

中国电信天翼对讲（CTChat）客户端（Android）标准版SDK接口规范V1.11

中国电信集团

2018年5月

目 录

[1综述 4](#_Toc18948)

[1.1 适用范围 4](#_Toc14578)

[1.2术语 4](#_Toc12628)

[1.3参考文档 4](#_Toc23128)

[1.4历史修改记录 5](#_Toc17344)

[2 描述 7](#_Toc18787)

[2.1 SDK总体结构图 7](#_Toc25948)

[3 接口说明 8](#_Toc29561)

[3.1 登录登出 8](#_Toc10486)

[3.1.1 数据类型 8](#_Toc5933)

[3.1.2 接口方法和回调 9](#_Toc21997)

[3.2 修改密码 15](#_Toc7473)

[3.2.1 数据类型 15](#_Toc7626)

[3.2.2 接口方法和回调 15](#_Toc13824)

[3.3 信息获取 17](#_Toc1999)

[3.3.1 数据类型 17](#_Toc18876)

[3.3.2 接口方法和回调 19](#_Toc2519)

[3.4 预定义组获取 27](#_Toc29713)

[3.4.1 数据类型 27](#_Toc16689)

[3.4.2 接口方法和回调 27](#_Toc21534)

[3.5 会话管理 31](#_Toc4959)

[3.5.1 数据类型 31](#_Toc30898)

[3.5.2 接口方法和事件回调 33](#_Toc32234)

[3.6 即时消息 44](#_Toc13148)

[3.6.1 数据类型 44](#_Toc21924)

[3.6.2 接口方法和回调 46](#_Toc12581)

[3.7 消息推送 55](#_Toc30981)

[3.7.1 数据类型 55](#_Toc28139)

[3.7.2 接口方法和事件回调 55](#_Toc22165)

[3.8 参数设置 58](#_Toc32690)

[3.8.1 数据类型 58](#_Toc28541)

[3.8.2 接口方法和事件回调 58](#_Toc24937)

[4 SDK接入说明 60](#_Toc32685)

[4.1 导入SDK 60](#_Toc24823)

[4.2 添加代码 60](#_Toc69)

[4.3 混淆设置 60](#_Toc12939)

# 1综述

## 适用范围

本文档重点说明了中国电信天翼对讲客户端（Android）开放SDK的相关接口，适用于使用此SDK开发中国电信天翼对讲客户端的开发人员和测试人员。

## 1.2术语

|  |  |
| --- | --- |
| **术语名称** | **说明** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

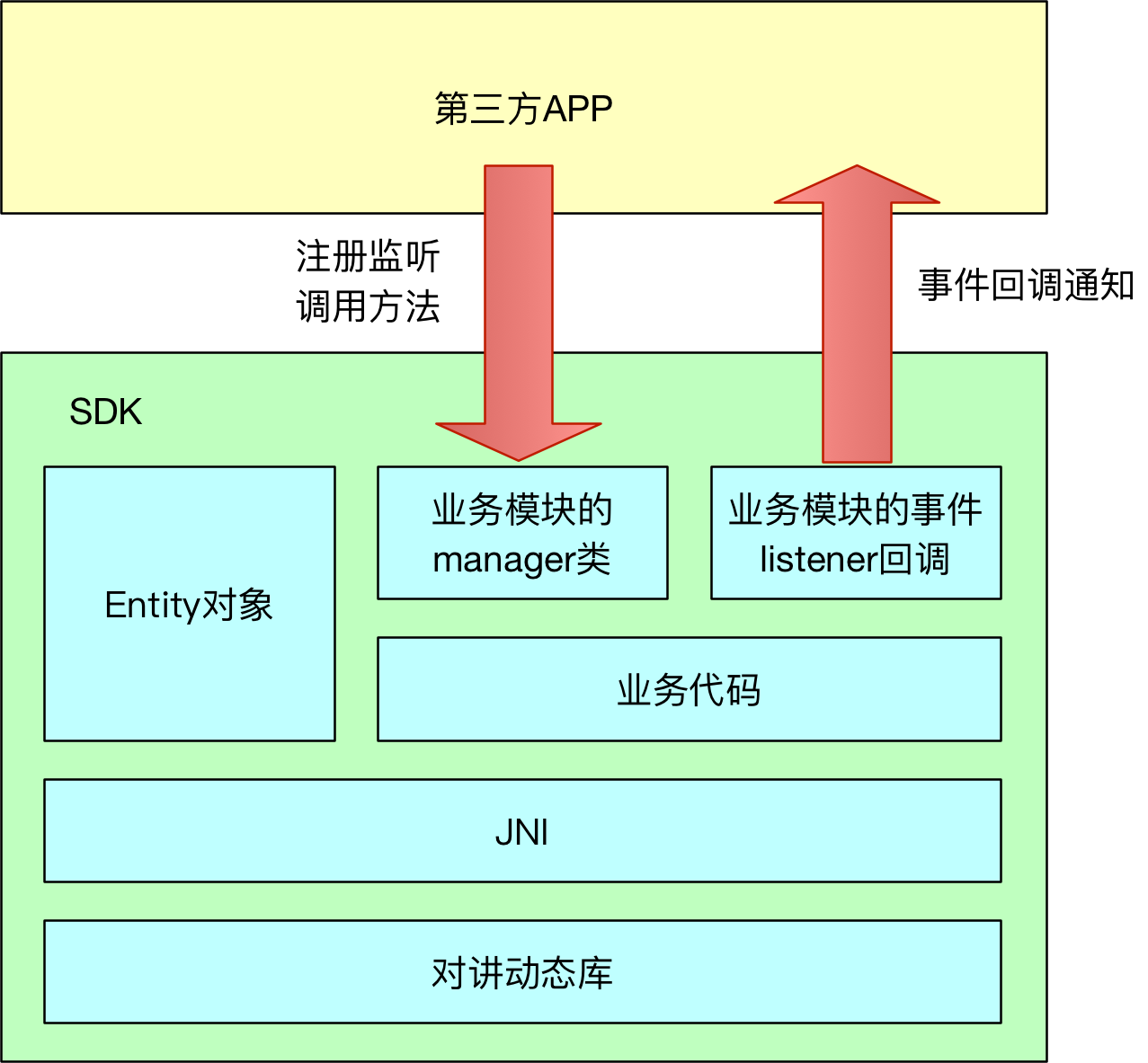
## 1.3参考文档

## 1.4历史修改记录

|  |  |  |
| --- | --- | --- |
| 版本 | 修订时间 | 修订内容 |
| 1.0 | 2018/06/15 | 初稿 |
| 1.1 | 2018/07/31 | 修改SDK配置接口 |
| 1.2 | 2018/8/6 | 添加获取广播呼叫实例接口、添加广播呼叫权限接口、  混淆配置修改 |
| 1.3 | 2018/8/17 | 增加修改密码接口；增加登录成功接口回调参数 |
| 1.4 | 2018/8/27 | 去除话权队列回调  完善部分接口描述； 添加即时消息播放、停止播放语音消息接口、清除会话未读消息数目接口； 增加设置imsi接口、清除登录数据接口； 修改、删减了登录的错误码； 删除广播呼叫相关接口； 增加账号心跳事件回调接口。  增加通过mdn查询用户在线状态的接口  增加获取本地缓存的预定义组列表接口 |
| 1.5 | 2018/9/3 | 去除预定义组话权抢占开关设置接口，默认开启话权抢占  去除申请话权失败的TALK\_END\_REASON \_SPEAKING\_FUL（允许讲话人数已达上限）错误码  去除音频放大器开关设置接口  修改设置imsi接口为必选接口，去除登录的错误码：无效的终端识别号LOGIN\_ERROR\_INVALID\_DEVICE\_ID |
| 1.6 | 2018/9/7 | 会话锁定相关接口描述修改：详见锁定接口说明。  添加广播消息相关功能接口。  添加预定义组群呼上线相关功能接口。 |
| 1.7 | 2018/9/14 | 完善获取会话在线人数接口描述  登录增加两个错误码：LOGIN\_ERROR\_REMOTE\_FORBID、LOGIN\_ERROR\_MUCH\_PASSWORD\_REQUEST  增加开关PTT录音接口  呼叫限制接口描述修改：详见呼叫限制接口说明  添加开始播放会话者声音的事件接口 |
| 1.8 | 2018/10/12 | 添加会话建立状态常量  修改会话状态描述  新增心跳频度上下限获取接口  设置心跳频度接口添加是否设置成功的返回值。 |
| 1.9 | 2018/10/22 | 添加位置消息相关经纬度及地址获取接口  修改加载会话消息接口描述 |
| 1.10 | 2018/11/2 | 添加预定义组限制与恢复的事件通知OnPushEventListener.onPushChannelLimit |
| 1.11 | 2018/11/23 | 即时消息中发送位置消息添加说明：要发送的位置坐标需要转换为地球坐标系的位置坐标。  即时消息发送图片、视频接口接口与说明修改：发送的图片（默认允许png、jpg）、视频文件类型（默认允许mp4、3gp）与文件大小要在允许（默认25MB）的范围内。  会话建立事件onSessionEstablished中会话建立的结果参数描述修改  添加日志输出方式选择接口  添加获取临时会话实例以及成员列表接口  添加媒体共享图片和视频接受失败重新下载的接口  会话成员变更事件onSessionMemberUpdate参数描述修改。 |

# 描述

## SDK总体结构图



本SDK旨在为第三方应用在Android平台快速实现对讲及流媒体业务提供的开发组件。第三方应用通过调用对应业务方法，SDK在处理完相关业务后会异步回调给相应事件的监听方，第三方app通过实现具体的监听接口来处理对应的事件以完成整个业务流程，详见结构说明中的各业务API说明。

# 接口说明

## 登录登出

### 数据类型

#### 数据常量

|  |  |  |
| --- | --- | --- |
| Class | com.ctchat.sdk.auth.api.ErrorCode | |
| 标签 | 常量 | 描述 |
| ERROR\_CODE | LOGIN\_ERROR\_INVALID\_APP\_KEY | 无效的appKey，无法登录 |
| LOGIN\_ERROR\_INVALID\_IMSI\_FORBID | 无效的imsi，禁止登录 |
| LOGIN\_ERROR\_INVALID\_IMSI\_PHONE\_SMS | 无效的imsi，允许手动输入手机mdn号+验证码登录 |
| LOGIN\_ERROR\_RECEIVE\_SMS\_CODE | 验证码已接收，可以用手机mdn号+验证码登录 |
| LOGIN\_ERROR\_INVALID\_IMSI\_PHONE\_PASSWORD | 无效的imsi，可以手动输入手机号+密码登录 |
| LOGIN\_ERROR\_INVALID\_ACCOUNT | 未授权的mdn号码，无法登录 |
| LOGIN\_ERROR\_NEED\_SMS\_VERIFICATION | 该mdn号码需要短信验证码登录 |
| LOGIN\_ERROR\_WRONG\_PASSWORD\_OR\_SMS | 密码或短信验证码错误 |
| LOGIN\_ERROR\_REQUEST\_TOO\_MUCH\_PIN\_CODE | 今日获取短信验证码次数已达上限（最多为6次） |
| LOGIN\_ERROR\_INVALID\_PIN\_CODE | 无效的短信验证码 |
| LOGIN\_ERROR\_INVALID\_API\_KEY | 无效的apiKey，需要清除登录数据之后登录 |
| LOGIN\_ERROR\_INVALID\_TOKEN | 无效的token，需要清除登录数据之后登录 |
| LOGIN\_ERROR\_SERVER\_UNAVAILABLE | 服务器异常 |
| LOGIN\_ERROR\_NETWORK | 网络异常 |
| LOGIN\_ERROR\_REMOTE\_FORBID | 该号码处于遥毙状态，禁止登录 |
| LOGIN\_ERROR\_MUCH\_PASSWORD\_REQUEST | 密码验证错误次数超过指定次数，需要等待系统指定时间后再尝试 |

### 接口方法和回调

#### SDK配置接口

class：com.ctchat.sdk.AccountManager

说明：客户端SDK连接服务器的域名或IP配置，以下接口均需要在登录之前进行调用。

##### 设置地址接口（必选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setServerAddr | | 配置登录服务器的域名或者IP |
| 参数 | String:rootUrl | 连接服务器根地址 |
| 返回值 | 无 | 无 |

##### 设置imsi接口（必选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setImsi | | 配置imsi。sdk内部会使用开发者配置的imsi进行登录 |
| 参数 | String:cdmaImsi | cdma(3G)模式下的imsi |
| String:lteImsi | lte(4G)模式下的imsi |
| 返回值 | 无 | 无 |

##### 设置AppKey接口（必选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setAppKey | | 配置厂商标识 |
| 参数 | String:appKey | 厂商标识 |
| 返回值 | 无 | 无 |

##### 设置录音来源接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setAudioRecordSource | | 开发者根据手机的rom设置合适的录音来源，默认值为MediaRecorder.AudioSource.DEFAULT，参考类MediaRecorder.AudioSource里面的常量 |
| 参数 | int:audioRecordSource | 录音来源 |
| 返回值 | 无 | 无 |

##### 开关日志接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setEnableLog | | 配置日志打开或关闭，默认关闭日志 |
| 参数 | boolean:enableLog | 日志开关 |
| 返回值 | 无 | 无 |

##### 设置日志输出方式接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setLogPrintMode | | 配置日志输出方式，该方法需要在setEnableLog方法之前调用，默认输出到文件 |
| 参数 | int:mode | mode值定义:AccountManager.TRACE\_MODE\_LOGCAT:表示控制台输出 AccountManager.TRACE\_MODE\_FILE:输出到文件 |
| 返回值 | 无 | 无 |
|  |  |  |

##### 设置音频播放类型接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setStreamType | | 配置音频播放类型 |
| 参数 | int:streamType | 音频播放类型，参照AudioManager中的常量，默认为STREAM\_MUSIC |
| 返回值 | 无 | 无 |

##### 设置声音频谱图（收听）接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setAudioPlayVisualize | | 配置声音频谱图（收听） |
| 参数 | boolean:audioPlayVisualize | 声音频谱图（收听）开关 |
| 返回值 | 无 | 无 |

##### 设置声音频谱图（录音）接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setAudioRecordVisualize | | 配置声音频谱图（录音） |
| 参数 | boolean:audioRecordVisualize | 声音频谱图（录音）开关 |
| 返回值 | 无 | 无 |

##### 设置声音频谱数量接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setAudioVisualizeNum | | 设置声音频谱数量 |
| 参数 | int:audioVisualizeNum | 声音频谱数量，默认为18 |
| 返回值 | 无 | 无 |

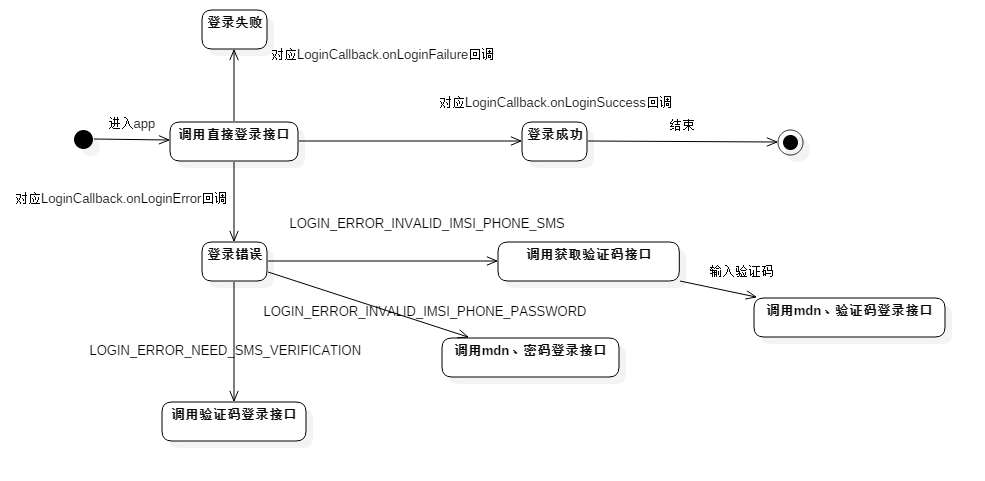
##### 开关PTT录音接口（可选）

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法：setEnablePttRecord | | 开关PTT录音 |
| 参数 | boolean:enablePttRecord | 开关PTT录音，默认为开启状态 |
| 返回值 | 无 | 无 |

#### 登录接口

class：com.ctchat.sdk.AccountManager

说明：各种登录方式接口调用，下面是登录的状态图



##### 直接登录接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:login | | 每次打开客户端只能使用直接登录方式进行首次登录 |
| 参数 | Context:context | 上下文 |
| 返回值 | 无 | 无 |
| 函数回调 | LoginCallback:callback | 登录结果接口回调 |

##### 手机mdn号+密码登录接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:login | | 如果直接登录失败，并且根据错误码确定需要手机mdn号+密码登录，才调用此方法。 |
| 参数 | Context:context | 上下文 |
| String:mdn | 手机号 |
| String:password | 密码 |
| 返回值 | 无 | 无 |
| 函数回调 | LoginCallback:callback | 登录结果接口回调 |

##### 手机mdn号+验证码登录接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:loginBySmsCode | | 如果直接登录失败，并且根据错误码确定需要mdn号+验证码登录，才调用此方法。 |
| 参数 | Context:context | 上下文 |
| String:mdn | 手机号 |
| String:smsCode | 验证码 |
| 返回值 | 无 | 无 |
| 函数回调 | LoginCallback:callback | 登录结果接口回调 |

##### 验证码登录接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法: loginBySmsCode | | 如果直接登录失败，并且根据错误码确定需要验证码登录，才调用此方法。 |
| 参数 | Context:context | 上下文 |
| String:smsCode | 验证码 |
| 返回值 | 无 | 无 |
| 函数回调 | LoginCallback:callback | 登录结果接口回调 |

##### 获取验证码接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:getSmsCode | | 要求mdn号+验证码登录的时候，需要先通过mdn号获取验证码 |
| 参数 | Context:context | 上下文 |
| String:mdn | mdn号码 |
| 返回值 | 无 | 无 |
| 函数回调 | LoginCallback:callback | 登录结果接口回调 |

#### 登出接口

class：com.ctchat.sdk.AccountManager

说明：账号登出接口工具类

##### 直接登出接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:logout | | 账号登出，需要重新登录 |
| 参数 | Context:context | 上下文 |
| 返回值 | 无 | 无 |
| 函数回调 | LogoutCallback:callback | 登出结果接口回调 |

#### 清除登录数据接口

class：com.ctchat.sdk.AccountManager

说明：账号相关操作的工具类

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:clearLoginData | | 清除登录数据，需要重新登录方可正常使用sdk接口 |
| 参数 | Context:context | 上下文 |
| 返回值 | 无 | 无 |
| 函数回调 | 无 | 无 |

#### 回调接口

##### 登录结果接口回调

interface：com.ctchat.sdk.LoginCallback

说明：登录事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：登录成功 | | 描述 |
| 虚方法：onLoginSuccess | | 登录成功 |
| 参数 | boolean:needChangePassword | 该用户是否需要修改密码。当用户采用平台侧预设的初始密码时，建议修改密码。 |
| 返回值 | 无 | 无 |

|  |  |  |
| --- | --- | --- |
| 事件：登录错误 | | 描述 |
| 虚方法：onLoginError | | 接收到服务器返回的登录错误消息，开发者需要根据错误码（errorCode）处理错误。 |
| 参数 | int:errorCode | 登录错误的错误码，参考ERROR\_CODE |
| 返回值 | 无 | 无 |

|  |  |  |
| --- | --- | --- |
| 事件：登录失败 | | 描述 |
| 虚方法：onLoginFailure | | 无法请求到服务器，登录失败。可能是网络不佳、服务器地址错误、服务器异常等原因 |
| 参数 | 无 | 无 |
| 返回值 | 无 | 无 |

##### 登出接口回调

interface：com.ctchat.sdk.listerner.LogoutCallback

说明：登出事件回调接口，在接收到登出事件后，在进行下次登录操作前，应用需要进行退出进程的操作。

|  |  |  |
| --- | --- | --- |
| 事件：登出 | | 描述 |
| 虚方法：onLogout | | 登出成功 |
| 参数 | 无 | 无 |
| 返回值 | 无 | 无 |

## 修改密码

### 数据类型

#### 数据常量

class:com.ctchat.sdk.password.PasswordError

说明：在修改密码时，可能产生的错误码

|  |  |  |
| --- | --- | --- |
| Class | com.ctchat.sdk.password.PasswordError | |
| 标签 | 常量 | 描述 |
| PASSWORD\_ERROR | ERROR\_SAME\_PASSWORD | 新密码与原密码相同 |
| ERROR\_WRONG\_ORIGINAL\_PASSWORD | 原密码错误 |
| ERROR\_INVALID\_NEW\_PASSWORD | 新密码不符合规范，密码规则为：8-12位，大写英文、小写英文、数字、特殊符号同时至少每项都包含一个 |

### 接口方法和回调

#### 密码管理

class：com.ctchat.sdk.password.PasswordManager

说明：用于修改密码的类。用户登录成功后，如果该用户的密码为平台预设的初始密码（根据onLoginSuccess中的needChangePassword参数判断），我们建议开发者强制要求用户修改密码。

##### 修改密码接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 静态方法:changePassword | | 调用此方法进行修改密码 |
| 参数 | Context:context | 上下文 |
| String:orgPassword | 原始密码 |
| String:newPassword | 新密码 |
| 返回值 | 无 | 无 |
| 函数回调 | ChangePasswordCallback:callBack | 修改密码接口回调 |

#### 回调接口

##### 修改密码接口回调

interface：com.ctchat.sdk.password.ChangePasswordCallback

说明：修改密码回调接口

|  |  |  |
| --- | --- | --- |
| 事件：修改密码成功 | | 描述 |
| 虚方法：onChangePasswordSuccess | | 修改密码成功 |
| 参数 | 无 | 无 |
| 返回值 | 无 | 无 |

|  |  |  |
| --- | --- | --- |
| 事件：修改密码错误 | | 描述 |
| 虚方法：onChangePasswordError | | 接收到服务器返回的修改密码错误消息。 |
| 参数 | int:errorCode | 修改密码错误的错误码，参考PASSWORD\_ERROR |
| 返回值 | 无 | 无 |

|  |  |  |
| --- | --- | --- |
| 事件：修改密码失败 | | 描述 |
| 虚方法：onChangePasswordFailure | | 无法请求到服务器，修改密码失败。可能是网络不佳、服务器地址错误、服务器异常等原因 |
| 参数 | 无 | 无 |
| 返回值 | 无 | 无 |

## 信息获取

### 数据类型

#### 当前登陆用户信息

class:com.ctchat.sdk.ptt.tool.entity.UserInfo

说明：用户信息类，当前登陆用户登陆鉴权成功后的用户信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getName | 获取用户名称 |  | String:用户名称 |
| getMdn | 获取当前登陆用户的mdn号码 |  | String:用户mdn号码 |
| getMaxHeartbeatFrequency | 获取允许设置的心跳频度上限 |  | Int：心跳频度上限（秒） |
| getMinHeartbeatFrequency | 获取允许设置的心跳频度下限 |  | Int：心跳频度下限（秒） |

#### 组织（企业）

class：com.ctchat.sdk.ptt.tool.entity.CompanyEntity

说明：企业联系人中的企业信息，呈树形结构。

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getCompanyId | 组织编号 |  | String:组织编号 |
| getCompanyName | 组织名称 |  | String:组织名称 |
| getChildCompanys | 该组织下的子组织列表 |  | List<CompanyEntity>:子组织列表 |
| getChildContacts | 该组织下的联系人列表 |  | List<ContactEntity>:组织下的联系人列表 |
| getParentCompanyId | 该组织的父组织节点编号 |  | String:父组织节点编号，根节点编号为0 |

#### 组织成员（企业联系人）

class：com.ctchat.sdk.ptt.tool.entity.ContactEntity

说明：某个企业下的企业联系人信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getDisplayName | 联系人名称 |  | String:联系人名称 |
| getState | 联系人状态 |  | int:参考常量CONTACT\_STATE |
| getMdn | 联系人mdn号码 |  | String:联系人mdn号码 |
| getCompanyId | 联系人所在组织的组织编号 |  | String:联系人所在组织的组织编号 |
| getCompanyName | 联系人所在组织的组织名 |  | String:联系人所在组织的组织名 |
| getCuserType | 联系人在会话中的级别 |  | int:参考常量CURSETYPE |
| getStateInChat | 联系人在会话中的状态 |  | int:参考常量IN\_CHAT\_STATE |

#### 广播消息

class：com.ctchat.sdk.ptt.tool.entity.BroadCastEntity

说明：系统广播类，接收到心跳广播消息推送后，从服务器获取的系统广播消息实体。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 | |
| getId | 消息对应的ID |  | String:消息ID | |
| getTitle | 消息标题 |  | String:消息的title标题 | |
| getContent | 消息内容 |  | String:消息内容 | |
| getCreater | 发布该消息的用户名称 |  | String:发布消息的用户名称 | |
| getReadState | 消息已读状态 |  | int:参考READ\_STATE | |
| getDate | 信息发布的时间 |  | String:信息发布的时间 | |
| getType | 消息类型 |  | int | 1:普通广播消息 |
| 2:sos报警 |

#### 数据常量

|  |  |  |
| --- | --- | --- |
| Class | com.ctchat.sdk.ptt.tool.entity.ContactEntity | |
| 标签 | 常量 | 描述 |
| CURSETYPE | CURSETYPE\_NONE | 未设置 |
| CURSETYPE\_MANAGER | 管理员，高级别用户 |
| CURSETYPE\_USER | 普通用户 |
| CURSETYPE\_LISTEN\_ONLY | 只能收听，无讲话权限 |
| CURSETYPE\_LOW\_PRIORITY | 低级别用户 |
| CONTACT\_STATE | CONTACT\_STATE\_NONE | 联系人不在线 |
| CONATCT\_STATE\_ONLINE | 联系人在线 |
| IN\_CHAT\_STATE | IN\_CHAT\_STATE\_ONLINE | 用户在此会话中 |
| IN\_CHAT\_STATE\_OFFLINE | 用户不在此会话中 |

### 接口方法和回调

#### 获取用户信息

class：com.ctchat.sdk.ptt.tool.userinfo.UserInfoManager

说明：用户信息相关类，用于获取当前登陆用户的相关信息，网络变化以及电话冲突等情况的处理等，如：登陆用户信息mdn号和名称、登陆状态等信息，网络连接状态变更以及来去电冲突等情况的处理方法。

##### 单例对象和设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 静态方法：getInstance | | 访问该类中的方法需要通过该方法获取到实例进行访问 |
| 参数 | 无 |  |
| 返回值 | UserInfoManager | UserInfoManager实例 |

##### 当前登陆用户信息获取接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getUserInfo | | 获取当前登陆用户的信息 |
| 参数 | Context:context | 上下文 |
| 返回值 | UserInfo:userInfo | 当前登陆用户的信息实例 |

##### 查询账号登陆状态接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getAccountLoginStatus | | 查询账号是否处于登陆状态 |
| 参数 | 无 |  |
| 返回值 | boolean | true：已登录，false：未登录 |
| 函数回调 | 无 |  |

##### 网络状态切换处理接口

说明：第三方通过Android事件监听处理机制，实现一个BroadcastReceiver，捕获android.net.conn.CONNECTIVITY\_CHANGE事件，然后在网络连接时调用onNetworkOpen接口，在网络连接断开时调用onNetworkClose接口。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：onNetworkOpen | | 第三方应用通过监听网络状态，当监听到网络连接打开事件时需要调用此接口。 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：onNetworkClose | | 第三方应用通过监听网络状态，当监听到网络连接断开事件时需要调用此接口。 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 电话冲突处理接口

说明：第三方通过Android事件监听处理机制，实现一个BroadcastReceiver，捕获android.intent.action.PHONE\_STATE和 android.intent.action.NEW\_OUTGOING\_CALL 事件，在事件发生时调用以下接口处理。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：onCallStart | | 当有电话事件发生时，无论是来电还是去电，需要调用此接口。 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：onCallStop | | 当电话结束后，需要调用此接口。 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

#### 获取组织结构信息

class：com.ctchat.sdk.ptt.tool.contact.ContactManager

说明：企业与企业联系人相关，用于从服务器端加载企业联系人信息，订阅联系人在线状态等。

第三方应用从服务器端同步企业联系人数据后，可通过mdn号匹配联系人实列ContactEntity、通过联系人实例可以查询联系人在线状态、通过订阅接口来监听所有组织联系人的在线状态变更、通过设置组织结构监听来获取组织联系人网络同步的结果和联系人属性变更等。

##### 单例对象和设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 枚举值：INSTANCE | | 访问该类中的其他方法需要先通过该枚举获取该对象的实例 |
| 参数 | 无 |  |
| 返回值 | ContactManager | ContactManager实例 |

|  |  |  |
| --- | --- | --- |
| 注册组织结构监听 | | 描述 |
| 方法：registerLoadContactDataListener | | 注册从网络获取组织结构的监听 |
| 参数 | ContactDataListener:listener | ContactDataListener 实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 注销组织结构监听 | | 描述 |
| 方法：unregisterLoadContactDataListener | | 注销组织结构监听 |
| 参数 | 无 |  |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 注册组织成员状态监听 | | 描述 |
| 方法：registerContactPresenceListener | | 注册组织成员在线状态监听 |
| 参数 | ContactPresenceListener:listener | ContactPresenceListener实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 注销组织成员状态监听 | | 描述 |
| 方法：unregisterContactPresenceListener | | 注销组织成员在下状态监听 |
| 参数 | 无 |  |
| 返回值 | 无 |  |

##### 获取组织结构接口

|  |  |  |
| --- | --- | --- |
| 从服务端获取组织结构 | | 描述 |
| 方法：getOrgnizationInfo | | 从服务端获取组织结构 |
| 参数 | Context:context | 上下文 |
| String:lastUpdateTime | 上次更新时间戳 |
| 返回值 | 无 |  |
| 回调函数 | ContactDataListener | 接收获取组织结构获取的结果 |

|  |  |  |
| --- | --- | --- |
| 通过mdn号获取组织联系人 | | 描述 |
| 方法：getContactByMdn | | 通过mdn号获取本地缓存的联系人 |
| 参数 | String:mdn | 联系人mdn号 |
| 返回值 | ContactEntity:contact | 联系人实例 |

|  |  |  |
| --- | --- | --- |
| 获取所有本地缓存的组织联系人 | | 描述 |
| 方法：getLocalContactList | | 获取所有本地缓存的联系人 |
| 参数 | 无 |  |
| 返回值 | List<ContactEntity>:contacts | 企业联系人列表 |

|  |  |  |
| --- | --- | --- |
| 获取所有本地缓存的组织 | | 描述 |
| 方法：getLocalCompanyList | | 获取所有本地缓存的企业 |
| 参数 | 无 |  |
| 返回值 | List<CompanyEntity>:companys | 企业列表 |

|  |  |  |
| --- | --- | --- |
| 获取联系人在线状态 | | 描述 |
| 方法：getContactStateByEntity | | 通过联系人实例获取在线状态 |
| 参数 | ContactEntity:contactEntity | 联系人实例 |
| 返回值 | int:state | 联系人状态,参考常量CONTACT\_STATE |

|  |  |  |
| --- | --- | --- |
| 获取联系人在线状态 | | 描述 |
| 方法：getContactStateByMdn | | 通过联系人mdn号获取在线状态 |
| 参数 | String:mdn | 联系人mdn号 |
| 返回值 | int:state | 联系人状态,参考常量CONTACT\_STATE |

##### 在线状态订阅、解订阅接口

|  |  |  |
| --- | --- | --- |
| 订阅所有联系人的在线状态 | | 描述 |
| 方法：subscribeContactPresence | | 开启所有联系人的在线状态订阅，订阅的联系人状态变更可通过注册组织成员状态监听来获取。 |
| 参数 | Context:context | 上下文 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 取消在线状态订阅 | | 描述 |
| 方法：unSubscribeContactPresence | | 关闭在线状态订阅 |
| 参数 | 无 |  |
| 返回值 | 无 |  |

#### 获取广播消息

class：com.ctchat.sdk.ptt.tool.broadcast.BroadCastManager

说明：广播通知消息相关，用于加载服务器端的系统广播消息和获取本地缓存的广播消息，删改查本地缓存广播消息等

第三方应用可以通过设置接口来监听广播消息网络获取结果以及新消息推送、通过调用对应的删改查接口操作本地消息缓存、通过刷新接口从网络加载新消息列表等。

##### 单例对象和设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 方法：getInstance | | 访问该类中的方法需要通过该方法获取到实例进行访问 |
| 参数 | 无 |  |
| 返回值 | BroadCastManager | BroadCastManager实例 |

|  |  |  |
| --- | --- | --- |
| 注册广播消息获取监听 | | 描述 |
| 方法：registerBroadcastGetListener | | 注册广播消息获取监听 |
| 参数 | OnBroadcastGetListener:listener | OnBroadcastGetListener实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 注销广播消息监听 | | 描述 |
| 方法：unregisterBroadcastGetListener | | 注销广播消息监听 |
| 参数 | 无 |  |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 注册新广播消息推送监听 | | 描述 |
| 方法：registerNewBroadcastListener | | 注册广播消息获取监听 |
| 参数 | OnNewBroadCastSubscribe:listener | OnNewBroadCastSubscribe实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 注销新广播消息推送监听 | | 描述 |
| 方法：unregisterNewBroadcastListener | | 注销广播消息监听 |
| 参数 | OnNewBroadCastSubscribe:listener | OnNewBroadCastSubscribe实例 |
| 返回值 | 无 |  |

##### 广播消息接口

|  |  |  |
| --- | --- | --- |
| 刷新获取广播消息列表 | | 描述 |
| 方法 refreshBroadcastDatas | | 获取广播消息列表 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | OnBroadcastGetListener | 接收获取广播消息结果 |

|  |  |  |
| --- | --- | --- |
| 判断是否有新消息 | | 描述 |
| 方法：hasUnreadNoticeNum | | 判断是否有新的未读或未刷新的广播消息 |
| 参数 | 无 |  |
| 返回值 | boolean | true:有未读或未更新的新广播消息，false:无未读或未更新的广播消息 |

|  |  |  |
| --- | --- | --- |
| 删除某条广播消息 | | 描述 |
| 方法：broadcastRemove | | 指定删除某条广播消息 |
| 参数 | String:msgId | 广播消息ID |
| 返回值 | 无 |  |
| 函数回调 | OnBroadcastGetListener.onBroadCastUpdateComplete | 接收广播消息状态变更、广播消息列表变更（删除、清空操作）的结果 |

|  |  |  |
| --- | --- | --- |
| 更新广播消息状态 | | 描述 |
| 方法：updateBroadcastReadState | | 更新广播消息状态 |
| 参数 | BroadcastEntity:broadcastEntity | 广播消息实例 |
| 返回值 | 无 |  |
| 函数回调 | OnBroadcastGetListener.onBroadCastUpdateComplete | 接收广播消息状态变更、广播消息列表变更（删除、清空操作）的结果 |

|  |  |  |
| --- | --- | --- |
| 清空本地广播消息缓存 | | 描述 |
| 方法：broadcastClean | | 清空本地广播消息缓存 |
| 参数 | 无 | 无 |
| 返回值 | 无 |  |
| 回调函数 | OnBroadcastGetListener.onBroadCastUpdateComplete | 接收广播消息状态变更、广播消息列表变更（删除、清空操作）的结果 |

|  |  |  |
| --- | --- | --- |
| 获取本地缓存的广播消息列表 | | 描述 |
| 方法：getBroadcastList | | 获取本地缓存的广播消息列表 |
| 参数 | 无 | 无 |
| 返回值 | List<BroadcastEntity>:broacasList | 本地缓存的广播消息列表 |

#### 回调接口

##### 组织结构接口回调

interface：com.ctchat.sdk.ptt.tool.contact.ContatctDataListener

说明：组织结构事件回调接口，用于接收从服务器端获取到的企业联系人信息。

|  |  |  |
| --- | --- | --- |
| 事件：企业联系人获取结果 | | 描述 |
| 虚方法：onLoadContactSuccess | | 组织结构获取返回结果 |
| 参数 | List<ContactEntity>:contactList | 所有组织联系人列表 |
|  | List<CompanyEntity>:companyList | 所有组织列表 |
| String:updateTime | 发生更新时间戳 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：企业联系人加载失败 | | 描述 |
| 虚方法：onLoadContactListError | | 企业联系人加载失败 |
| 参数 | 无 | 无 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：企业联系人属性变更 | | 描述 |
| 虚方法：onContactUpdate | | 企业联系人属性发生变更 |
| 参数 | ContactEntity:contactEntity | 组织联系人 |
|  | String:updateTime | 发生变更的时间戳 |
| 返回值 | 无 |  |

interface：com.ctchat.sdk.ptt.tool.contact.ContactPresenceListener

说明：组织成员在线状态事件回调，用于接收联系人在线状态变更事件。

|  |  |  |
| --- | --- | --- |
| 事件：联系人状态 | | 描述 |
| 虚方法：onContactPresence | | 组织成员列表状态变化，当用户登录后或订阅后，系统会一次性返回所有用户群的在线状态。为呈现每个联系人的具体状态，可通过getContactStateByMdn方法获取每一个用户的状态值 |
| 参数 | boolean:isSubscribed | 订阅状态 |
| HashMap<String,Integer>:hashMap | 成员状态列表，String:联系人mdn号，Integer:联系人状态（参考CONTACT\_STATE） |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：联系人状态 | | 描述 |
| 虚方法：onContactPresence | | 组织成员状态变化 |
| 参数 | boolean:isSubscribed | 订阅状态 |
| String:mdn | 联系人mdn号 |
| int :state | 联系人状态，参考CONTACT\_STATE |
| 返回值 | 无 |  |

##### 广播信息接口回调

interface：com.ctchat.sdk.ptt.tool.broadcast.onBroadCastGetListener

说明：广播消息接口回调接口，用于接收从服务器端同步的系统广播消息。

|  |  |  |
| --- | --- | --- |
| 事件：获取广播消息列表成功 | | 描述 |
| 虚方法：onBroadCastGetSuccess | | 成功获取广播消息 |
| 参数 | List<BroadCastEntity>:list | 广播消息列表 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：获取广播消息列表失败 | | 描述 |
| 虚方法：onBroadCastGetError | | 获取广播消息失败 |
| 参数 | String | 失败原因 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：广播消息状态变更、消息列表变更 | | 描述 |
| 虚方法：onBroadCastUpdateComplete | | 接收广播消息状态变更、消息列表变更事件，如：修改广播消息已读状态、删除或清空广播消息列表等都会触发该事件 |
| 参数 | List<BroadcastEntity>:list | 更新后的广播消息列表 |
| 返回值 | 无 |  |

Interface：com.ctchat.sdk.ptt.tool.broadcast.OnNewBroadCastSubscribe

说明：新广播消息推送接收回调，接收到心跳新广播消息推送。

|  |  |  |
| --- | --- | --- |
| 事件：新广播消息推送 | | 描述 |
| 虚方法：onNewBroadcast | | 接收到心跳的新系统广播消息推送 |
| 参数 | String:content | 新系统广播消息推送内容 |
| 返回值 | 无 |  |

## 预定义组获取

### 数据类型

#### 预定义组

class：com.ctchat.sdk.ptt.entity.ChannelEntity

说明：预定义组信息类

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getName | 预定义组名称 |  | String:预定义组名称 |
| getId | 预定义组ID |  | String:预定义组ID |
| getMemberAll | 预定义组成员列表 |  | List<ContactEntity> |
| getSessionEntity | 预定义组会话实例 |  | SessionEntity：预定义组会话实例 |

#### 数据常量

|  |  |  |
| --- | --- | --- |
| Class | com.ctchat.sdk.ptt.tool.channel. ChannelManager | |
| 标签 | 常量 | 描述 |
| RET\_ALERT | RET\_ALERT\_SUCCESS | 群呼上线成功 |
| RET\_ALERT\_FAILED | 群呼上线失败 |
| RET\_ALERT\_NET\_ERROR | 网络异常 |

### 接口方法和回调

#### 预定义组信息接口

class：com.ctchat.sdk.ptt.tool.channel.ChannelManager

说明：负责预定义组信息的操作方法

##### 单例对象和事件监听设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 静态方法：INSTANCE | | 访问该类中的其它方法，必须通过此静态方法获取实例进行访问 |
| 参数 | 无 |  |
| 返回值 | ChannelManager | ChannelManager实例 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerLoadChannelDataListener (OnChannelDataListener listener) | 关心 OnChannelDataListener事件的实例，均需要通过此接口注册，否则将收不到相应事件 |

|  |  |
| --- | --- |
| 反注册监听回调接口 | 描述 |
| unregisterLoadChannelDataListener() | 不再关心OnChannelDataListener事件的实例，需要通过此接口反注册 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerChannelAlertListener (OnChannelAlertListener listener) | 关心 OnChannelAlertListener事件的实例，均需要通过此接口注册，否则将收不到相应事件 |

|  |  |
| --- | --- |
| 反注册监听回调接口 | 描述 |
| unregisterChannelAlertListener () | 不再关心OnChannelAlertListener事件的实例，需要通过此接口反注册 |

##### 获取预定组列表接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：getChannelList | | 从服务器获取当前用户所属的预定义组列表 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | OnChannelDataListener.onGetChannelList | 接收预定义组列表数据 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：getLoadedChannels | | 从本地缓存获取当前用户所属的预定义组列表 |
| 参数 | 无 |  |
| 返回值 | List<ChannelEntity> | 用户所属的预定义组列表 |
| 函数回调 | 无 |  |

##### 获取预定义组成员列表接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：getChannelMembers | | 从服务器获取指定预定义组成员列表 |
| 参数 | String:channelId | 预定义组的ID |
| 返回值 | 无 |  |
| 函数回调 | OnChannelDataListener. onGetChannelMemberList | 接收预定义组成员列表数据 |

##### 预定组群呼上线接口

说明：会对预定义组会话中的成员发起群呼上线通知。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：channelAlertOnline | | 对指定预定义组发起群呼上线通知 |
| 参数 | String:channelId | 预定义组的ID |
| 返回值 | 无 |  |
| 函数回调 | OnChannelAlertListener. onChannelAlertRet | 返回群呼上线结果 |

#### 回调接口

##### 预定义组信息接口回调

interface：com.ctchat.sdk.ptt.tool.channel.OnChannelDataListener

说明：预定义组信息事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：获取预定义组列表事件 | | 描述 |
| 虚方法：onGetChannelList | | 预定义组列表获取结果 |
| 参数 | List<ChannelEntity>:list | 预定义组列表 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：获取预定义组成员事件 | | 描述 |
| 虚方法：onGetChannelMemberList | | 预定义组成员获取结果 |
| 参数 | String:channelId | 预定义组ID |
| List<ContactEntity> list | 获取到该预定义组的成员列表 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：预定义组新增事件 | | 描述 |
| 虚方法：onNotifyChannelCreate | | 预定义组新增通知 |
| 参数 | ChannelEntity:channel | 后台新增的预定义组 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：预定义组删除事件 | | 描述 |
| 虚方法：onNotifyChannelDelete | | 预定义组删除通知 |
| 参数 | ChannelEntity:channel | 后台删除的预定义组 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：预定义组成员新增事件 | | 描述 |
| 虚方法：onNotifyChannelMemberAdd | | 预定义组成员新增通知 |
| 参数 | ChannelEntity:channel | 成员新增的预定义组 |
| List<ContactEntity>:contacts | 新增的成员表 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：预定义组成员移除事件 | | 描述 |
| 虚方法：onNotifyChannelMemberDelete | | 预定义组成员删除通知 |
| 参数 | ChannelEntity:channel | 成员删除的预定义组 |
| List<ContactEntity>:contacts | 移除的成员表 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：预定义组成员更新事件 | | 描述 |
| 虚方法：onNotifyChannelMemberUpdate | | 预定义组成员更新通知 |
| 参数 | ChannelEntity:channel | 成员更新的预定义组 |
| List<ContactEntity>:contacts | 更新的成员表 |
| 返回值 | 无 |  |

##### 预定义组群呼上线接口回调

interface：com.ctchat.sdk.ptt.tool.channel.OnChannelAlertListener

说明：预定义组群呼上线事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：群呼上线事件 | | 描述 |
| 虚方法：onChannelAlertRet | | 预定义组群呼上线事件回调 |
| 参数 | int:ret | 群呼上线结果，参照数据常量：RET\_ALERT |
| 返回值 | 无 |  |

## 会话管理

### 数据类型

#### 会话成员在线状态

class:com.ctchat.sdk.tool.entity.SessionEntity$SessionStateCount

说明：会话在线人数标识类

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getAllmemberCount | 获取全部成员数 |  | int:全员人数 |
| getChatOfflineCount | 获取后台在线人数，后台在线人数指成员未参与会话，但是平台在线的人数 |  | int:后台在线人数 |
| getChatOnlineCount | 获取前台在线人数，前台在线人数指成员参与会话的人数 |  | int:前台在线人数 |

#### 会话信息

class:com.ctchat.sdk.tool.entity.SessionEntity

说明：会话信息存储类，作为SDK的核心类，常用来作为调用呼叫或者即时消息的接口的参数

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getSessionId | 获取会话ID |  | String:会话ID |
| getSessionStatus | 获取会话状态 |  | int:会话状态，参考常量标签SESSION\_STATE |
| getMediaState | 获取会话话权状态 |  | int:会话话权状态，参考常量标签MEDIA\_STATE |
| getMemberAll | 获取会话所有成员 |  | List<ContactEntity>:成员列表 |
| getMemberSize | 获取会话所有人员个数 |  | int:所有成员总人数 |
| getSessionType | 获取会话类型 |  | int:参考常量标签SESSION\_TYPE |
| getSessionName | 获取临时会话名字，如果人数超过三人用省略号表示 |  | String:临时会话名字 |
| getChannel | 获取预定义组会话中群组实例 |  | ChannelEntity:预定义组信息实例 |
| getChannelName | 获取预定义组呼名字 |  | String:预定义组呼名 |
| getSpeaker | 获取会话说话人 |  | ContactEntity:说话人信息实例 |
| getCaller | 获取会话发起人 |  | ContactEntity:会话发起人信息实例 |
| isLocked | 判断会话是否锁定 |  | boolean:会话锁定状态 true:表示会话锁定 false:表示会话未锁定 |
| getMessages | 获取会话即时消息 |  | List<MessageEntity>:会话相关的即时消息列表 |
| getLastMessage | 获取会话最后一条即时消息 |  | MessageEntity:即时消息实例 |
| getMessageUnreadCount | 获取会话中即时消息未读个数 |  | int:即时消息未读个数 |
| setMessageUnreadCount | 设置会话中即时消息未读个数 | int:未读个数 |  |
| getMediaButtonState | 获取话权显示按钮所处的状态 |  | int:话权显示按钮状态参见常量标签  MEDIA\_BUTTON\_STATE |
| getMessagePlayback | 获取最新一条会话录音即时消息 |  | MessageEntity:即时消息实例 |

#### 数据常量

|  |  |  |
| --- | --- | --- |
| Class | com.ctchat.sdk.tool.entity.SessionEntity | |
| 标签 | 常量 | 描述 |
| SESSION\_STATE | SESSION\_STATE\_IDLE | 会话空闲 |
| SESSION\_STATE\_CALLING | 会话建立中 |
| SESSION\_STATE\_DIALOG | 会话进行中 |
| SESSION\_TYPE | TYPE\_DIALOG | 临时会话 |
| TYPE\_GROUP | 预定义组会话 |
| MEDIA\_STATE | MEDIA\_STATE\_IDLE | 会话话权空闲状态 |
| MEDIA\_STATE\_LISTEN | 会话话权被抢占状态 |
| MEDIA\_STATE\_TALK | 会话话权抢占状态 |
| SESSION\_RELEASE\_REASON | SESSION\_RELEASE\_REASON\_GENEAL | 会话正常结束 |
| SESSION\_RELEASE\_REASON\_ERROR | 会话因服务器异常错误结束 |
| SESSION\_RELEASE\_REASON\_NOTREACH | 会话因被叫离线结束 |
| SESSION\_RELEASE\_REASON\_BUSY | 会话因被叫忙结束 |
| SESSION\_RELEASE\_REASON\_FORBIDDEN | 会话因被禁止结束 |
| SESSION\_RELEASE\_REASON\_REJCTED | 会话因被叫拒绝结束 |
| SESSION\_RELEASE\_REASON\_NOANSWER | 会话因被叫无应答结束 |
| SESSION\_RELEASE\_REASON\_EXPIRE | 用户心跳周期已过期，需要重新登录 |
| SESSION\_RELEASE\_REASON\_NETWORK\_TERMINATE | 网络错误 |
| SESSION\_RELEASE\_REASON\_FREQUENTLY | 会话因呼叫频率过高结束 |
| SESSION\_ANSWER\_MODE | SESSION\_ANSWER\_MODE\_AUTO | 会话自动应答 |
| SESSION\_ANSWER\_MODE\_MANUAL | 会话手动应答 |
| TALK\_END\_REASON | TALK\_END\_REASON\_RELEASED | 话权正常释放 |
| TALK\_END\_REASON\_GRABED | 话权被更高级用户抢占 |
| TALK\_END\_REASON\_TIMEOUT | 申请话权超时，未申请到话权 |
| TALK\_END\_REASON\_TIMEUP | 用户发言时间过长 |
| TALK\_END\_REASON\_EXCEPTION | 由于网络异常导致话权中断 |
| TALK\_END\_REASON\_LISTEN\_ONLY | 用户只有听权限不能发言 |
| TALK\_END\_REASON\_NOBODY\_IN\_CHAT | 当前无人在线 |
| MEDIA\_BUTTON\_STATE | MEDIA\_BUTTON\_STATE\_IDLE | 话权按钮处于空闲状态 |
| MEDIA\_BUTTON\_STATE\_CONNECTING | 话权按钮处于连接状态 |
| MEDIA\_BUTTON\_STATE\_REQUESTING | 话权按钮处于申请状态 |
| MEDIA\_BUTTON\_STATE\_RELEASING | 话权按钮处于释放状态 |
| MEDIA\_BUTTON\_STATE\_TALKING | 话权按钮处于讲话状态 |
| SESSION\_CONNECT | SESSION\_CONNECT\_RESULT\_OK | 连接成功 |
| SESSION\_CONNECT\_RESULT\_FAIL | 由于网络或服务器原因连接失败 |
| SESSION\_CONNECT\_RESULT\_LIMIT | 呼叫被限制 |

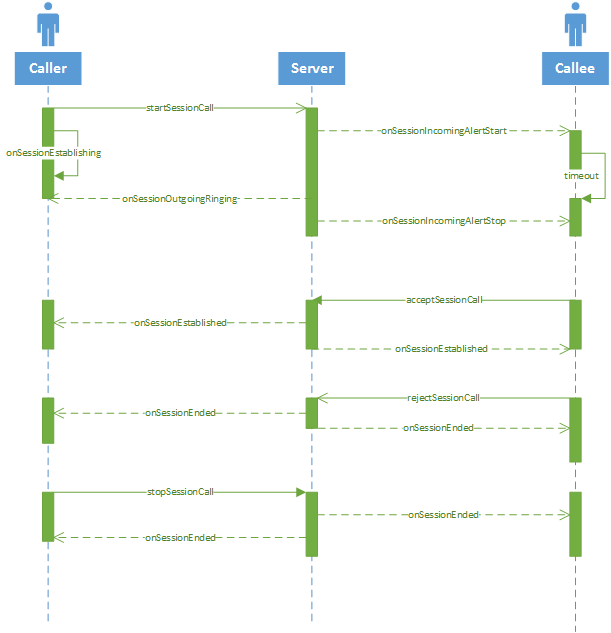
### 接口方法和事件回调

#### 会话管理

enum：com.ctchat.sdk.ptt..tool.intercom.IntercomManager

说明：会话管理类，通过该类可以发起会话和监听会话相关的事件

会话事件序列图：

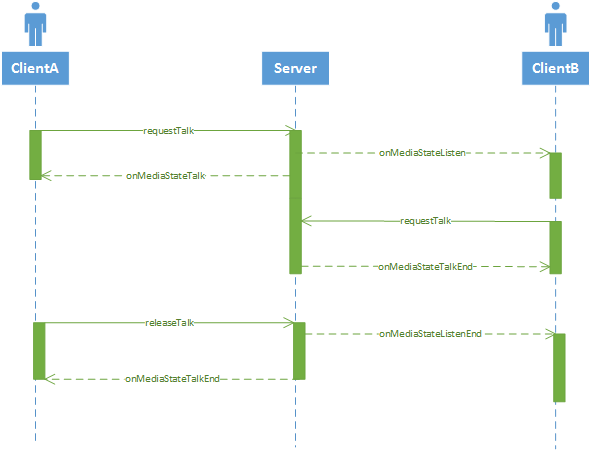


用户可以通过startSessionCall发起呼叫，该接口需要SessionEntity实例，该实例可以通过ContactEntity获取，呼叫成功发起时主叫方会先后收到onSessionEstablishing和onSessionOutgoingRinging事件，此时被叫方会收到onSessionIncomingAlertStart回调事件，被叫方接听超时会触发onSessionIncomingAlertStop回调事件。

用户可以用acceptSessionCall接听会话，接听后主叫和被叫方会接受到onSessionEstablished回调事件，用户也可以用rejectSessionCall拒接会话，拒接后主叫和被叫方会收到onSessionEnded回调事件

用户可以通过stopSessionCall来退出会话，调用该方法主叫和被叫方会收到onSessionEnded回调事件

话权事件序列图：



话权事件主要通过实现OnMediaEventListener接口获的，会话成员在调用requestTalk方法获取到话权后，会收到onMediaStateTalk回调事件，此时其他会话成员会收到onMediaStateListen回调事件。

在会话成员中如果有人已获取了话权，此时其他成员通过requestTalk申请话权时，则会收到onMediaStateTalkEnd的回调事件。

会话成员可以通过releaseTalk方法来释放话权，调用话权释放方法前，如果已抢到话权，则会收到OnMediaStateTalkEnd回调事件，其他成员收到onMediaStateListenEnd事件。

##### 单例对象和设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 枚举值：INSTANCE | | 访问会话管理类方法需要该值 |
| 参数 | 无 |  |
| 返回值 | IntercomManager | IntercomManage实例 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerSessionEventListener(OnSessionEventListener) | 关心会话事件需要注册该事件，详细参见OnSessionEventListener说明 |

|  |  |
| --- | --- |
| 取消注册监听回调接口 | 描述 |
| unRegisterSessionEventListener() | 取消注册会话事件监听接口 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerMediaEventListener(OnMediaEventListener) | 关心会话话权事件需要注册该事件，详细参见OnMediaEventListener说明 |

|  |  |
| --- | --- |
| 取消注册监听回调接口 | 描述 |
| unRegisterMediaEventListener() | 取消注册会话话权事件监听接口 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerSessionIncomingListener(OnSessionIncomingListener) | 关心临时会话来电事件需要注册该事件，详细参见OnSessionIncomingListener说明 |

|  |  |
| --- | --- |
| 取消注册监听回调接口 | 描述 |
| unRegisterSessionIncomingListener() | 取消注册临时会话来电事件监听接口 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerAudioVisualizerListener(OnAudioVisualizerListener) | 关心会话声音频谱事件需要注册该事件，详细参见OnAudioVisualizerListener说明 |

|  |  |
| --- | --- |
| 取消注册监听回调接口 | 描述 |
| unRegisterAudioVisualizerListener() | 取消注册会话声音频谱事件监听接口 |

##### 前后台在线人数和成员总数获取接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getSessionStateCount | | 获取前后台在线人数和成员总数，接口只在会话建立后调用有效，针对预定义组会话需要在获取到预定义组成员后有效。 |
| 参数 | SessionEntity:sessionEntity | 会话实例 |
| 返回值 | SessionStateCount | 前后台在线人数和成员总数存储，详细参见SessionStateCount类说明 |
| 函数回调 | 无 | 无 |

##### 临时呼叫接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：startSessionCall | | 发起临时单呼和组呼 |
| 参数 | SessionEntity:session | 用于发起会话的实例，可以通过被叫联系人实例获取会话实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 通过单个联系人获取会话实例接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getSessionEntityByContact | | 通过平台同步的单个联系人实例获取会话实例 |
| 参数 | ContactEntity:contact | 联系人实例 |
| 返回值 | SessionEntity | 会话实例 |
| 函数回调 | 无 |  |

##### 通过多个联系人获取会话实例接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getSessionEntityByContacts | | 通过平台同步的多个联系人实例获取会话实例,参数列表中不包含自己的联系人实例，且限制总个数不超过49个 |
| 参数 | List<ContactEntity>:contacts | 多个联系人实例 |
| 返回值 | SessionEntity | 会话实例 |
| 函数回调 | 无 |  |

##### 通过会话ID获取会话实例接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getSessionEntityBySessionId | | 通过会话ID获取会话实例 |
| 参数 | String:sessionId | 会话ID |
| 返回值 | SessionEntity | 会话实例 |
| 函数回调 | 无 |  |

##### 预定义组呼接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：startSessionCall | | 发起平台同步的预定义组呼叫 |
| 参数 | ChannelEntity:channel | 预定义组实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 通过预定义组获取会话实例接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getSessionEntityByChannel | | 通过平台同步的预定义组实例获取会话实例 |
| 参数 | ChannelEntity:channel | 预定义组实例 |
| 返回值 | SessionEntitiy | 会话实例 |
| 函数回调 | 无 |  |

##### 挂断会话接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：stopSessionCall | | 挂断临时呼叫和预定义组呼 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 会话中申请话权接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：requestTalk | | 会话中申请话权 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 会话中释放话权接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：releaseTalk | | 会话中释放话权 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 临时会话中邀请接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：inviteSessionCall | | 临时会话中邀请联系人加入会话 |
| 参数 | String:sessionId | 会话ID |
| List<ContactEntity>:contacts | 被邀请的联系人列表 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 会话锁定和解锁接口

说明：临时会话接通后SDK自动锁定会话，临时会话结束后SDK自动恢复上一次锁定得会话，当有多个临时会话时，按照先后顺序自动锁定和解锁，预定义组锁定和解锁会清除其它会话的锁定状态。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：lockGroupCall | | 针对多路正在进行中的会话，可以将指定的一路会话锁定，这样将只能听到被锁定的这路会话的实时语音，其它路会话的实时语音均不会播放。同一时间仅有一路会话可以锁定，当锁定其中一路会话后，其它路会话自动解锁。 |
| 参数 | SessionEntity:session | 会话实例 |
| boolean:locked | 是否锁定该会话 true:锁定 false:解锁 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 会话记录获取接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：getSessionEntityList | | 获取会话记录列表，列表数据存储相关的会话信息 |
| 参数 | 无 |  |
| 返回值 | List<SessionEntity> | 会话历史记录列表 |
| 函数回调 | 无 |  |

##### 删除单条会话记录接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：removeSessionEntity | | 删除单条会话记录 |
| 参数 | SessionEntity:session |  |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 清空历史会话记录接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法：cleanSessionEntityList | | 清空历史会话记录列表 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 临时会话来电拒绝接听接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 函数：rejectSessionCall | | 当临时会话来电的时候，用于拒绝接听临时会话来电 |
| 参数 | SessionEntity:session | 会话实例 |
| boolean:isBusy | 是否以忙碌的原因拒接来电，true为以忙碌的原因拒接，false为直接拒接来电 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 临时会话来电接入接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 函数：acceptSessionCall | | 当临时会话来电的时候，用于接听临时会话来电 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 判断是否有接通的临时会话接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 函数：hasSessionDialoging | | 判断是否有接通的临时会话 |
| 参数 | 无 | 无 |
| 返回值 | boolean | true:有会话接通中，false:空闲状态 |
| 函数回调 | 无 |  |

##### 判断是否有会话被锁定接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 函数：hasSessionLocked | | 判断是否有会话被锁定 |
| 参数 | 无 | 无 |
| 返回值 | boolean | true:有被锁定的会话，false:无被锁定的会话 |
| 函数回调 | 无 |  |

##### 获取临时会话实例及成员列表接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 函数：getSessionIdBySessionMember | | 根据临时成员列表获取临时会话实例及成员列表接口 |
| 参数 | List<ContactEntity> members | 临时成员列表 |
| 返回值 | 无 |  |
| 函数回调 | OnSessionEventListener.onSessionMemberUpdate | 返回会话实例以及成员列表 |

#### 事件回调接口

##### 会话事件回调

interface：com.ctchat.sdk.tool.intercom.OnSessionEventListener

说明：用户会话事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：临时会话主叫振铃 | | 描述 |
| 虚方法：onSessionOutgoingRinging | | 临时会话主叫振铃事件 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：会话开始建立 | | 描述 |
| 虚方法：onSessionEstablishing | | 临时或预定义组呼叫开始建立事件 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：会话建立 | | 描述 |
| 虚方法：onSessionEstablished | | 临时或预定义组会话建立事件 |
| 参数 | SessionEntity:session | 会话实例 |
| int:ret | 会话建立结果，参考常量标签：SESSION\_CONNECT |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：会话结束 | | 描述 |
| 虚方法：onSessionReleased | | 临时或预定义组会话结束事件 |
| 参数 | SessionEntity:session | 会话实例 |
| int:reason | 会话结束原因，参考常量标签SESSION\_RELEASE\_REASON |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：用户参与会话事件 | | 描述 |
| 虚方法：onSessionPresence | | 会话过程中，动态变化的用户参与此会话情况，均会通过此事件通知 |
| 参数 | SessionEntity:session | 会话实例 |
| List<ContactEntity>:membersAll | 会话预期所有人员 |
| List<ContactEntity>:membersPresence | 会话实际参与人员 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：会话成员变更事件 | | 描述 |
| 虚方法：onSessionMemberUpdate | | 会话过程中，成员数量变更事件 |
| 参数 | SessionEntity:session | 会话实例 |
| List<ContactEntity>:list | 会话成员列表 |
| boolean:b | 操作是否成功 |
| 返回值 | 无 |  |

##### 会话话权事件回调

interface：com.ctchat.sdk.tool.intercom.OnMediaEventListener

说明：会话中话权事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：本人开始讲话准备事件 | | 描述 |
| 虚方法：onMediaStateTalkPreparing | | 本人申请话语权时会第一个回调此事件,可用于 UI 对界面进行状态刷新 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：本人开始讲话事件 | | 描述 |
| 虚方法：onMediaStateTalk | | 本人获取到话权开始讲话事件 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：本人结束讲话事件 | | 描述 |
| 虚方法：onMediaStateTalkEnd | | 本人释放话权结束讲话事件 |
| 参数 | SessionEntity:session | 会话实例 |
| 参数 | int:reason | 本人会话结束原因 ，参考常量标签TALK\_END\_REASON |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：别人开始讲话事件 | | 描述 |
| 虚方法：onMediaStateListen | | 别人抢到话权开始讲话事件 |
| 参数 | SessionEntity:session | 会话实例 |
| ContactEntity:contact | 讲话人信息实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：别人讲话结束事件 | | 描述 |
| 虚方法：onMediaStateListenEnd | | 别人释放话权结束讲话事件 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：开始播放会话者声音的事件 | | 描述 |
| 虚方法：onMediaStateListenVoice | | 开始播放会话者声音的事件 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

##### 临时会话来电事件回调

interface：com.ctchat.sdk.tool.intercom. OnSessionIncomingListener

说明：来电会话事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：临时会话来电事件 | | 描述 |
| 虚方法：onSessionIncomingAlertStart | | 临时会话来电事件 |
| 参数 | SessionEntity:session | 会话实例 |
| ContactEntity:caller | 主叫人信息 |
| boolean:isAccepted | 是否已自动接听（当用户设置为自动应答时有效，SDK仅在当前仅存在一路临时呼叫是自动处理来电接听，如果有另外一路临时呼叫，SDK不执行自动接听，而是需要上层自行根据需求处理，即此时的isAccepted为false.） |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：临时会话来电结束事件 | | 描述 |
| 虚方法：onSessionIncomingAlertStop | | 临时会话来电提醒已结束,比如对方已挂断或取消等 |
| 参数 | SessionEntity:session | 会话实例 |
| 返回值 | 无 |  |

##### 会话声音频谱事件回调

interface：com.ctchat.sdk.tool.intercom. OnAudioVisualizerListener

说明：会话声音频谱事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：会话声音频谱事件 | | 描述 |
| 虚方法：onAudioVisualizerChanged | | 会话过程中声音流的频谱事件回调，可用于显示音频的频谱界面 |
| 参数 | byte[]:bytes | 频谱中每个采样值的大小，可结合num参数值将每个值取出用作显示 |
| int:num | 对应bytes参数的数组长度 |
| 返回值 | 无 |  |

## 即时消息

### 数据类型

#### 即时消息

class：com.ctchat.sdk.ptt.tool.entity.MessageEntity

说明：即时消息类

|  |  |  |  |
| --- | --- | --- | --- |
| 方法 | 描述 | 参数 | 返回值 |
| getType | 消息类型 |  | int:参考MESSAGE\_TYPE |
| getMessageCode | 消息唯一ID |  | String |
| getSessionId | 消息所在的会话ID |  | String |
| getState | 消息状态 |  | int:参考MESSAGE\_STATE |
| getBody | 消息文本内容 |  | String |
| getNameFrom | 消息发送方名称 |  | String |
| isSelf | 是否是自己发出的消息 | Context:context | boolean |
| getDate | 消息发送时的时间（yyyy-MM-dd） |  | String |
| getTime | 消息发送时的时间  （hh:mm:ss） |  | String |
| getImageUri | 消息的资源ID |  | String |
| getRecordTime | 语音消息的时长（单位:秒） |  | int |
| getRecordType | 语音消息的类型（区分普通语音和ptt语音） |  | int 参考RECORD\_TYPE |
| isRecordPlaying | 语音消息是否在播放 |  | boolean |
| setBody | 设置文本消息内容 | String:body | 无 |
| setImage | 设置图片字节数组 | byte[]:image | 无 |
| setVideoBytes | 设置视频字节数组 | byte[]:video | 无 |
| getLatitude | 获取位置消息纬度 |  | double：纬度 |
| getLongtitude | 获取位置消息经度 |  | double：经度 |
| getAddress | 获取位置消息地址 |  | String：地址 |

|  |  |  |
| --- | --- | --- |
| 常量 | 值 | 描述 |
| MESSAGE\_TYPE  （MessageType类下） | TYPE\_SYSTEM | 系统消息 |
| TYPE\_TEXT | 文字消息 |
| TYPE\_PICTURE | 图片消息 |
| TYPE\_RECORD | 录音消息 |
| TYPE\_VIDEO | 视频消息 |
| TYPE\_LOCATION | 位置消息 |
| MESSAGE\_STATE  （MessageEntity类下） | STATE\_NONE | 消息状态（无） |
| STATE\_NEW | 未读状态 |
| STATE\_RES\_DOING | 语音／图片／视频消息上传时的状态 |
| STATE\_RES\_FAILED | 语音／图片／视频上传失败的状态 |
| STATE\_SENDING | 消息发送中的状态 |
| STATE\_RESULT\_OK | 消息发送成功的状态 |
| STATE\_RESULT\_FAIL | 消息发送／下载失败的状态 |
| STATE\_DOWNLOADING | 消息下载中的状态 |
| RECORD\_TYPE  （MessageEntity类下） | RECORD\_TYPE\_NORMAL | 普通的录音消息 |
| RECORD\_TYPE\_PTT | PTT录音消息 |

### 接口方法和回调

#### 即时消息接口

class：com.ctchat.sdk.ptt.tool.chatIM.IMManager

说明：负责即时消息的操作接口

##### 单例对象和事件监听设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 静态方法：getInstance | | 访问该类中的其它方法，必须通过此静态方法获取实例进行访问 |
| 参数 | 无 |  |
| 返回值 | IMManager | IMManager实例 |

|  |  |
| --- | --- |
| 注册消息列表加载事件监听回调接口 | 描述 |
| registerEventListener (OnMessageEventListener listener) | 关心 OnMessageEventListener事件的实例，均需要通过此接口注册，否则将收不到相应事件 |

|  |  |
| --- | --- |
| 反注册消息列表加载事件监听回调接口 | 描述 |
| unregisterEventListener () | 不再关心 OnMessageEventListener事件的实例，需要通过此接口反注册 |

##### 每次加载消息数量的设置接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：setMessageListNumberMax | | 设置加载消息列表中一次性加载消息的数量（基于性能考虑，sdk不会一次性加载出会话所有的即时消息列表，如需查看历史消息，需要调用加载更多消息接口按该方法指定的加载数量分批次加载） |
| 参数 | Int：num | 每次加载的消息数 |
| 返回值 | 无 |  |

##### 加载更多消息接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：messageListMoreLoad | | 如需加载历史消息或查看更多历史消息则调用此接口 |
| 参数 | String:sessionId | 要加载的更多消息会话的sessionId |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageListLoad | 接收即时消息列表 |

##### 发送文本消息接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendMessageBySession | | 向指定会话发送文字消息 |
| 参数 | SessionEntity:session | 发送消息的会话 |
| MessageEntity:message | 发送的文本消息（需要调用MessageEntity的setBody接口传入要发送的文本内容） |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendMessageByChannel | | 向预定义组发送文字消息 |
| 参数 | ChannelEntity:channel | 发送消息的预定义组 |
| MessageEntity:message | 发送的文本消息（需要调用MessageEntity的setBody接口传入要发送的文本内容） |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

##### 语音消息录制接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：startRecordBySession | | 开始会话的语音消息录制 |
| 参数 | SessionEntity:session | 开始录制语音消息的会话 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageRecordStart | 语音消息开始录制事件 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：startRecordByChannel | | 开始预定义组会话的语音消息录制 |
| 参数 | ChannelEntity:channel | 开始录制语音消息的预定义组会话 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageRecordStart | 语音消息开始录制事件 |

##### 语音消息停止录制接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：stopRecord | | 结束或取消语音消息录制 |
| 参数 | boolean:cancel | true:为取消录制不进行发送 false:停止录制并进行发送 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageRecordStop | 语音消息停止录制事件 |

##### 语音消息重发接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：resendRecordMessageBySession | | 会话重发本地缓存的语音消息 |
| 参数 | SessionEntity:session | 重发语音消息的会话 |
| MessageEntity:message | 重发的语音消息（需要指定重发语音消息的  messageCode字段：来自于MessageEntity.getMessageCode()、  imageUri资源id字段：来自于MessageEntity.getImageUri()、  imageLength语音消息时长字段：来自于MessageEntity.getRecordTime()） |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：resendRecordMessageByChannel | | 预定义会话重发本地缓存的语音消息 |
| 参数 | ChannelEntity:channel | 重发语音消息的预定义组会话 |
| MessageEntity:message | 重发的语音消息（需要指定重发语音消息的  messageCode字段：来自于MessageEntity.getMessageCode()、  imageUri资源id字段：来自于MessageEntity.getImageUri()、  imageLength语音消息时长字段：来自于MessageEntity.getRecordTime()） |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

##### 图片消息发送接口

说明：要发送的图片类型要在允许的范围内，默认允许的格式为jpg、png，否则不会进行发送；文件大小要在允许的范围内，默认大小限制为25MB，否则不会进行发送。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendImageBySession | | 会话发送图片消息接口 |
| 参数 | SessionEntity:session | 发送图片消息的会话 |
| MessageEntity:message | 要发送的图片消息（需要调用MessageEntity.setImage()接口传入要发送到图片字节数组、调用MessageEntity.setLocalPath()接口传入要发送的图片文件的本地路径） |
| 返回值 | boolean | 是否允许发送：true：允许，false：不允许 |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendImageByChannel | | 预定义组发送图片消息接口 |
| 参数 | ChannelEntity:channel | 发送图片消息的预定义组会话 |
| MessageEntity:message | 要发送的图片消息（需要调用MessageEntity.setImage()接口传入要发送到图片字节数组、调用MessageEntity.setLocalPath()接口传入要发送的图片文件的本地路径） |
| 返回值 | boolean | 是否允许发送：true：允许，false：不允许 |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

##### 视频消息发送接口

说明：要发送的视频类型要在允许的范围内，默认允许的格式为mp4、3gp，否则不会进行发送；文件大小要在允许的范围内，默认大小限制为25MB，否则不会进行发送。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendVideoBySession | | 会话发送视频消息接口 |
| 参数 | SessionEntity:session | 发送视频消息的会话 |
| MessageEntity:message | 要发送的视频消息（需要调用MessageEntity.setVideoBytes()接口传入要发送的视频文件字节数组、调用MessageEntity.setLocalPath()接口传入要发送的视频文件的本地路径） |
| 返回值 | boolean | 是否允许发送：true：允许，false：不允许 |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendVideoByChannel | | 预定义组发送视频消息接口 |
| 参数 | ChannelEntity:channel | 发送视频消息的预定义组会话 |
| MessageEntity:message | 要发送的视频消息（需要调用MessageEntity.setVideoBytes()接口传入要发送的视频文件字节数组、调用MessageEntity.setLocalPath()接口传入要发送的视频文件的本地路径） |
| 返回值 | boolean | 是否允许发送：true：允许，false：不允许 |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

##### 位置消息发送接口

说明：要发送的位置坐标需要转换为地球坐标系的位置坐标。

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendLocationBySession | | 会话发送视频消息接口 |
| 参数 | SessionEntity:session | 发送图片消息的会话 |
| double:latitude | 坐标纬度 |
| double:longtitude | 坐标经度 |
| String:address | 坐标地点名称 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：sendLocationByChannel | | 预定义组会话发送位置消息接口 |
| 参数 | ChannelEntity:channel | 发送位置消息的预定义组会话 |
| double:latitude | 坐标纬度 |
| double:longtitude | 坐标经度 |
| String:address | 坐标地点名称 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageOutgoingSent | 消息发送事件 |

##### 语音消息播放接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：startPlayRecordMessage | | 开始播放一条指定的语音消息（播放完成后、会自动停止） |
| 参数 | MessageEntity:message | 需要播放的语音消息实例 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageRecordPlayLoading或OnMessageEventListener.onMessageRecordPlayStart | 若语音消息还未缓存，则会先从网络服务中预加载语音数据 |

##### 语音消息停止播放接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：stopPlayRecordMessage | | 停止正在播放的语音消息 |
| 参数 | 无 |  |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener.onMessageRecordPlayStop |  |

##### 本地系统消息生成接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：generateSystemMessage | | 生成本地系统消息（文字），第三方应用可通过此接口在即时消息列表中生成一些会话相关的系统消息，比如“呼叫未接”等 |
| 参数 | SessionEntity:session | 生成本地消息的会话 |
| String:messageBody | 消息内容 |
| boolean:isNew | true设置为新消息 false设置为已读消息 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener. onMessageIncomingRecv | 接收到消息事件 |

##### 单条消息删除接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：messageRemove | | 删除会话中的指定消息 |
| 参数 | String:sessionId | 会话id |
| String:msgCode | 消息id |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 清除指定会话消息接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：messageRemoveAll | | 删除会话的所有即时消息 |
| 参数 | String:sessionId | 会话id |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 清除会话中未读消息个数接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：cleanMessageUnreadCount | | 清除指定会话的未读消息个数 |
| 参数 | SessionEntity:session | 要清除未读消息个数的会话实例 |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 更多消息清理接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：messageListMoreClean | | 清除缓存中指定会话预加载的聊天记录数据 |
| 参数 | String:sessionId | 会话id |
| 返回值 | 无 |  |
| 函数回调 | 无 |  |

##### 图片下载接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：downloadImage | | 媒体共享接收失败后重新下载图片 |
| 参数 | MessageEntity messageEntity | 消息实例 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener.onMessageUpdated | 下载成功回调 |

##### 视频下载接口

|  |  |  |
| --- | --- | --- |
| 方法 | | 描述 |
| 方法名：downloadVideo | | 媒体共享接收失败后重新下载视频 |
| 参数 | MessageEntity messageEntity | 消息实例 |
| 返回值 | 无 |  |
| 函数回调 | OnMessageEventListener.onMessageUpdated | 下载成功回调 |

#### 回调接口

##### 即时消息事件回调

class：com.ctchat.sdk.ptt.tool.chatIM.OnMessageEventListener

说明：消息的事件回调，在接收到事件回调后通过MessageEntity.getState()接口可查询消息所处的状态。

|  |  |  |
| --- | --- | --- |
| 事件：收到新消息事件 | | 描述 |
| 虚方法：onMessageIncomingRecv | | 接收到新消息事件 |
| 参数 | MessageEntity: message | 接收到的新消息 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：收到新消息事件 | | 描述 |
| 虚方法：onMessageIncomingRecv | | 接收到多条新消息事件 |
| 参数 | List<MessageEntity>: list | 接收到的新消息列表 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：消息发送中的事件 | | 描述 |
| 虚方法：onMessageOutgoingSent | | 消息发送中状态变更 |
| 参数 | MessageEntity:message | 发送状态变更的消息实例  注意: 此事件可能会被触发多次，每次返回的是发送的中间状态通过getState获得，具体状态值，参照MESSAGE\_STATE |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：消息下载时的事件 | | 描述 |
| 虚方法：onMessageUpdated | | 消息的状态变更事件 |
| 参数 | MessageEntity:message | 消息状态变更事件 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息开始录制事件 | | 描述 |
| 虚方法：onMessageRecordStart | | 语音消息开始录制事件 |
| 参数 | 无 |  |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息开始录制事件 | | 描述 |
| 虚方法：onMessageRecordStop | | 语音消息停止录制事件 |
| 参数 | int :seconds | 录音成功则返回录音的秒数  其它可能的返回值：  -2：录音未满1秒以上  -3：录音被取消  特别说明:录音时间长度设定为最小 1 秒钟，最大 60 秒钟 |
| String:msgCode | 语音消息的code |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息开始录制事件 | | 描述 |
| 虚方法：onMessageRecordTransfered | | 语音消息录音数据传输完成事件 |
| 参数 | String:msgCode | 语音消息的code |
| String:resId | 资源id |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息开始录制事件 | | 描述 |
| 虚方法：onMessageRecordPlayLoading | | 语音消息播放开始加载数据事件 |
| 参数 | String :msgCode | 语音消息的code |
| String:resId | 资源id |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息开始录制事件 | | 描述 |
| 虚方法：onMessageRecordPlayLoaded | | 语音消息播放加载完成数据事件 |
| 参数 | boolean:isOk | 是否加载成功 |
| String :msgCode | 语音消息的code |
| String:resId | 资源id |
| byte[]:resBytes | 资源字节数组 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息开始播放事件 | | 描述 |
| 虚方法：onMessageRecordPlayStart | | 语音消息开始播放事件 |
| 参数 | String :msgCode | 语音消息的code |
| String:resId | 资源id |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：语音消息停止播放事件 | | 描述 |
| 虚方法：onMessageRecordPlayStop | | 语音消息停止播放事件 |
| 参数 | String :msgCode | 语音消息的code |
| String:resId | 资源id |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：实时通话的本地录音完成事件 | | 描述 |
| 虚方法：onMessagePttRecord | | 实时通话的本地录音完成事件 |
| 参数 | SessionEntity:session | Ptt语音所在的session |
| MessageEntity:message | Ptt语音消息 |
| String:msgCode | 消息的Code |
| String:resId | Ptt语音消息的资源id |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：消息列表加载完成事件 | | 描述 |
| 虚方法：onMessageListLoad | | 消息列表加载完成事件 |
| 参数 | String:sessionId | 加载消息列表的会话ID |
| List<MessageEntity>: messageEntityList | 加载的消息列表 |
| 返回值 | 无 |  |

## 消息推送

### 数据类型

### 接口方法和事件回调

#### 消息推送管理

enum：com.ctchat.sdk.ptt.tool.MessagePush.PushManager

说明：消息推送管理类

##### 单例对象和设置接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 枚举值：INSTANCE | | 访问其他类需要该值 |
| 参数 | 无 |  |
| 返回值 | PushManager | PushManager实例 |

|  |  |
| --- | --- |
| 注册监听回调接口 | 描述 |
| registerPushEventListener(OnPushEventListener) | 关心心跳推送事件需要注册该事件，详细参见OnPushEventListener说明 |

|  |  |
| --- | --- |
| 去除注册监听回调接口 | 描述 |
| unRegisterPushEventListener(OnPushEventListener) | 去除心跳推送事件监听注册 |

#### 事件回调接口

##### 消息推送事件回调

interface：com.ctchat.sdk.ptt.tool.MessagePush.OnPushEventListener

说明：心跳消息推送事件回调接口

|  |  |  |
| --- | --- | --- |
| 事件：远程登录消息推送 | | 描述 |
| 虚方法：onPushRemoteLogin | | 当用户账号远程登陆后收到消息推送 |
| 参数 | 无 |  |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：账户心跳事件 | | 描述 |
| 虚方法：onHeartbeat | | 在运行过程中，网络变化引起的或与服务器实时连接引起的异常变化，会自动产生此事件通知 |
| 参数 | int:i | 参见常量标签ERROR\_CODE:LOGIN\_SUCCESS,LOGIN\_ERROR\_SERVER\_UNAVAILABLE,LOGIN\_ERROR\_NETWORK,LOGIN\_ERROR\_INVALID\_ACCOUNT |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：用户行为控制消息推送 | | 描述 |
| 虚方法：onPushSystemUserCtrl | | 当用户账号远程登陆后收到消息推送 |
| 参数 | int:cmd | 参数常量PushManager.ACCOUNT\_TURN\_OFF表示账号被遥毙，PushManager.ACCOUNT\_TURN\_ON表示账号被复活，PushManager.ACCOUNT\_TURN\_RESET表示账号被重置需要重新登录 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：用户呼入呼出限制 | | 描述 |
| 虚方法：onPushSystemCallLimit | | 用户呼叫限制（限制呼入、解除呼入、限制呼出、解除呼出限制、预定义组通话限制、解除预定义组通话限制）注：当处于会话中的会话仍可以讲话。 |
| 参数 | int:i | 用户受限制或解除限制的action(操作类型：0：限制临时会话呼入 1：限制临时会话呼出 2：限制呼入预定义组 10：解除临时会话呼入限制 11：解除临时会话呼出限制 12：解除预定义组呼入限制); |
| String:s | 空字符串 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：接收到群呼上线事件 | | 描述 |
| 虚方法：onPushChannelAlert | | 收到了指定预定义组的群呼上线通知 |
| 参数 | String：channelID | 预定义组的ID |
| String：channelName | 预定义组名称 |
|  | String： initiator | 发起人 |
| 返回值 | 无 |  |

|  |  |  |
| --- | --- | --- |
| 事件：指定的预定义组被限制/恢复 | | 描述 |
| 虚方法：onPushChannelLimit | | 平台侧对指定的预定义组进行了限制或恢复操作。注：被限制的预定义组会话实例会从SDK的预定义组缓存列表中进行清除，被恢复的预定义组会话实例会被添加至SDK的预定义组缓存列表中。 |
| 参数 | String：channelID | 指定的预定义组ID |
|  | String：channelName | 指定的预定义组名称 |
|  | int：operation | 限制或恢复操作：PushManager.CHANNEL\_TRUN\_ON：指定的预定义组被恢复；  PushManager.CHANNEL\_TURN\_OFF：指定的预定义组被限制； |
| 返回值 | 无 | 无 |

## 参数设置

### 数据类型

3.10.1.1数据常量

|  |  |  |
| --- | --- | --- |
| Class | com.ctchat.sdk.ptt.tool.setting.SettingManager | |
| 标签 | 常量 | 描述 |
| AUDIO\_AMR\_MODE | SESSION\_AUDIO\_AMR\_MODE\_0 | 默认会话音质，等级越高音质越高 |
| SESSION\_AUDIO\_AMR\_MODE\_1 | 会话音质等级1 |
| SESSION\_AUDIO\_AMR\_MODE\_2 | 会话音质等级2 |
| SESSION\_AUDIO\_AMR\_MODE\_3 | 会话音质等级3 |
| SESSION\_AUDIO\_AMR\_MODE\_4 | 会话音质等级4 |
| SESSION\_AUDIO\_AMR\_MODE\_5 | 会话音质等级5 |
| SESSION\_AUDIO\_AMR\_MODE\_6 | 会话音质等级6 |
| SESSION\_AUDIO\_AMR\_MODE\_7 | 会话音质等级7 |

### 接口方法和事件回调

#### 参数设置管理

enum：com.ctchat.sdk.ptt.tool.setting.SettingManager

说明：参数设置管理类，第三方应用可以通过设置接口，设置会话应答方式：自动应答和手动应答、设置免打扰、设置音频质量和音频放大器以及静音模式、设置心跳频度等。

##### 单例对象获取接口

|  |  |  |
| --- | --- | --- |
| 单例访问 | | 描述 |
| 枚举值：INSTANCE | | 访问其他类需要该值 |
| 参数 | 无 |  |
| 返回值 | SettingManager | SettingManager实例 |

##### 设置会话应答方式接口

说明：设置自动应答后，在临时会话来电时，不需要调用acceptSessionCall方法来接入会话，否则需要调用方法来接入会话。

|  |  |  |
| --- | --- | --- |
| 方法：setSessionAutoAnswerMode | | 设置会话的应答方式 |
| 参数 | int:mode | 自动应答：SessionEntity.ANSWER\_MODE\_AUTO |
| 手动应答：SessionEntity.ANSWER\_MODE\_MANNUAL |
| 返回值 | 无 |  |

##### 免打模式扰设置接口

|  |  |  |
| --- | --- | --- |
| 方法：setSessionSilentMode | | 设置免打扰模式开开关 |
| 参数 | boolean:isSilent | true:开启免打扰模式；false：关闭免打扰模式 |
| 返回值 | 无 |  |

##### 增强会话音质设置接口

|  |  |  |
| --- | --- | --- |
| 方法：setSessionAudioQuailty | | 设置会话音质 |
| 参数 | int:mode | 参考AUDIO\_AMR\_MODE，设置等级越高音质越强 |
| 返回值 | 无 |  |

##### 设置静音模式接口

|  |  |  |
| --- | --- | --- |
| 方法：setMuteMode | | 设置静音模式开关 |
| 参数 | boolean:flag | true:开启静音模式；false:关闭静音模式 |
| 返回值 | 无 |  |

##### 设置心跳频度接口

说明：设置的心跳频度要位于用户信息(UserInfo类)中的心跳频度上下限之间，否则会设置失败。

|  |  |  |
| --- | --- | --- |
| 方法：setHeartBeatFrequency | | 设置心跳频度 |
| 参数 | int:seconds | 心跳频度值，单位秒。 |
| 返回值 | boolean | true:设置成功；false：设置失败 |

# SDK接入说明

## 导入SDK

4.1.1 将aar文件放入Android Studio项目所要使用CTChatSDK的Module所在的Libs里面。

4.1.2 在Module的build.gradle里面将libs加入仓库（repositories）：

repositories{  
     flatDir{  
         dirs 'libs' //就是你放aar的目录地址  
     }  
}

4.1.3 在Module的build.gradle里面添加依赖（dependencies）：

dependencies {  
     ....//你的其他依赖  
     compile name:'CTChatSDK-<version>',ext:'aar'  
}

## 添加代码

4.2.1 配置AndroidManifest.xml，添加以下权限：

<uses-permission android:name="android.permission.READ\_PHONE\_STATE" />  
<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE" />  
<uses-permission android:name="android.permission.READ\_SMS" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />  
<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

## 混淆设置

请在您的混淆脚本中添加如下的配置，跳过对CTChatSDK的混淆操作：

**-keep** class com.ptt.sdk.\*\*{\*;}  
**-dontwarn** com.ptt.sdk.\*\*  
**-keep** class com.ctchat.sdk.\*\*{\*;}  
**-dontwarn** com.ctchat.sdk.\*\*