场景融合
视频直播
Web 1.X

2024-08-30

视频直播概述

更新时间:2024-08-30

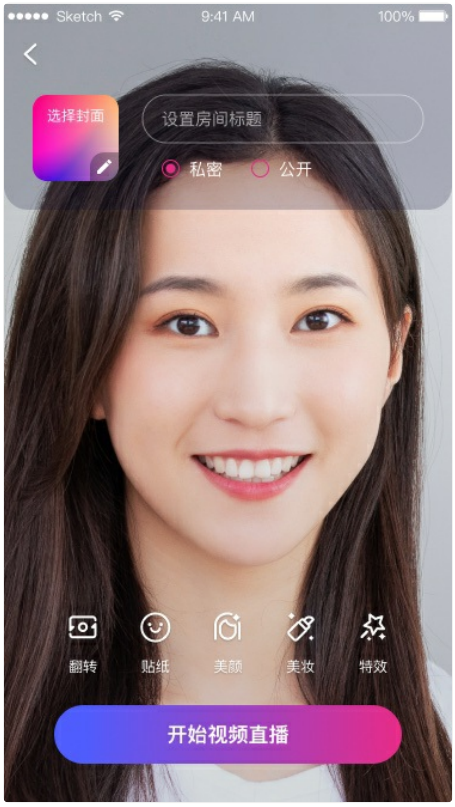
RCLiveRoomLib 是融云针对视频直播场景设计的 SDK，API 设计参考了市面主流视频直播 App 的功能和场景特点。SDK 融合 **RongIMLib** 和 **RongRTCLib** 实现视频直播场景，将复杂逻辑（订阅和发布流、上下麦、连麦布局等）进行了封装，降低开发者的接入难度。相较于开发者自己接入低延迟直播和即时通讯，SDK 具备以下优势：

1. **贴近业务：** API 调用和命名贴近视频直播业务场景与客户端功能特性；
2. **使用简单：** 核心 API 数量少（房主 3 个、观众 2 个和常规操作 13 个），学习成本低，可快速上手；
3. **扩展性强：** 支持房间自定义属性，提高扩展性，不管是游戏直播、社交直播，还是电商直播等场景均可覆盖；

主要功能

视频直播： RCLiveRoomLib 只需两步即可开启直播：

1. 准备视频直播：设置视频信息，输出视频流；
2. 开启视频直播：加入房间并发布视频直播流。



视频连麦： RCLiveRoomLib 支持邀请上麦和申请上麦两种连麦机制。SDK 封装了连麦布局、布局结果回调和事件点击。

房间管理： 视频直播包含丰富的房间属性，RCLiveRoomLib支持自定义属性。

功能列表

RCLiveRoomLib 的功能包括但不限于以下内容：

功能	描述
视频流预处理	直播推流前可以对视频流进行处理，比如：美颜

功能	描述
支持1对1连麦	连麦用户支持小窗事件点击和区域位置通知
申请连麦	轻松处理申请机制，特定用户允许才可上麦
取消连麦申请	观众取消连麦申请，房主接收回调
同意连麦申请	房主同意连麦申请，并与观众开始连麦
拒绝连麦申请	房主拒绝连麦申请，观众接收回调
查询连麦申请	查询当前房间上麦申请信息
邀请上麦	房主邀请观众进行视频连麦
取消上麦邀请	房主取消连麦邀请，观众接收到回调
同意上麦邀请	观众同意连麦邀请，并与房主开始连麦
拒绝上麦邀请	观众拒绝连麦邀请，房主接收到回调
查询上麦邀请	查询当前房间上麦邀请信息
自定义房间属性	丰富的房间扩展属性，轻松构建不同类型的视频直播房间

开通融云 CDN 服务

更新时间:2024-08-30

欢迎前往融云官网了解直播 CDN 产品。本文介绍如何开始配置融云 CDN 服务。

开通服务

融云 CDN 是付费增值服务，且需要开通后才能使用。您可以访问控制台的融云 CDN 页面开通服务。

开通融云 CDN 服务后，可看到域名配置、证书管理、防盗链配置。

选择推拉流模式

融云 CDN 支持三种不同的推拉流地址模式：

- **开播自动推流**：在后台配置后，所有直播房间的直播流都会自动推流到 CDN，观众可在客户端加入房间时获取 CDN 信息，或者使用客户端 SDK 提供的监听房间内 CDN 资源的方法。如果您希望直接从融云服务端获取拉流地址，可从 App 服务端调用融云服务端 API，详见获取融云 CDN 拉流地址。
- **开播手动推流**：在直播主播开始推流后，您需要根据产品设计决定何时调接口让融云服务开始或停止推流到 CDN。观众可在客户端加入房间时获取 CDN 信息，或者使用客户端 SDK 提供的监听房间内 CDN 资源的方法。如果您希望直接从融云服务端获取拉流地址，可从 App 服务端调用融云服务端 API，详见获取融云 CDN 拉流地址。
- **自行生成推拉流地址**：融云服务端不负责生成推拉流地址，您需要在您的应用服务器自行生成 CDN 推拉流地址。

自行拼接推拉流地址

在**自行生成推拉流地址**模式下，可自行拼接生成 CDN 推拉流地址。需要拼接的字段包括推/拉流域名、自定义的 {AppName}

和自定义的 {StreamName}。您还可以通过在 CDN 拉流地址中添加分辨率、码率参数 {with}、{height}、{fps}，拉取 CDN 转码流。

CDN 推流地址拼接规则

融云 CDN 仅支持 RTMP 协议的推流：

RTMP：rtmp://推流域名/{AppName}/{StreamName}

拼接推流地址后，可调用客户端 SDK 或服务端 API 的旁路推流接口，传入 CDN 推流地址进行推流。

CDN 拉流地址拼接规则

融云 CDN 支持使用 RTMP、FLV、HLS 协议进行拉流。

如需使用 CDN 流的原始分辨率、帧率：

- **RTMP**：`rtmp://拉流域名/{AppName}/{StreamName}`
- **FLV**：`http(s)://拉流域名/{AppName}/{StreamName}.flv`
- **HLS**：`http(s)://拉流域名/{AppName}/{StreamName}.m3u8`

如需拉取不同分辨率的 CDN 直播流（拉取非原始分辨率、码率的流会触发 CDN 转码服务，产生转码费用）：

- **RTMP**：`rtmp://拉流域名/{AppName}/{StreamName}_{with}_{height}_{fps}`
- **FLV**：`http(s)://拉流域名/{AppName}/{StreamName}_{with}_{height}_{fps}.flv`
- **HLS**：`http(s)://拉流域名/{AppName}/{StreamName}_{with}_{height}_{fps}.m3u8`

{with}、{height} 参数为要拉取的 CDN 转码流的分辨率宽高，参考下方枚举值。{fps} 为帧率参数，支持 10/15/24/30。

宽高限制列表：

```
[
"176*144", "180*180", "256*144", "240*180", "320*180", "240*240", "320*240",
"360*360", "480*360", "640*360", "480*480", "640*480", "720*480", "848*480",
"960*720", "1280*720", "1920*1080", "144*176", "144*256", "180*240", "180*320",
"240*320", "360*480", "360*640", "480*640", "480*720", "480*848", "720*960",
"720*1280", "1080*1920"
]
```

例如，拉取 flv 协议的 CDN 流，并指定宽x高为 720*920，帧率为 15，则拼接后的地址如下：

```
http(s)://yourdomain/apppkey/roomid_720_960_15.flv
```

推荐使用融云 CDN 播放器进行播放。

> **注意：**

- 如在控制台启用了防盗链，则您拼接的推拉流地址中还必须携带防盗链信息。防盗链地址的具体拼接规则请参见下文「防盗链配置」。
- 2022 年 6 月前开通融云 CDN 服务的客户，如获取到的 HLS 拉流地址无法正常播放，请提交工单咨询。

域名配置

在域名配置标签下配置并保存了推流域名和拉流域名后，会看到完整的配置界面。

配置推拉流域名

推拉流域名都需要填写二级域名，请确保此域名没有没有在别的 App Key 下配置过。

- 域名是成对的，推流和拉流有绑定关系，新增和修改时需要一并修改。
- 您需要确保一级域名已经备案过。如果因域名没有备案或涉及非法业务等原因导致推流或者拉流域名不可用，由此产生的任何后果由您自身承担。

推流与拉流域名配置生效后，系统会自动给该域名分配一个 CNAME。使用您的推流或拉流域名进行直播之前，需要需要您公司开发或服务运维人员在域名解析配置中添加对应的 CNAME 记录，让您的域名指向 CNAME。具体如何配置可以咨询域名供应商。

HTTPS 设置

HTTPS 设置是可选项。FLV/HLS 协议的直播流默认使用 HTTP，配置 SSL 证书后可以使用 HTTPS。

如果您没有在融云平台上传过证书，请点击**新增证书**，将证书上传。下图展示了提交证书的界面。

防盗链

防盗链配置是可选项。配置后会提升推拉流域名的安全。

开启后一定要配置推流和拉流地址有效期时间和加密 URL 的密文 KEY。

防盗链开启后，如果观众端遇到弱网（融云 SDK 内部帮您在网络恢复后重新订阅成功）会比没有防盗链多出一些恢复订阅时间。

自行生成防盗链推拉流地址

如果您选择的推拉流模式为**自行生成推拉流地址**，在您的服务端生成推拉流地址时在地址中加入防盗链相关信息。

相比普通推拉流地址，生成防盗链推拉流地址还需要用到以下防盗链参数：

防盗链参数	描述	补充说明
wsTime	生成推拉流地址时的 UNIX 时间。防盗链地址中直接携带该参数。	格式为 16 进制 UNIX 时间。防盗链地址中直接携带该参数。
KEY	MD5 计算方式的密钥，在控制台配置，仅用于计算 `wsSecret`。	可以自定义。
wsSecret	播放 URL 中的加密参数。防盗链地址中直接携带该参数。	值是通过将 `KEY`，`URI`，`wsTime` 依次拼接的字符串进行 MD5 加密算法得出，即 wsSecret = MD5（wsTime+URI+KEY）。

防盗链的推拉流地址生成示例：

假设原始 url（已包含推/拉流域名 + AppName + StreamName）为：http://cdn.rongcloud.cn/myappname/stream.flv

1. 获取在控制台**防盗链配置**填入的 KEY，假设 KEY 为：`rongcloud`。

2. 获取 wsTime，即生成推拉流地址时的 UNIX 时间，假设为 1601026312，转换成 16 进制5f6db908。在计算 wsSecret 时需要用到。防盗链 URL 中也会直接携带该参数。

3. 计算 wsSecret。防盗链 URL 中会直接携带该参数。

   1. 获取计算 `wsSecret` 需要用到的 `URI` 参数值。需要用到您自定义的 AppName 与 StreamName。拼接规则为 `/{AppName}/{StreamName}`。从本例的原始 URL 可得出 URI 为 `/app/stream.flv`。
   2. 依次拼接 `wsTime` + `URI` + `KEY`，获取签名字符串。在本例中该拼接字符串为：5f6db908/app/stream.flvrongcloud
   3. 对签名字符串计算 md5hash，得到 `wsSecret`。即，`wsSecret` = md5sum("5f6db908/app/stream.flvrongcloud") = 79aead3bd7b5db4adeffb93a010298b5

4. 使用上述步骤中得到的 wsSecret 与 wsTime 参数，拼接成防盗链 URL：

   http://cdn.rongcloud.cn/app/stream.flv?wsSecret=79aead3bd7b5db4adeffb93a010298b5&wsTime=5f6db908

当用户发起带时间戳防盗链的 url 请求后，CDN 服务器会对 url 内容进行校验，判断时间有效性及 MD5 值，两个值其中有一个不符合要求，返回 403。

生成防盗链代码示例（golang）

```
package main

import (
"crypto/md5"
"encoding/hex"
"fmt"
"strconv"
"strings"
"time"
)

func main() {
secret, t := GenWsSecret("/live/abc", "key")
fmt.Println(secret)
fmt.Println(t)
}

func GenWsSecret(uri, key string) (string, string) {
uri = fmt.Sprintf("/%s", strings.TrimLeft(uri, "/"))

t := strconv.FormatInt(time.Now().Unix(), 16)

ori := fmt.Sprintf("%s%s%s", t, uri, key)

h := md5.New()
h.Write([]byte(ori))

return hex.EncodeToString(h.Sum(nil)), t
}
```

> **注意事项：**
>
> 1. 在 20：00 至次日 2：00 进行的配置不会立即生效，会在 2 点后依次配置生效。在此时间段外进行配置会

在 30 分钟后生效。

2. 新配置生效后，原有的配置即刻作废。建议开发者避开业务高峰时段，并给还在使用老配置的房间用户发送提醒通知，重新订阅新地址。

快速集成 环境要求

更新时间:2024-08-30

- **node**：14.0.0 以上

开通音视频服务

您在融云创建的应用默认不会启用音视频服务。在使用融云提供的任何音视频服务前，您需要前往控制台，为应用开通音视频服务。

使用语聊房业务要求开通「音视频直播」服务，具体步骤请参阅 开通音视频服务 。

> 服务开通、关闭等设置完成后 30 分钟后生效。

导入 SDK

直播房业务依赖即时通讯业务提供的信令通道。基于 RCLiveRoomLib 开发应用，您必须安装以下库：

- 即时通讯基础能力库 Engine、IMLib
- 实时音视频基础能力库 RTCLib
- 直播房能力库 RCLiveRoomLib

> 从 1.1.0 版本开始，直播房 SDK 依赖的 Engine、IMLib 和 RTCLib SDK 需要单独安装及初始化。如果您从 1.1.0 以下版本升级，请务必注意修改导入及初始化代码。

安装 Engine

要求安装 RongEngine 5.1.2 或之后版本。以下安装最新版：

```
# 安装 RongEngine v5
npm install @rongcloud/engine --save
```

安装 IMLib

要求安装 IMLib 5.1.2 或之后版本。以下安装最新版：

```
# 安装 RongIMLib v5
npm install @rongcloud/imlib-next --save
```

安装 RTCLib

要求安装 RTCLib 5.3.1 或之后版本。以下安装最新版：

```
# 安装 RTCLib
npm install @rongcloud/plugin-rtc --save
```

安装 RCLiveRoomLib

```
# 安装 RCLiveRoomLib
npm install @rongcloud/rcliveroomlib --save
```

代码集成

```
// 非 ESModule
const RongEngine = require('@rongcloud/engine')
const RongIMLib = require('@rongcloud/imlib-next')
const RongRTCLib = require('@rongcloud/plugin-rtc')
const RCLiveRoomLib = require('@rongcloud/rcliveroomlib')
// ESModule
import * as RongEngine from "@rongcloud/engine";
import * as RongIMLib from "@rongcloud/imlib-next";
import * as RongRTCLib from "@rongcloud/plugin-rtc";
import RCLiveRoomLib from '@rongcloud/rcliveroomlib';
```

> Web 端 RCLiveRoomLib 依赖于融云 Engine 、IMLib 与 RTCLib，兼容参考 IMLib 兼容说明 和 RTCLib 浏览器支持清单 。

集成后的包大小

集成视频直播 SDK 后，会增加一定的包大小。

Web 端： 增量大约 200 KB

版本依赖说明

RCLiveRoomLib 依赖融云 Engine 、 IMLib 与 RTCLib。

从 RCLiveRoomLib v1.1.0 开始，依赖版本如下。

依赖组件	版本
Engine	5.1.2 及以上
IMLib	5.1.2 及以上
RTCLib	5.3.1 及以上

依赖组件	版本

初始化

更新时间:2024-08-30

> 从 1.1.0 版本开始，直播房 SDK 依赖的 Engine、IMLib 和 RTCLib SDK 需要单独安装及初始化。如果您从 1.1.0 以下版本升级，请务必注意修改导入及初始化代码。

> 在使用 RCLiveRoomLib 之前，请确保已完成以下操作：
>
> - 您已开通融云开发者账号，并申请了融云 App Key。
> - 您已为 App Key 开通音视频直播服务。

RCLiveRoomLib 初始化

调用 RCLiveRoomLib 的初始化接口时，务必保证 RongIMLib 、RongRTCLib 已初始化完毕，且 RCLiveRoomLib 初始化接口在应用全生命周期内仅被调用一次。

```
// RongIMLib 初始化,请务必保证此过程只被执行一次
RongIMLib.init({appkey:'<Your-App-Key>'});
// RongRTCLib 初始化,请务必保证此过程只被执行一次
const rtcClient = RongIMLib.installPlugin(RongRTCLib.installer,{});
// RCLiveRoomLib 初始化,请务必保证此过程只被执行一次
RCLiveRoomLib.init({
RongIMLib: RongIMLib,
RongRTCLib: rtcClient,
});
```

建立连接

App Key 是应用的唯一性标识，Token 则是用户的唯一性标识，是用户连接融云 IM 服务所必需的身份凭证。Token 一般由开发者的应用服务器调用融云 Server API 获取 Token 接口获取之后，由应用服务器下发到应用客户端。

> 以下示例代码假定客户端已获取 Token。
> 从 1.1.0 版本开始，直播房 SDK 改为使用 RongIMLib 的 connect() 方法。如果您从 1.1.0 以下版本升级，请注意修改连接代码。

调用 RongIMLib 的 connect() 方法：

```
RongIMLib.connect('<Your-Token>')
```

后续所有代码示例中的 RCLiveRoomLib 均指通过初始化并连接服务后的 RCLiveRoomLib 实例对象

开启和结束视频直播

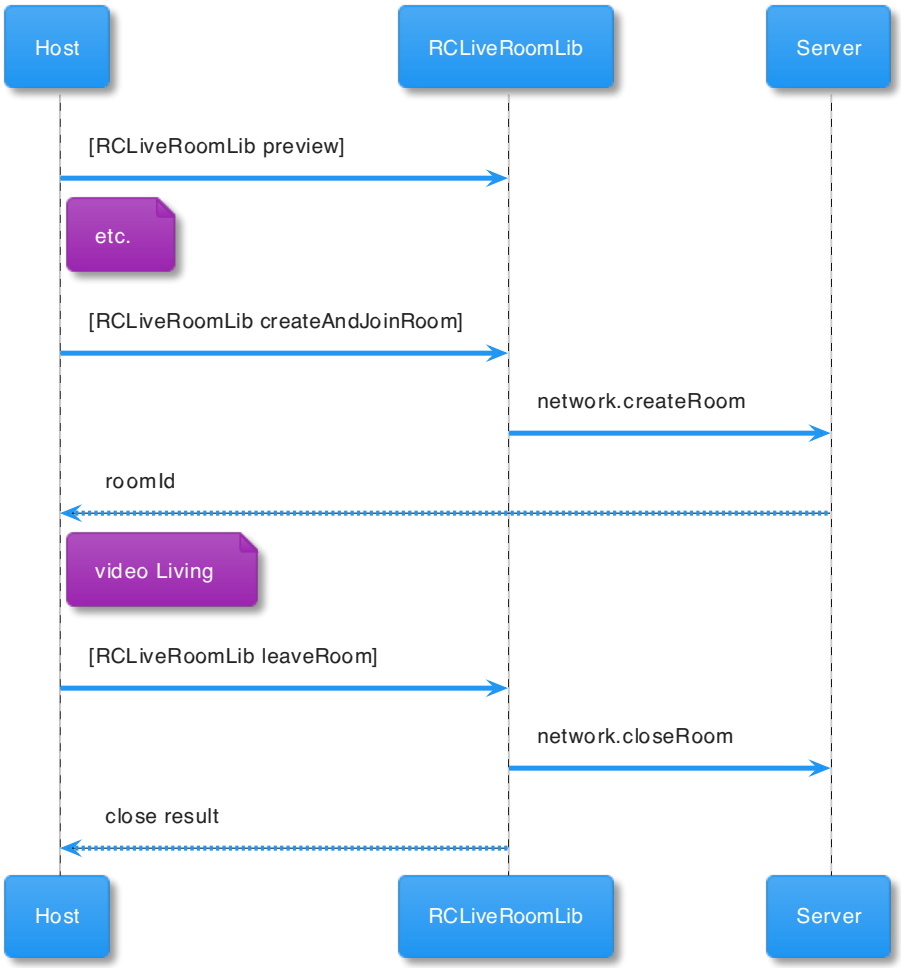
更新时间:2024-08-30

本节介绍如何使用 RCLiveRoomLib 开启和结束视频直播。

开启和结束视频直播相关接口

- 准备视频直播：调用 preview 接口，将要预览的视频元素传入 SDK。
- 开启视频直播：调用 createAndJoinRoom 接口开启直播。
- 结束视频直播：调用 leaveRoom 接口结束直播。

开启和结束视频直播时序图



开启视频直播

开启视频直播步骤如下：

1. 调用视频直播准备接口 preview，此时 SDK 会开始采集摄像头和麦克风资源，并在传入的 video 标签中播放预览。

示例代码

```
RCLiveRoomLib.preview({
videoNode: yourVideoNode
});
```

参数	类型	必填	说明
videoNode	HTMLVideoElement	是	需要预览的 video 标签页面元素

2. 调用业务服务器的接口创建一个房间，获得房间 `roomId`。

3. 获取房间 ID 之后，调用 createAndJoinRoom 方法。

示例代码

```
/// @param roomId 房间id
/// @param roomType 房间类型
/// @param roomName 房间名称
/// @param videoNode 页面视频video标签 播放主视频流
/// @param miniNode 页面视频video标签 播放连麦视频流
/// @param resolution 推流分辨率
/// @param fps 推流码率
/// @param publishURL 需要推第三方 CDN 时，上报第三方 CDN 推流地址
RCLiveRoomLib.createAndJoinRoom({
roomId: roomId,
roomName: roomName,
roomType: "live",
videoNode: videoNode,
miniNode: miniNode,
resolution: resolution,
fps: fps,
publishURL:publishURL
})
```

参数	类型	必填	说明
roomId	String	是	自定义直播间 ID
roomName	String	否	直播间名称为空时，默认以 `roomid` 为名称
roomType	String	是	房间类型参数必传 `live` 为视频直播类型
videoNode	HTMLVideoElement	是	默认视频直播窗口
miniNode	HTMLVideoElement	是	连麦视频小窗

参数	类型	必填	说明
resolution	resolution	否	设置推流分辨率。从 SDK 1.1.0 版本开始支持。
fps	fps	否	设置推流码率。从 SDK 1.1.0 版本开始支持。
publishURL	publishURL	否	需要推第三方 CDN 时，上报第三方 CDN 推流地址。从 SDK 1.1.0 版本开始支持。

4. 创建并加入房间之后，可以调用 setCodeRate 方法修改观众端的直播视频分辨率、码率。

> SDK 从 1.1.0 版本开始支持 setCodeRate 方法。

```
RCLiveRoomLib.setCodeRate({
coderate: coderate,
FPS: FPS,
})
```

参数	类型	必填	说明
coderate	RCResolution	是	设置观众端拉流分辨率
FPS	RCFrameRate	是	设置观众端拉流码率

添加、删除第三方 CDN 推流地址

当主播加入直播间之后，就可添加、删除第三方 CDN 推流地址。

> SDK 从 1.1.0 版本开始支持该功能。

1. 添加第三方 CDN 推流地址

示例代码

```
/// @param url 第三方 CDN 推流地址
RCLiveRoomLib.addPublishStreamUrls(url)
```

参数	类型	必填	说明
url	String	是	第三方 CDN 推流地址

2. 删除第三方 CDN 推流地址

示例代码

```
/// @param url 第三方 CDN 推流地址
RCLiveRoomLib.removePublishStreamUrls(url)
```

参数	类型	必填	说明
url	String	是	第三方 CDN 推流地址

结束视频直播

结束视频直播的步骤如下：

1. 调用业务服务器接口关闭房间；
2. 调用 leaveRoom 方法离开房间。

示例代码

```
RCLiveRoomLib.leaveRoom(roomId)
```

参数	类型	必填	说明
roomId	String	是	自定义直播间 ID

> 注意：SDK 不具备创建和关闭视频直播房间的功能，请调用业务服务器接口实现。

API 参考

- preview
- createAndJoinRoom
- leaveRoom

邀请视频直播连麦

更新时间:2024-08-30

本节介绍如何使用 RCLiveRoomLib 邀请视频直播连麦。

邀请视频直播连麦接口

- 房主：

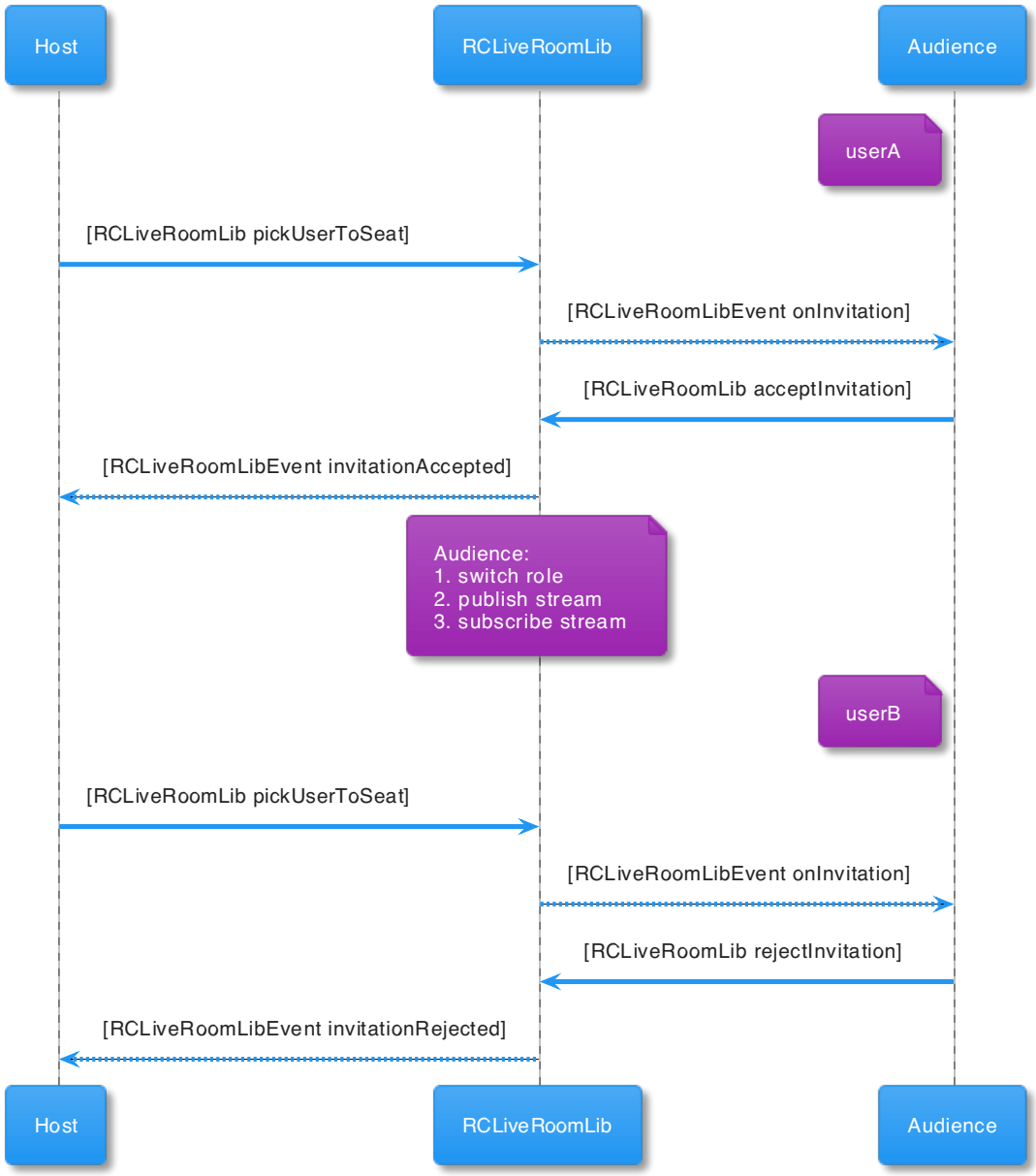
1. 发起上麦邀请：调用 `pickUserToSeat` 接口邀请观众上麦，观众接收到上麦邀请事件 `onInvitation` ；
2. 取消上麦邀请：调用 `cancelInvitation` 接口取消上麦邀请，观众接收到上麦请邀请取消事件 `onInvitationCancel` ；

- 观众：

1. 同意上麦邀请：调用 `acceptInvitation` 接口同意上麦邀请并自动连麦，房主接到同意回调事件 `invitationAccepted` ；
2. 拒绝上麦邀请：调用 `rejectInvitation` 接口拒绝上麦邀请，房主接收到拒绝回调事件 `invitationRejected` ；

> 注意：
>
> - 邀请上麦最多支持 **10** 人等待，当邀请人数达到10人后，再次调用邀请上麦接口会报错 `RCLiveVideoInvitationIsFull`；
> - 邀请上麦请求被同意后，如果上麦的麦位已被占用，SDK 会自动查询第一个空麦位；

邀请视频直播连麦时序图



邀请上麦

在视频直播 SDK 中，并没有权限的概念，也就是说，加入房间的任何人都可以发起邀请上麦，您需要根据自己业务的需求，确定哪些人可以发起邀请，比如：房主、管理员等。以房主为例，邀请上麦步骤如下：

1. 房主调用 pickUserToSeat 邀请用户上麦：

```
/// 邀请用户上麦，最多支持向10人发起邀请
/// @param userId 目标用户id
RCLiveRoomLib.pickUserToSeat(userId)
```

参数	类型	必填	说明
roomId	String	是	自定义直播间 ID

2. 观众触发事件 `onInvitation` ：

```
/// 接收到上麦邀请：只有受邀请者收到回调
/// @param id 目标用户id
/// @param seatIndex 受邀请上的麦位号
RCLiveRoomLib.on("onInvitation", (o) => {
//todo 收到邀请后的业务逻辑
}
```

参数	类型	必填	说明
id	String	是	被邀请用户的 ID
seatIndex	Number	是	被邀请上的麦位序号

3. 观众处理邀请调用 `acceptInvitation` 或 `rejectInvitation` ：

```
/// 接受上麦邀请，并上麦
/// @param userId 目标用户id
RCLiveRoomLib.acceptInvitation(userId)
/// 拒绝上麦邀请
/// @param userId 目标用户id
RCLiveRoomLib.rejectInvitation(userId)
```

参数	类型	必填	说明
userId	String	是	目标用户 ID

取消上麦邀请

房主发出邀请上麦后，在观众处理之前，可以调用接口 cancelInvitation 取消上麦邀请：

```
/// 取消上麦邀请
/// @param userId 目标用户id
RCLiveRoomLib.cancelInvitation(userId)
```

参数	类型	必填	说明
userId	String	是	目标用户 ID

API 参考

- pickUserToSeat
- on
- cancelInvitation

视频直播房间属性

更新时间:2024-08-30

本节介绍如何使用 RCLiveRoomLib 自定义视频直播房间属性。

视频直播房间属性接口

- 设置视频直播房间属性：调用 `updateKeyValue` 接口设置房间信息，房间内其他用户收到 `keyValueDidUpdate` 回调。

> 注意：房间自定义属性数量不能大于 50 个。房间属性最多 100 个。SDK 内部有固定的属性，如果设置属性总数超过 100，接口将返回错误 `RCLiveVideoUpdateRoomInfoFailed`。

设置房间属性

视频直播 SDK 提供了自定义属性接口 updateKeyValue，针对不同的直播场景（教育、游戏、社交等）开发者可以自定义属性。设置属性后，其他用户触发 keyValueDidUpdate 回调。步骤如下：

1. 设置房间属性：

```
/// 更新房间自定义属性，建议属性数量<50
/// @param roomInfo 房间信息属性<key, value>
RCLiveRoomLib.updateKeyValue({
key: "ROOM_INFO_PRE_notice",
value: "notice",
});
```

参数	类型	必填	说明
key	String	是	自定义房间属性名称
value	String	是	自定义房间属性值 复杂数据建议用 JSON 串转换

2. 其他用户触发回调：

```
/// 房间信息更新
/// @param key 房间信息属性
/// @param value 房间信息内容
RCLiveRoomLib.on("keyValueDidUpdate",(o)=>{
//todo 业务逻辑
//o.key 自定义键名
//o.value 自定义键值
});
```

参数	类型	必填	说明
key	String	是	自定义房间属性名称
value	String	是	自定义房间属性值 复杂数据建议用JSON串转换

API 参考

- updateKeyValue
- on

视频直播用户管理

更新时间:2024-08-30

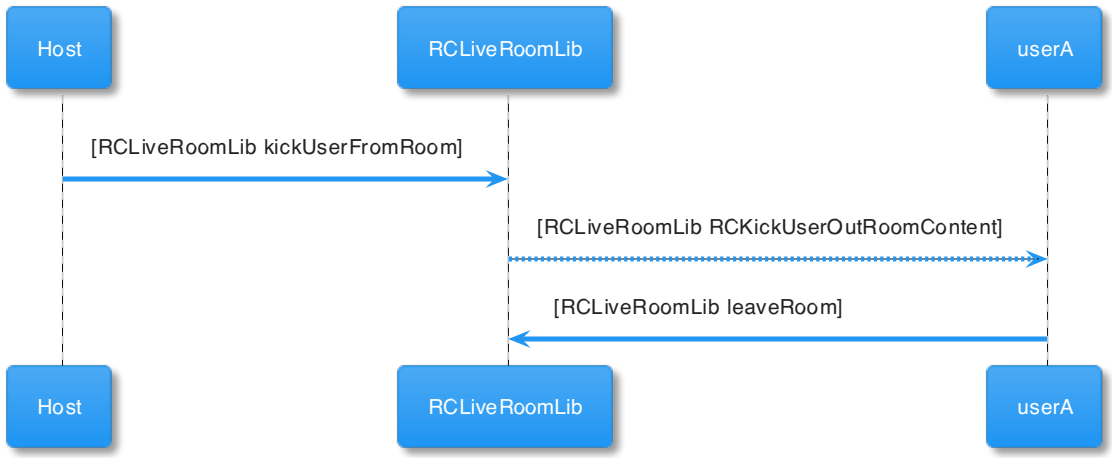
本节介绍如何使用 RCLiveRoomLib 管理用户。

视频直播用户管理接口

- 踢出房间：调用 `kickUserFromRoom` 接口，观众接收到 `RCKickUserOutRoomContent` 事件。

> 注意：在视频直播 SDK 中，并没有权限的概念，也就是说，房间的任何人都可以调用踢出房间的接口，您需要根据自己业务的需求，确定哪些人可以调用，比如：房主、管理员等。

踢出房间时序图



踢出房间

1. 调用 kickUserFromRoom 接口，将用户踢出房间。

```
/// 房主将用户踢出房间
RCLiveRoomLib.kickUserFromRoom(userId)
```

参数	类型	必填	说明
userId	String	是	被踢出的用户 ID

2. 踢出后，观众接收到 RCKickUserOutRoomContent 事件，目标用户在回调方法中处理被踢出的逻辑。

```
/// 被踢出房间的用户接收到回调，离开房间
RCLiveRoomLib.on("RCKickUserOutRoomContent",(o)=>{
//TODO 根据业务逻辑判断是否离开房间。 以下示例为直接离开房间
if (o.targetId == this.userId){
RCLiveRoomLib.leaveRoom(roomId)
}

})
```

参数	类型	必填	说明
targetId	String	是	被踢出房间的 userId

API 参考

- kickUserFromRoom

加入和离开视频直播

更新时间:2024-08-30

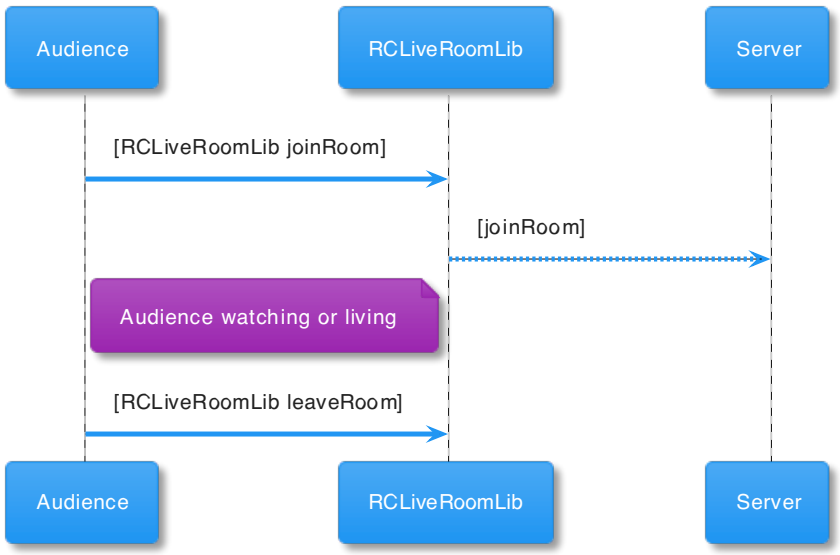
本节介绍如何使用 RCLiveRoomLib 加入和离开视频直播。

加入和离开视频直播接口

- 加入视频直播：调用 `joinRoom` 接口加入视频直播；
- 离开视频直播：调用 `leaveRoom` 接口离开视频直播；

> 注意：仅可加入已存在的视频直播房间。

加入和离开视频直播时序图



加入视频直播

观众加入已创建的视频直播房间，调用 joinRoom 接口：

```
/// 加入房间，只能加入已创建的房间
/// @param roomId 房间id
/// @param roomType 房间类型
/// @param videoNode 页面视频video标签 播放主视频流
/// @param miniNode 页面视频video标签 播放连麦视频流
/// @param CDNInfo.livePlayType 播放视频模式 取值:
/// @param CDNInfo.livePlayType = 0 为 MCU 低延迟直播
/// @param CDNInfo.livePlayType = 1 为融云 CDN 直播
/// @param CDNInfo.livePlayType = 2 为第三方 CDN 直播
/// @param CDNInfo.innerCDNFPS 可选（融云 CDN 直播时，可用于设置播放视频码率）
/// @param CDNInfo.innerCDNPreset 可选（融云 CDN 直播时，可用于设置播放视频分辨率）
RCLiveRoomLib.joinRoom({
roomId: roomId,
roomType: "live",
videoNode: videoNode,
miniNode: miniVideoNode,
CDNInfo:CDNInfo
})
```

参数	类型	必填	说明
roomId	String	是	自定义直播间 ID
roomType	String	是	房间类型参数，必传 `live` 为视频直播类型
videoNode	HTMLVideoElement	是	默认视频直播窗口
miniNode	HTMLVideoElement	是	连麦视频小窗
CDNInfo	CDNInfo	是	播放视频模式。SDK 从 1.1.0 版本开始支持。

监听播放模式处理融云 CDN 直播、第三方 CDN 直播

观众端监听 PlayThirdCDNStream 方法，使用第三方播放器播放融云 CDN 直播、第三方 CDN 直播，从 SDK 1.1.0 版本开始支持。

```
/// 观众端加入已创建的房间或主播修改为第三方 CDN 时触发
/// @param item.roomId 需要播放的房间id
/// @param item.isPlay 是否需要使用第三方播放器
/// @param item.type 播放模式 取值: 'MCU' 低延迟直播、'CDN' 融云 CDN 直播、'thirdCDN' 第三方 CDN 直播
/// @param item.pullUrl 拉流地址（当 item.type 为 'CDN' 时，参数中会附带 融云 CDN 直播的播放地址）
RCLiveRoomLib.on("PlayThirdCDNStream", async (item) => {
console.log(item)
})
```

离开视频直播

观众离开视频直播房间需调用 leaveRoom 接口：

```
/// 离开视频直播房间
RCLiveRoomLib.leaveRoom(roomId)
```

API 参考

- joinRoom
- leaveRoom

申请视频直播连麦

更新时间:2024-08-30

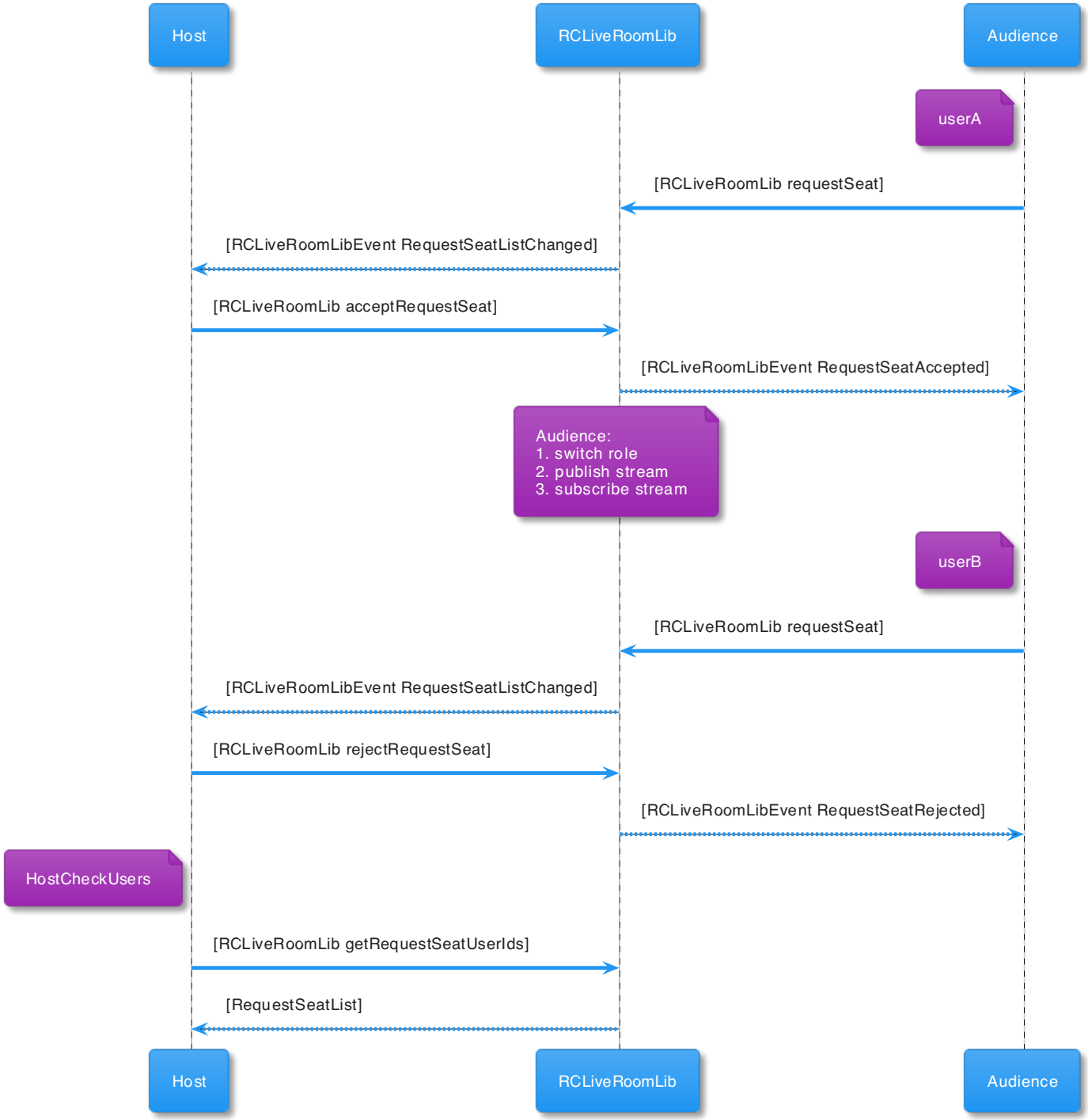
本节介绍如何使用 RCLiveRoomLib 申请视频直播连麦。

申请视频直播连麦接口

- 观众：
  1. 发起上麦申请：调用 `requestSeat` 接口申请上麦，房主接收到上麦请求变化 `RequestSeatListChanged` ；
  2. 取消上麦申请：调用 `cancelRequestSeat:` 接口取消上麦申请，房主接收到上麦请求变化 `RequestSeatListChanged` ；
  3. 查询上麦申请：调用 `getRequestSeatUserIds` 接口查询上麦邀请。
- 房主：
  1. 同意上麦申请：调用 `acceptRequestSeat` 接口同意用户上麦请求，申请者接到同意回调并自动连麦 `RequestSeatAccepted` ；
  2. 拒绝上麦申请：调用 `rejectRequestSeat` 接口拒绝用户上麦请求，申请者接收到拒绝回调 `RequestSeatRejected` ；
  3. 查询上麦申请：调用 `getRequestSeatUserIds` 接口查询上麦邀请。

> 注意：
>
> - 申请上麦最多支持 **10** 人等待，当申请人数达到10人后，再次调用申请上麦接口会报错 `RCLiveVideoReqeustIsFull`；
> - 申请上麦请求被同意后，如果上麦的麦位已被占用，SDK 会自动查询第一个空麦位；

申请视频直播连麦时序图



申请上麦

在视频直播 SDK 中，并没有权限的概念，也就是说，当房间某个用户申请上麦时，任何人都可以接收申请麦位变化的回调，您需要根据自己业务的需求，确定哪些人可以处理申请，比如：房主、管理员等。以房主为例，申请上麦步骤如下：

1. 观众调用 requestSeat 申请上麦：

```
/// 请求视频直播连麦，最多支持10人等待
RCLiveRoomLib.requestSeat();
```

2. 房主触发回调 RequestSeatListChanged ：

```
/// 上麦申请列表发生变化
RCLiveRoomLib.on("RequestSeatListChanged" , (RequestSeatList) => {
//todo 收到邀请后的业务逻辑
console.log(RequestSeatList);
})
```

参数	类型	必填	说明
RequestSeatList	Array	是	正在申请上麦的userId列表数组

3. 房主处理申请 `acceptRequestSeat` 或 `rejectRequestSeat` ：

```
/// 接受上麦请求
/// @param userId 目标用户id
RCLiveRoomLib.acceptRequestSeat(userId)
/// 调用此方法后目标用户自动上麦并开始推流

/// 拒绝上麦请求
/// @param userId 目标用户id
RCLiveRoomLib.rejectRequestSeat(userId)
```

参数	类型	必填	说明
userId	String	是	用户ID

取消上麦申请

观众发出申请上麦后，在房主处理之前，可以调用接口 `cancelRequestSeat` 取消上麦申请：

```
/// 取消上麦请求
/// @param 结果回调
RCLiveRoomLib.cancelRequestSeat()
```

API 参考

- requestSeat
- on

常见问题

更新时间:2024-08-30

创建视频直播房间返回失败

这通常是由于您没有开通 App Key 的音视频直播服务，或者是免费时长已用完时。您可以通过控制台查看是否已经开通音视频服务。

申请连麦时，谁有权限通过或拒绝申请？

在视频直播 SDK 中，并没有权限的概念，也就是说，当房间某个用户申请上麦时，任何人都可以接收申请麦位变化的回调。您需要根据自己业务的需求，确定哪些人可以处理申请。

聊天室发言功能接口

视频直播 SDK 1.0 版本 集成了 im 和 rtc 可以通过 RCLiveRoomLib.im.body / RCLiveRoomLib.rtc.body 访问到相关实例并调用相关方法。具体文档请参考 im rtc

更新日志 v1.1.0

更新时间:2024-08-30

发布日期：2022/09/23

修复问题：

1. 修复加入房间错误时无回调的问题
2. 修复观众端视频模糊问题

接口变更：

1. 方法 `init()` 只注册场景化直播房 SDK 能力，IMLib、RTCLib 功能需要单独安装并注册。
2. 删除 `connect()` 方法,连接需调用 IMLib 的 `connect()` 方法。
3. 新增 `addPublishStreamUrls()` 方法和 `removePublishStreamUrls()` 方法，添加、删除第三方 CDN 推流地址。
4. 新增 `setCodeRate()` 方法，创建并加入房间之后，修改观众端的直播视频分辨率、码率。
5. 新增主播开播添加三方 CDN 地址
6. 新增观众订阅三方 CDN 方法

状态码

更新时间:2024-08-30

状态码	说明
70000	操作成功
70001	加入聊天室失败
70002	加入RTC房间失败
70003	离开房间失败用
70005	发布流失败
70006	订阅流失败
70007	获取房间信息失败
70008	更新房间信息失败
70009	加入视频连麦失败
70010	离开rtc房间失败
70013	已经在排麦列表中了
70014	邀请连麦达到最大值
70015	排麦请求发送失败
70016	取消排麦请求发送失败
70017	同意排麦请求发送失败

状态码	说明
70018	拒绝排麦请求发送失败