIONTTT, // 互动模式
```

```
ROOMMODELIVETTT, // 直播模式
```

```
ROOMMODEGAMEFREETTT, // 游戏自由模式
```

```
ROOMMODEGAMECOMMANDTTT, // 游戏指挥模式
```

```
ROOMMODECONFERENCETTT, // 会议模式
```

```
ROOMMODEMANUALTTT, // 手动模式
```

```
ROOMMODEUNFINETTT, // 未定义模式
```

```
}RoomMode_TTT;
```

输出：0表示成功，其他表示失败，

7、Int setClientRole（int role）

设置用户的角色

输入：role，具体取值参考如下

```
enum CLIENTROLETYPETTT
```

```
{ CLIENTROLEBROADCASTERTTT = 1,
CLIENTROLEANCHOR_TTT = 2,
/**
 * 观众角色(默认)
 */
CLIENTROLEAUDIENCE_TTT = 3,
};
输出：0表示成功，其他表示失败
```

8、Int setupLocalVideo（int renderMode， Uint64 uid）

设置本地视频显示模式

参数一：renderMode，具体取值参考如下：

参数二：uid，本地视频对应的用户id

输出：0表示成功，其他表示失败

9、Int enableVideo()

打开视频模式。可以在加入频道前或者通话中调用，在加入频道前调用，则自动开启视频模式，在通话中调用则由音频模式切换为视频模式。调用 disableVideo() 方法可关闭视频模式。

输入：无

输出：0表示成功，其他表示失败

10、Bool startPreview()

启动本地视频预览。在开启预览前，必须先调用 setupLocalVideo 设置预览窗口及属性，且必须调用 enableVideo() 开启视频功能

输入：无

输出：true表示开启成功

11、Bool stopPreview()

停止本地视频预览

输入：无

输出：true表示成功

12、Bool setVideoProfile(int profile, Bool swapWidthAndHeight)

设置视频编码属性（Profile）。每个属性对应一套视频参数，如分辨率、帧率、码率等。当设备的摄像头不支持指定的分辨率时，SDK 会自动选择一个合适的摄像头分辨率，但是编码分辨率仍然用 setVideoProfile() 指定的

参数一：profile，预设的视频参数，具体取值参考如下：

视频属性 枚举值 分辨率(宽x高) 帧率(fps) 码率(kbps)

VIDEOPROFILE120P 110 160120 15 651000

VIDEOPROFILE180P 111 320180 15 1401000

VIDEOPROFILE240P 112 320240 15 2001000

VIDEOPROFILE360P 113 640360 15 4001000

VIDEOPROFILE480P 114 640480 15 5001000

VIDEOPROFILE720P 115 1280720 15 11301000

VIDEOPROFILE1080P 116 19201080 15 20801000

参数二：swapWidthAndHeight，是否将视频宽高互换

输出：true表示成功，false表示失败

13、Int setupRemoteVideo(int renderMode, uint64 uid, string devId)

设定远端用户的视频显示模式

参数一：renderMode，具体取值参考如下：

1 视频放缩后充满窗体； 2 视频全部显示，空出的窗体部分补黑边

参数二：uid，远端视频对应的用户ID

参数三: devId, 视频设备ID
输出: 0表示成功, 其他表示失败

14、Bool enableLastmileTest()

启动网路测试
输入: 无
输出: true表示成功, false表示失败

16、Bool disableLastmileTest()

停止网路测试
输入: 无
输出: true表示成功, false表示失败

17、Int disableVideo()

关闭视频模式。可以在加入频道前或者通话中调用, 在加入频道前调用, 则自动开启纯音频模式, 在通话中调用则由视频模式切换为纯音频模式。
调用 enableVideo() 方法可开启视频模式。
输入: 无
输出: 0表示成功, 其他表示失败

18、Int enableAudio()

打开音频通话模式,
输入: 无
输出: 0表示成功, 其他表示失败

19、Int disableAudio()

关闭音频通话模式。关闭后, 本次通话为纯视频模式
输入: 无
输出: 0表示成功, 其他表示失败

20、Bool startScreenCapture(int left, int top, int right, int bottom)

启动屏幕分享功能, 通过参数设定分享的屏幕矩形的坐标点
参数一: left, 左顶点的x坐标
参数二: top, 左顶点的y坐标
参数三: right, 右下角顶点的x坐标
参数四: bottom, 右下角顶点的y坐标
输出: 0表示成功, 其他表示失败

21、Int stopScreenCapture()

停止屏幕分享
输入: 无
输出: 0表示成功, 其他表示失败

22、Int configPublisher (string url)

设置推流地址, 允许服务器把频道中的音视频推送到直播服务器上, 进行直播。
输入: url, 直播服务器地址, 可以是第三方的直播服务器或者CDN服务器。
输出: 0表示成功, 其他表示失败

23、Bool muteLocalAudioStream(Bool mute)

静音/取消静音。该方法用于允许/禁止往网络发送本地音频流。

输入: mute, true表示静音, false表示取消静音

输出: true表示成功, false表示失败

24、Bool muteAllRemoteAudioStreams(bool mute)

允许/禁止接收和播放远端用户的音频流, 即对所有远端用户进行静音与否。

输入: mute, true表示静音, false表示取消静音

输出: true表示成功, false表示失败

25、Bool muteRemoteAudioStream(uint64 uid, bool mute)

静音指定远端用户/对指定远端用户取消静音。

参数一: uid, 指定用户的id

参数二: mute, true表示静音, false表示取消静音

输出: true表示成功, false表示失败

26、Bool muteLocalVideoStream(bool mute)

打开/关闭本地视频流数据, mute为true是表示本地视频流不上行, 远端客户将无法看到这个用户的视频

输入: mute, true表示关闭本地视频流, false表示打开本地 视频流

输出: true表示成功, false表示失败

27、Bool muteAllRemoteVideoStreams(bool mute)

允许/禁止接收和播放所有人的视频流。

输入: mute, true表示停止所有人视频流, false表示开始接收所有人的视频流

输出: true表示成功, false表示失败

28、Bool enableAudioVolumeIndication(int interval, int smooth)

启用音量提示回调

参数一: interval, 指定音量提示的时间间隔。小于0禁用音量提示功能, 单位为毫秒

参数二: smooth, 平滑系数。默认可以设置为3

输出: true表示成功, false表示失败

29、Int startAudioMixing(string filepath, bool lookback, bool replace, int cycle)

指定本地音频文件来和麦克风采集的音频流进行混音和替换(用音频文件替换麦克风采集的音频流), 可以通过参数选择是否让对方听到本地播放的音频和指定循环播放的次数。

参数一: filepath, 指定需要混音的本地音频文件名和文件路径。支持以下音频格式: mp3, aac, m4a, 3gp, wav, flac

参数二: loopback, True只有本地可以听到混音或替换后的音频流, False本地和对方都可以听到混音或替换后的音频流

参数三: replace, True音频文件内容将会替换本地录音的音频流, False音频文件内容将会和麦克风采集的音频流进行混音

参数四: 指定音频文件循环播放的次数。正整数代表循环的次数

输出: 0表示成功, 其他表示失败

30、Bool stopAudioMixing()

停止文件混音

输入: 无

输出: true 表示成功, false表示失败

31、Bool pauseAudioMixing()

暂停文件混音

输入: 无

输出：true 表示成功，false表示失败

32、Bool resumeAudioMixing()

恢复文件混音

输入：无

输出：true 表示成功，false表示失败

33、Bool adjustAudioMixingVolume(int volume)

调整文件伴奏混音的音量，

输入：无

输出：true 表示成功，false表示失败

34、Int getAudioMixingDuration()

获取伴奏文件的长度

输入：无

输出：伴奏文件的时长

35、Int getAudioMixingCurrentPosition()

获取伴奏文件的当前进度

输入：无

输出：返回非0的整形。

36、Bool setAudioMixingPosition(int pos)

调整当前伴奏文件的进度

输入：pos，新的进度

输出：true 表示成功，false表示失败

37、Int setHighQualityAudioParamters(bool enable)

设置音频高音质选项。切勿在加入频道后再次调用本方法。

输入：enable，true表示启用，false表示关闭

输出：0表示成功，其他表示失败

38、Bool resizeReview（）

所有视频充满窗体

输入：无

输出：true表示成功，false表示失败

39、Bool muteRemoteVideoStream(uint64 uid, bool mute)

允许/禁止播放指定的远端视频流。

参数一：uid，指定远端用户的id

参数二：mute，true表示禁止，false表示允许

输出：true表示成功，false表示失败

40、Bool muteRemoteSpeaking(uint64 uid, bool mute)

将某个远端用户静音，其他所有用户都无法听到该用户的说话

参数一：uid，指定远端用户的id

参数二：mute，true表示静音，false表示解除静音

输出：true表示成功，false表示失败

41、Bool kickChannelUser (uint64 uid)

将某个用户请出直播房间

输入: uid, 即将被移除房间的用户id

输入: true表示成功, false表示失败

42、Int setSpeakerphoneVolume(int volume)

调整本地麦克风的音量

输入: volume, 新的音量

输出: 0表示成功, 其他表示失败

43、Int sendChatMessage(uint64 uid, int type, string msg)

发送聊天消息

参数一: uid, 消息发送目标id

参数二: type, 消息类型: 文字, 图片, 语音

参数三: msg, 即将发送的消息

输出: 0表示成功, 其他表示失败

44、Bool startRecordingDeviceTest(int indicationInterval)

测试麦克风设备测试录音音频设备, 知道它是否能正常工作。

输入: indicationInterval, 本地采集声音音量上报回调的时间间隔

输出: true表示成功, false表示失败

45、Bool stopRecordingDeviceTest()

停止录制音频设备测试

输入: 无

输出: true表示成功, false表示失败

46、Bool startPlaybackDeviceTest(string filepath)

开始回放音频文件, 测试扬声器是否正常

输入: filepath, 音频文件的路径

输出: true表示成功, 其他表示失败

47、Bool stopPlaybackDeviceTest()

停止回放音频文件

输入: 无

输出: true表示成功, 其他表示失败

48、Bool startVideoDeviceTest()

开始视频设备测试

输入: 无

输出: true表示成功, 其他表示失败

49、Bool stopVideoDeviceTest()

停止视频测试测试

输入: 无

输出: true表示成功, 其他表示失败

50、int GS_GetNumberOfVideoDevices()

获取系统视频采集设备的数量

输入：无

输出：视频设备数量

51、String GS_GetVideoDeviceName(int index)

获取指定索引对应的设备名称

输入：index，设备列表索引

输出：设备名称

52、Int GS_SetDeviceCaptureParams(string wcharname, int width, int height, int frames, int bitrate)

设置本地采集参数

参数一：视频设备名称

参数二：宽度

参数三：高度

参数四：帧率

参数五：码率

输出：0表示成功，其他表示失败

53、Int GS_SetDefaultDevice(string devicename)

设置默认设备

输入：devicename，设备名称

输出：0表示成功，其他表示失败

54、Int setSpeakerVolume(int volume)

设置扬声器音量

输入：volume，音量值

输出：0表示成功，其他表示失败

55、Int setMicVolume (int volume)

设置麦克风音量 输入：volume，音量值 输出：0表示成功，其他表示失败

56、Int getNumOfRecordingDevices()

获取录音设备数量

输入：无

输出：返回值<0表示失败；返回值>0表示成功

57、Int getNumOfPlayoutDevices()

获取扬声器设备数量

输入：无

输出：返回值<0表示失败；返回值>0表示成功

58、String getRecordingDefaultDevice()

获取默认录音设备的id

输入：无

输出: json格式的字符串, 格式如下: { "errCode": "0", //0表示成功, 其他表示失败
"deviceId": "设备名称"
}

59、String getPlaybackDefaultDevice()

获取默认扬声器的ID

输入: 无

输出: json格式的字符串, 格式如下:

```
{  
"errCode": "0", //0表示成功, 其他表示失败  
"deviceId": "设备名称"  
}
```

60、String getRecordingDevice(int index)

根据索引获取录音设备的名称和ID

输入: index, 需要获取的设备索引

输出: json格式的字符串, 格式如下:

```
{  
"errCode": "0", //0表示成功, 其他表示失败  
"deviceId": "设备ID",  
"deviceName": "设备名称"  
}
```

61、String getPlaybackDevice(int index)

更具索引获取播放设备的信息

输入: index, 设备索引

输出: json格式的字符串, 格式如下:

```
{  
"errCode": "0",  
"deviceId": "设备名称",  
"deviceName": "liyong de diannao"  
}
```

62、Int setRecordingDevice(string deviceId)

设置录音设备

输入: deviceId、, 录用设备的ID

输出: 0表示成功, 其他表示失败

63、Int setSpeakerDevice(string deviceId)

设置放音设备

输入: deviceId, 放音设备的id

输出: 0表示成功, 其他表示失败。

二，事件通知

sdk的不同的事件通知发送给js不同的json串。事件通知对应的json串如下。

加入频道回调

```
{ "callbackname": "onJoinChannelSuccess", "channel": "123456", "uid": "123456" }
```

表示客户端已经成功进入指定房间。

名称 描述 channel 频道名称 uid 登录的用户ID

发生错误回调

```
{ "callbackname": "onError", "err": 203 }
```

表示 SDK 运行时出现了（网络或媒体相关的）错误。

名称 描述
err 错误码
错误码：

名称 错误码 原因描述

- ERRORKICKBY_HOST 201 被房主移出房间
- ERRORKICKBY_PUSHRTMPFAILED 202 RTMP推流失败
- ERRORKICKBY_SERVEROVERLOAD 203 服务器过载
- ERRORKICKBYMASTEREXIT 204 房主已经离开房间
- ERRORKICKBY_RELOGIN 205 重复登录
- ERRORKICKBY_NOAUDIODATA 206 长时间没有上行音频数据
- ERRORKICKBY_NOVIDEODATA 207 长时间没有上行视频数据
- ERRORKICKBY_NEWCHAIRENTER 208 其他人以主播身份进入房间
- ERRORKICKBY_CHANNELKEYEXPIRED 209 CHANNEL KEY失效了

离开频道回调

```
{ "callbackname": "onLeaveChannel" }
```

说话声音音量提示回调

```
{ "callbackname": "onAudioVolumeIndication", "nUserID": "1234", "audioLevel": 3, "audioLevelFullRange": 3434 }
```

提示谁在说话及其音量。默认禁用。可以通过 enableAudioVolumeIndication 方法设置。

名称 描述 nUserID 说话者的用户ID audioLevel 说话者的音量，在0-9之间 audioLevelFullRange 说话者的音量，音量范围更大0-32767.5之间

其他用户加入当前频道回调

```
{ "callbackname": "onUserJoined", "uid": 3453, "identity": 1 }
```

提示有主播加入了频道。如果该客户端加入频道时已经有人在频道中，SDK 也会向应用程序上报这些已在频道中的用户

名称 描述 uid 加入频道的用户ID identity 加入频道的用户的身份，副播或观众

其他用户离开当前频道回调

```
{ "callbackname": "onUserOffline", "uid": 3453, "reason": 1 }
```

提示有主播离开了频道。

名称 描述 uid 离开频道的用户ID reason 该用户离开频道的原因

统计数据回调

```
{ "callbackname": "onRtcStats", "rxAudioKBitRate": 3453, "rxBytes": 1, "rxVideoKBitRate": 1, "txAudioKBitRate": 1, "txBytes": 1, "txVideoKBitRate": 1 root["users"] = 2344 }
```

txBytes 发送的总字节数

rxBytes 接收的总字节数
rxAudioKBitRate 当前发送音频的比特率
rxAudioKBitrate 当前接收音频的比特率
txVideoKBitRate 当前发送视频的比特率
rxVideoKBitrate 当前接收视频的比特率
user 参与直播的用户总数

连接丢失回调

```
{ "callbackname": "onConnectionLost" }
```

该回调方法表示 SDK 和服务器失去了网络连接。

本地视频显示回调

```
{ "callbackname": "onFirstLocalVideoFrame" }
```

提示第一帧本地视频画面已经显示在屏幕上。

名称 描述
width 视频流宽(像素) height 视频流高(像素)

远端视频显示回调

```
{ "callbackname": "onFirstRemoteVideoFrame" }
```

第一帧远程视频显示在视图上时，触发此调用

名称 描述
uid 加入频道的用户ID width 视频流宽(像素) height 视频流高(像素)

远端视频接收解码回调

```
{ "callbackname": "onFirstRemoteVideoDecoded" }
```

收到第一帧远程视频流并解码成功时，触发此调用。

名称 描述
uid 加入频道的用户ID
width 视频流宽(像素)
height 视频流高(像素)

其他用户启用/关闭视频

```
{ "callbackname": "onUserEnableVideo" }
```

提示有其他用户启用/关闭了视频功能。关闭视频功能是指该用户只能进行语音通话，不能显示、发送自己的视频，也不能接收、显示别人的视频。

名称 描述
uid 用户ID
enabled 用户已启用/关闭了视频功能

本地视频统计回调

```
{ "callbackname": "onLocalVideoStats" }
```

名称 描述

stats 本地视频相关的统计信息
报告更新本地视频统计信息，该回调函数每两秒触发一次

远端视频统计回调

```
{ "callbackname": "onRemoteVideoStats" }
```

报告更新远端视频统计信息，该回调函数每两秒触发一次。

名称 描述 stats 远端视频的统计信息

收到聊天消息回调

`{ "callbackname" : "OnChatMessageRecived" }`
收到其他用户发来的消息时触发此回调。

名称 描述
nSrcUserID 消息发送方userID

type 消息类型

strData 消息内容